SIAM J. ALG. DISC. METH.
Vol. 6, No. 1, January 1985

POLYA'S URN MODEL AND COMPUTER AIDED GEOMETRIC DESIGN*

RONALD N. GOLDMAN†

Abstract. In this paper Polya's urn model is used to generate blending functions for computer aided geometric design. There are over a dozen geometric properties which are currently considered to be desirable for computer aided geometric design. Curves and surfaces which use blending functions generated from Polya's urn model are shown to share many of these geometric properties. Derivations of these geometric properties are traced back to the probabilistic interpretation of the blending functions.

CR categories and subject description: I3.5 (Computer Graphics): Computational Geometry and Object Modeling-Curve Representations

Key words. discrete probability distribution, urn model

1. Introduction. Typical in computer aided geometric design is the following interaction between system and user.

1. A designer introduces a collection of control points which to him describe the shape of some curve or surface.
2. The system loosely approximates the designer's intent applying some internal scheme, the details of which are often hidden from the user, to construct a smooth curve or surface.
3. The designer moves some of his control points in directions which he feels intuitively will help the computer to improve its approximation.
4. The system modifies the curve or surface in an attempt to conform more closely to the designer's intent.

Steps 3, 4 may be repeated several times until the user is satisfied.

Generally, systems try to approximate a designer's intent by using an internal collection of predefined functions to blend smoothly the designer's control points. Thus, given an ordered collection of points $P_0, \cdots, P_n$, a system will construct a curve

$$P(t) = \sum_k B_k^n(t) P_k, \qquad 0 \leqq t \leqq 1,$$

where $B_0^n(t), \cdots, B_n^n(t)$ are an internal collection of predefined blending functions. Lagrange polynomials, Bezier curves and B-splines are all defined precisely in this manner.

The final shape of the curve depends both on the blending functions available to the system and on the control points selected by the user, but the geometric properties of the curve depend only on the blending functions. Thus it is natural to ask:

1. What geometric properties do we wish to build into our curves?
2. Where can we find suitable blending functions?

With the advantages of experience and hindsight we shall try to answer the first question. The answer to the second question is the main theme of this paper.

This paper is divided into two main parts. Section 2 reviews in detail the geometric properties which are desirable for the curves of computer aided geometric design. Readers already familiar with computer aided geometric design may quickly skim this section. The remainder of this paper is devoted to our main theme: the intimate

* Received by the editors, April 5, 1982, and in revised form September 15, 1983.
† Control Data, Arden Hills, Minnesota 55112.

relationship between classical discrete probability theory and computer aided geometric design. This discussion commences in § 3. Knowledgeable readers may wish to begin here.

2. Computer aided geometric design. We wish to determine which curves

$$P(t) = \sum_k B_k^n(t) P_k, \qquad 0 \leqq t \leqq 1,$$

defined by a collection of control points $P_0, \cdots, P_n$ and a collection of blending functions $B_0^n(t), \cdots, B_n^n(t)$ are suitable for computer aided geometric design. We shall begin by listing those properties which experience has shown us are desirable. We shall then discuss each property in turn to determine its implications for the blending functions. The *desirable properties for the curves of computer aided geometric design* are the following: 1. well-defined, 2. convex hull, 3. smooth, 4. interpolates end points, 5. extends to surfaces, (a. rectangular, b. triangular), 6. symmetry, 7. geometric construction algorithm, 8. exactly reproduces points and lines, 9. nondegenerate, 10. subdivision algorithm, 11. augmentation algorithm, 12. variation diminishing and 13. local control.

2.1. Well-defined. A curve

$$P(t) = \sum_k B_k^n(t) P_k, \qquad 0 \leqq t \leqq 1$$

is said to be well-defined iff it depends only on the points $P_0, \cdots, P_n$ and not on the choice of the coordinate origin. Equivalently, the curve $P(t)$ is well-defined iff translating each point P_k by the same vector v, translates the entire curve by the same vector v; that is, iff

$$P(t)_{\text{new}} = P(t)_{\text{old}} + v.$$

Thus for every vector v and every parameter t,

$$\sum_k B_k^n(t) P_k + [\sum_k B_k^n(t)] v = \sum_k B_k^n(t)(P_k + v) = P(t)_{\text{new}} = P(t)_{\text{old}} + v = \sum_k B_k^n(t) P_k + v,$$

so

$$[\sum_k B_k^n(t)] v = v$$

for every vector v and every parameter t. But this can happen iff

$$\sum_k B_k^n(t) = 1$$

for every value of t. Thus we have shown that

$$P(t) \text{ is well-defined} \Leftrightarrow \sum_k B_k^n(t) = 1.$$

That the curves must be well-defined is crucial for computer aided geometric design. A designer wishes to describe a shape. He should not have to be concerned with artifacts of the system such as the current position of the coordinate origin, and he would certainly be astonished if identical collections of control points generated physically distinct curves merely due to some internal change in the system. Therefore to be effective for computer aided geometric design, a curve must be well-defined.

2.2. Convex hull. A set S is said to be convex iff whenever 2 points P, Q lie in S, the entire line segment PQ lies within S. The intersection of convex sets is clearly convex. By definition, the convex hull of a set S is the intersection of all the convex sets which contain S. Equivalently, the convex hull of S is the smallest convex set which contains S. It follows easily by induction on n that for any collection of points $P_0, \cdots, P_n$

$$\text{convex hull } (P_0, \cdots, P_n) = \{\sum_k c_k P_k | c_k \geqq 0 \text{ and } \sum_k c_k = 1\}.$$

Thus a well-defined curve

$$P(t) = \sum_k B_k^n(t) P_k$$

lies in the convex hull of the points $P_0, \cdots, P_n$ iff

$$B_k^n(t) \geqq 0, \qquad 0 \leqq t \leqq 1.$$

That is, for well-defined curves

$$\text{convex hull property} \Leftrightarrow B_k^n(t) \geqq 0, 0 \leqq t \leqq 1.$$

Since in computer aided geometric design the points define the curve, there must be some obvious relationship between the exact location of the control points and the approximate location of the actual curve. The convex hull property localizes the curve to the proximity of its control points. This feature is of great practical importance for computer assisted geometric design.

2.3. Smooth. A curve is said to be smooth iff it is differentiable. The more derivatives it has the smoother it is said to be. For a curve

$$P(t) = \sum B_k^n(t) P_k$$

the derivative is

$$P'(t) = \sum \frac{dB_k^n}{dt} P_k.$$

Thus

$$P(t) \text{ is smooth} \Leftrightarrow B_k^n(t) \text{ is differentiable.}$$

That the curves used in computer aided geometric design must generally be smooth is obvious; usually several derivatives are required. Thus the differentiability of the blending functions is a crucial characteristic of these curves.

2.4. Interpolates end points. In computer aided geometric design we do not require that the curve pass through all the points specified by the designer. After all, the designer only uses the points to describe the general flow of the curve, not its exact location. We are trying to approximate shape, not interpolate position. However the start and the end points are special. Where else could the curve start but at the designers first point; where else could it terminate but at his last point? After the initial point, he may wish to indicate only the general flow of the curve, but he may as well tell us exactly where to start; why make us guess? Similarly, from symmetry considerations (see below), he may as well indicate the exact terminus of his curve.

Aside from these arguments, there is an even more compelling reason to insist that curves exactly interpolate their first and last control points. Often a designer will

wish to attach several curves end to end. If their end points are not exactly specified by the user, it would be extremely difficult for the system to insure continuity between contiguous curves. However for curves which pass through their end points, the user can enforce continuity simply by selecting the last point of his previous curve as the first point of his next curve.

A curve

$$P(t) = \sum_k B_k^n(t) P_k, \qquad 0 \leqq t \leqq 1$$

passes through its end points P_0, P_n iff

$$P(0) = P_0, \qquad P(1) = P_n.$$

Thus, in general, $P(t)$ interpolates its end points iff

$$B_k^n(0) = \begin{cases} 0, & k \neq 0, \\ 1, & k = 0, \end{cases} \quad \text{and} \quad B_k^n(1) = \begin{cases} 0, & k \neq n, \\ 1, & k = n. \end{cases}$$

2.5. Extensions to surfaces. A sequence of control points defines a curve; a grid of control points defines a surface. In computer aided geometric design the grid is usually either triangular or rectangular depending upon whether the designer wishes to construct 3 sided or 4 sided surface patches. Given a curve

$$P(t) = \sum B_k^n(t) P_k, \qquad 0 \leqq t \leqq 1,$$

we say that it has extensions to surfaces iff there are surfaces

$$Q(u, v) = \sum B_{ij}^n(u, v) Q_{ij},$$

whose boundary curves have the same blending functions as $P(t)$ and whose control points are the boundary points of the grid. The trick is to find nontrivial blending functions $B_{ij}^n(u, v)$ which either vanish or reduce to $B_k^n(t)$ on the boundaries.

For curves which interpolate their end points it is always possible to generate rectangular extensions simply by defining

$$B_{ij}^n(u, v) = B_i^n(u) B_j^n(v), \qquad Q(u, v) = \sum B_i^n(u) B_j^n(v) Q_{ij}, \qquad 0 \leqq u, v \leqq 1.$$

On the boundaries we get

$$Q(0, v) = \sum B_j^n(v) Q_{0j}, \qquad Q(1, v) = \sum B_j^n(v) Q_{nj},$$
$$Q(u, 0) = \sum B_i^n(u) Q_{i0}, \qquad Q(u, 1) = \sum B_i^n(u) Q_{in},$$

as required. The surface $Q(u, v)$ is called the tensor product surface, and it is standard in computer aided geometric design.

Nondegenerate triangular patches are more difficult to generate. Given a triangular grid $\{Q_{ij}\}$, $i + j \leqq n$, we need to construct a well-defined, nondegenerate surface

$$Q(u, v) = \sum B_{ij}^n(u, v) Q_{ij}, \qquad 0 \leqq u + v \leqq 1$$

such that on the boundaries

$$Q(0, v) = \sum B_j^n(v) Q_{0j}, \quad Q(u, 0) = \sum B_i^n(u) Q_{i0}, \quad Q(u, 1-u) = \sum B_i^n(u) Q_{i,n-i}.$$

In general this can happen iff there exist blending functions $B_{ij}^n(i, v)$ which satisfy

$$B_{ij}^n(0, v) = \begin{cases} 0, & i \neq 0, \\ B_j^n(v), & i = 0, \end{cases} \qquad B_{ij}^n(u, 0) = \begin{cases} 0, & j \neq 0, \\ B_i^n(u), & j = 0, \end{cases}$$

$$B_{ij}^n(u, 1-u) = \begin{cases} 0, & i+j \neq n, \\ B_i^n(u), & i+j = n. \end{cases}$$

Recently triangular Bezier patches have been studied by Sabin [11], by Farin [4], [5] and by Goldman [8].

2.6. Symmetry. The order in which a designer selects his control points is critical in the determination of his intent. In general, the very same control points chosen in different order will generate very different curves (see Fig. 1).

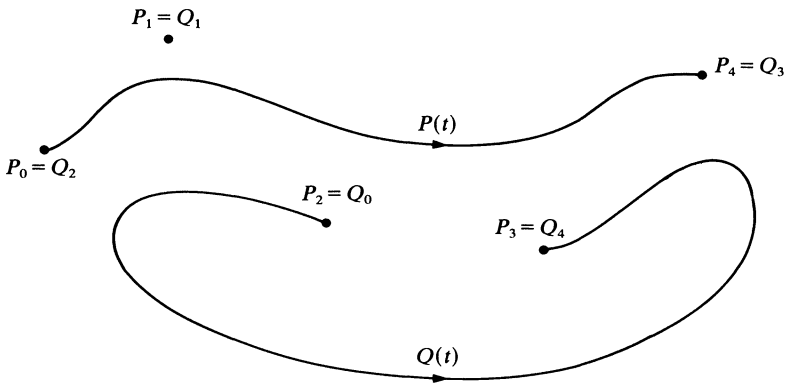


FIG. 1

However a strict inversion of order should not lead to a distinctly different curve, but only to a reversal in orientation (see Fig. 2).

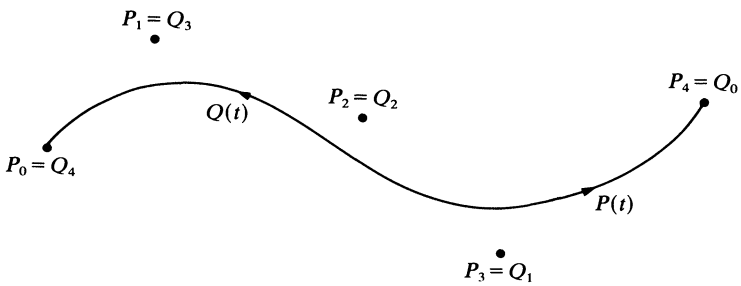


FIG. 2

This symmetry is part of a designer's natural intent and must therefore be captured by the curves used in computer aided geometric design.

Now a curve

$$P(t) = B[P_0, \cdots, P_n](t) = \sum B_k^n(t) P_k, \qquad 0 \leqq t \leqq 1$$

will have the required symmetry property iff

$$B[P_n, \cdots, P_0](t) = B[P_0, \cdots, P_n](1-t)$$

that is, iff

$$\sum B_k^n(t)P_{n-k} = \sum B_k^n(1-t)P_k.$$

In general, this will be true iff

$$B_k^n(t) = B_{n-k}^n(1-t).$$

Thus for curves

$$\text{symmetry} \Leftrightarrow B_k^n(t) = B_{n-k}^n(1-t).$$

Similar but more complex symmetry conditions will be required for the surfaces of computer aided geometric design.

2.7. Geometric construction algorithm. Let

$$P(t) = B[P_0, \cdots, P_n](t) = \sum B_k^n(t)P_k$$

be a curve defined by a collection of control points $P_0, \cdots, P_n$ and a collection of blending functions $B_0^n(t), \cdots, B_n^n(t)$. In general, the blending functions may be complicated expressions and therefore either difficult or expensive to evaluate. A geometric construction algorithm provides a simple, numerically stable technique for evaluating $P(r)$ for any parameter r.

The basic idea is to construct, recursively, a triangular array of points $\{P_k^L(r)\}$, $k+L \leqq n$, such that:

1. $P_k^0(r) = P_k$;
2. $P_k^L(r)$ lies on the straight line joining $P_k^{L-1}(r)$ and $P_{k+1}^{L-1}(r)$;
3. $P_0^n(r) = P(r)$;

(see Fig. 3).

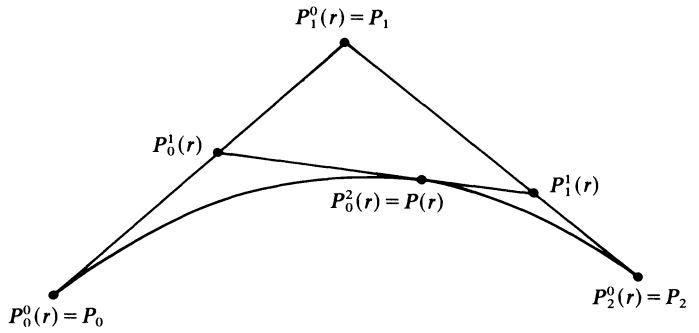


FIG. 3. *Geometric construction algorithm.*

Since $P_k^L(r)$ lies on the straight line joining $P_k^{L-1}(r)$ and $P_{k+1}^{L-1}(r)$, there must be functions $f_k^{n-L}(r)$, $s_k^{n-L}(r)$ such that

$$P_k^L(r) = f_k^{n-L}(r)P_k^{L-1}(r) + s_k^{n-L}(r)P_{k+1}^{L-1}(r), \qquad f_k^{n-L}(r) + s_k^{n-L}(r) = 1.$$

Thus to compute $P(r)$, instead of evaluating the blending functions $B_k^n(r)$, we need only evaluate the functions $f_k^{n-L}(r)$, $s_k^{n-L}(r)$. The hope is that these functions are relatively simple and therefore easy and inexpensive to compute. But where do they come from, and how do we get our hands on them?

The answer, of course, is that somehow they must be related to the original blending functions. Till now, we have tacitly assumed that there exist collections of

blending functions $\{B_k^n(t)\}$ for every value of n, and that these collections of blending functions for the various n's are in someway related. We shall now show that if there is a simple, explicit, recursion formula relating the functions $\{B_k^{n+1}(t)\}$ to the functions $\{B_k^n(t)\}$, then a geometric construction algorithm exists, and the functions $f_k^{n-L}(r)$, $s_k^{n-L}(r)$ are just those that appear in the recursion formula.

Assume a simple recursion relation

$$B_k^{n+1}(t) = f_k^n(t)B_k^n(t) + s_{k-1}^n(r)B_{k-1}^n(t),$$

where

$$f_k^n(t) + s_k^n(t) = 1.$$

That is, assume that $B_k^{n+1}(t)$ can be constructed from the functions $B_k^n(t)$, $B_{k-1}^n(t)$ and some simple multiplier functions $f_k^n(t)$, $s_{k-1}^n(t)$. If we use these multiplier functions to construct the collection of points $\{P_k^L(r)\}$, then we can prove the following results.

LEMMA 2.1. $B[P_0^1(r), \cdots, P_{n-1}^1(r)](r) = B[P_0, \cdots, P_n](r)$.

Proof. $B[P_0^1(r), \cdots, P_{n-1}^1(r)](r) = \sum B_k^{n-1}(r)P_k^1(r)$
$= \sum B_k^{n-1}(r)[f_k^{n-1}(r)P_k + s_k^{n-1}(r)P_{k+1}] = \sum [f_k^{n-1}(r)B_k^{n-1}(r) + s_{k-1}^{n-1}(r)B_{k-1}^{n-1}(r)]P_k$
$= \sum B_k^n(r)P_k = B[P_0, \cdots, P_n](r)$. □

LEMMA 2.2. $B[P_0^k(r), \cdots, P_{n-k}^k(r)](r) = B[P_0, \cdots, P_n](r)$.

Proof. This result follows immediately from Lemma 2.1 by induction on k. □

LEMMA 2.3. $P_0^n(r) = P(r)$.

Proof. This result follows immediately from Lemma 2.2 with $k = n$. □

Thus we have shown that

recursion formula⇒geometric construction algorithm.

2.8. Exactly reproduces points and lines. Suppose that a designer selects all his control points P_k at the same location P_0. Then he would expect his curve to collapse to the single point P_0. A curve is said to exactly reproduce points iff

$$P_k = P_0 \text{ for all } k \Rightarrow P(t) = P_0 \text{ for all } t.$$

Thus the curve $P(t)$ exactly reproduces points iff

$$[\sum B_k^n(t)]P_0 = P_0$$

or iff

$$\sum B_k^n(t) = 1.$$

Thus exactly reproducing points is equivalent to being well-defined.

Now suppose that a designer selects control points which are equally spaced along a straight line. He would then expect the system to generate exactly the straight line along which the points lay. Any oscillations around this line, any deviations from linearity, would be unacceptable; by selecting his points along a straight line, he is specifically requesting that the curve not wiggle.

Let

$$P(t) = \sum B_k^n(t)P_k, \qquad 0 \leqq t \leqq 1$$

be a well-defined curve, and let

$$L(t) = At + B, \qquad 0 \leqq t \leqq 1$$

be a straight line. We can select $n+1$ equally spaced points, $P_0, \cdots, P_n$, along the line

$L(t)$ by setting

$$P_k = L\left(\frac{k}{n}\right) = A\left(\frac{k}{n}\right) + B.$$

The curve $P(t)$ exactly reproduces the line $L(t)$ iff

$$P(t) = L(t), \qquad 0 \leqq t \leqq 1,$$

for this particular choice of control points $P_0, \cdots, P_n$. In this case

$$At + B = L(t) = P(t) = \sum B_k^n(t) P_k = \sum B_k^n(t)\left[A\left(\frac{k}{n}\right) + B\right]$$

$$= \frac{A}{n}\sum kB_k^n(t) + B\sum B_k^n(t) = \frac{A}{n}\sum kB_k^n(t) + B.$$

In general this can be true iff

$$\sum kB_k^n(t) = nt.$$

Thus we have shown that

$$P(t) \text{ exactly reproduces points} \Leftrightarrow \sum B_k^n(t) = 1,$$

$$P(t) \text{ exactly reproduces lines} \Leftrightarrow \sum kB_k^n(t) = nt.$$

2.9. Nondegenerate. A well-defined curve

$$P(t) = \sum B_k^n(t) P_k$$

is said to be nondegenerate iff

$$P(t) = P_0 \text{ for all } t \Rightarrow P_k = P_0 \text{ for all } k.$$

That is, a curve is said to be nondegenerate iff the only time it collapses to a single point is when all the control points are located at that same point.

THEOREM 2.1. *Let $P(t) = \sum B_k^n(t) P_k$ be a well-defined curve. Then $P(t)$ is nondegenerate iff the blending functions $\{B_k^n(t)\}$ are linearly independent.*

Proof. Suppose that the blending functions are linearly independent. If

$$\sum B_k^n(t) P_k = P_0$$

then since the curve is well defined

$$\sum B_k^n(t) P_k = \sum B_k^n(t) P_0, \qquad \sum B_k^n(t)(P_k - P_0) = 0.$$

Therefore for every vector v,

$$\sum B_k^n(t)[(P_k - P_0) \cdot v] = 0.$$

Now since the blending functions $\{B_k^n(t)\}$ are linearly independent, we must have

$$(P_k - P_0) \cdot v = 0$$

for every vector v and every index k. Hence

$$P_k = P_0 \quad \text{for every index } k.$$

Thus, the curve $P(t)$ is nondegenerate.

Conversely, suppose that the curve is nondegenerate, and that

$$\sum c_k B_k^n(t) = 0.$$

Let v be any nonzero vector, and let

$$P_k = P_0 + c_k v.$$

Then,

$$P(t) = \sum B_k^n(t) P_k = \sum B_k^n(t)(P_0 + c_k v) = [\sum B_k^n(t)] P_0 + [\sum c_k B_k^n(t)] v = P_0.$$

Hence, since $P(t)$ is nondegenerate, it follows that

$$P_k = P_0, \quad \text{for all } k.$$

Therefore,

$$c_k = 0 \quad \text{for all } k.$$

Thus the functions $\{B_k^n(t)\}$ are linearly independent. □

Thus we have shown that

$$P(t) \text{ is nondegenerate} \Leftrightarrow \{B_k^n(t)\} \text{ are linearly independent.}$$

It is important for computer aided geometric design that the curves be nondegenerate. After all, a curve which collapses unexpectedly to a single point is not of much use to a designer. Also, we wish to avoid burning holes in the screen with very bright spots caused by degenerate curves.

2.10. Subdivision algorithm. Consider again a curve

$$P(t) = B[P_0, \cdots, P_n](t) = \sum B_k^n(t) P_k, \qquad 0 \leqq t \leqq 1$$

and fix 2 points $P(a)$, $P(b)$ along $P(t)$. A subdivision algorithm is a technique for constructing a sequence of points $Q_0, \cdots, Q_n$ such that if

$$Q(t) = B[Q_0, \cdots, Q_n](t) = \sum B_k^n(t) Q_k, \qquad 0 \leqq t \leqq 1,$$

then

$$Q(t) \subseteq P(t), \quad 0 \leqq t \leqq 1, \quad Q(0) = P(a), \quad Q(1) = P(b).$$

Subdivision algorithms are important in computer aided geometric design for many reasons. They enable us to trim curves. They allow us to apply formulas initially developed only for the end points of a curve, where the parameter is 0 or 1, at arbitrary locations along the curve. Thus they help simplify the computations of tangent, curvature and torsion. When combined with the convex hull property, they lead to accurate, iterative, intersection routines [9].

If the functions $\{B_k^n(t)\}$ are nth degree polynomials in t, then the linear independence of the blending functions implies that they form a basis for all nth degree polynomials in t. In particular since the functions $\{B_k^n(rt)\}$ are also nth degree polynomials in t, there are constants $\{B_{ik}^n(r)\}$ such that

$$B_i^n(rt) = \sum_k B_{ik}^n(r) B_k^n(t).$$

Let

$$Q_k(r) = \sum_i B_{ik}^n(r) P_i.$$

PROPOSITION 2.1. $B[Q_0(r), \cdots, Q_n(r)](t) = B[P_0, \cdots, P_n](r$
Proof.

$$B[Q_0(r), \cdots, Q_n(r)](t) = \sum_k B_k^n(t) Q_k(r) = \sum_k B_k^n(t) \sum_i B_{ik}^n(r) P_i$$
$$= \sum_i [\sum_k B_{ik}^n(r) B_k^n(t)] P_i = \sum_i B_i^n(rt) P_i = B[P_0, \cdots, P_n](rt). \qquad \square$$

Thus the points $Q_0(r), \cdots, Q_n(r)$ subdivide the curve $P(t)$ from $P(0)$ to $P(r)$. Using symmetry, we can also subdivide the curve $P(t)$ from $P(r)$ to $P(1)$. By applying these two subdivision algorithms one after the other, we can subdivide the curve $P(t)$ between any two points $P(a)$ and $P(b)$. Hence we have shown that

polynomial basis⇒subdivision algorithm.

To actually subdivide the curve $P(t)$, we need explicit expressions for the constants $B_{ik}^n(r)$. It is often easier to prove the existence of such constants than to actually compute them. However, we shall show in § 3 that for the Bernstein polynomials

$$B_{ik}^n(r) = B_i^k(r), \quad B_i^n(rt) = \sum_k B_i^k(r) B_k^n(t), \quad Q_k(r) = \sum_i B_i^k(r) P_i = B[P_0, \cdots, P_k](r).$$

2.11. Augmentation algorithm. Suppose that for each integer n we have a collection of blending functions $\{B_k^n(t)\}$. Given a curve

$$P(t) = B[P_0, \cdots, P_n](t) = \sum B_k^n(t) P_k$$

an augmentation algorithm is a technique for finding new control points $Q_0, \cdots, Q_{n+1}$ such that

$$B[Q_0, \cdots, Q_{n+1}](t) = B[P_0, \cdots, P_n](t).$$

Thus an augmentation algorithm is a technique for representing the same exact curve with one additional control point. Augmentation algorithms are useful in computer aided geometric design because the additional control points they generate allow us greater flexibility in determining the final shape of our curves.

If the functions $\{B_k^n(t)\}$ are a polynomial basis for each n, then we can write the nth degree polynomials $\{B_k^n(t)\}$ in terms of the $(n+1)$st degree polynomials $\{B_k^{n+1}(t)\}$. That is, there must be constants $\{A_{ik}^n\}$ such that

$$B_k^n(t) = \sum_i A_{ik}^n B_i^{n+1}(t).$$

Let

$$Q_i = \sum_k A_{ik}^n P_k, \qquad 0 \leqq i \leqq n+1.$$

PROPOSITION 2.2. $B[Q_0, \cdots, Q_{n+1}](t) = B[P_0, \cdots, P_n](t)$.
Proof.

$$B[Q_0, \cdots, Q_{n+1}](t) = \sum_i B_i^{n+1}(t) Q_i = \sum_i B_i^{n+1}(t) \sum_k A_{ik}^n P_k$$
$$= \sum_k [\sum_i A_{ik}^n B_i^{n+1}(t)] P_k = \sum_k B_k^n(t) P_k = B[P_0, \cdots, P_n](t). \qquad \square$$

Hence we have shown that

polynomial basis⇒augmentation algorithm.

To actually augment a curve $P(t)$, we need to know the values of the constants A_{ik}. As with subdivision, it is often easier to prove the existence of these constants than to actually compute them. Nevertheless in § 3 we shall compute these constants for a whole family of blending functions.

2.12. Variation diminishing. A curve

$$P(t)=\sum B_k^n(t)P_k, \qquad 0\leqq t\leqq 1$$

is said to be variation diminishing iff for every collection of points $P_0,\cdots,P_n$ and every plane R, the number of times $P(t)$ crosses R is less than or equal to the number of times the polygon determined by the ordered vertices $P_0,\cdots,P_n$ crosses R. Thus, intuitively a curve is variation diminishing iff it does not oscillate any more than the chords which connect its control points. Clearly to have any hope of being variation diminishing, a curve must be nondegenerate and lie in the convex hull of its control points.

Nondegeneracy and the convex hull property are necessary but not sufficient conditions. To obtain a sufficient condition, we will appeal to the following rule.

Descartes' law of signs. A collection of functions $\{B_k^n(t)\}$ is said to satisfy Descartes' law of signs in the interval (a, b) iff for every collection of constants $(c_0,\cdots,c_n)$

$$\text{zeros in } (a,b)\,[\textstyle\sum c_kB_k^n(t)]\leqq \text{sign alternations of } (c_0,\cdots,c_n).$$

It is well known that the power functions $\{t^k\}$ satisfy Descartes' law of signs in the interval $(0,\infty)$ [12]. Using this fact, it is easy to prove that the Bernstein polynomials $\binom{n}{k}t^k(1-t)^{n-k}$ satisfy Descartes' law of signs in the interval $(0,1)$ [10] (see Theorem A.2).

THEOREM 2.2. *Let* $P(t)=\sum B_k^n(t)P_k$, $0\leqq t\leqq 1$ *be a well-defined curve. If the blending functions* $\{B_k^n(t)\}$ *satisfy Descartes' law of signs in the interval* $(0,1)$, *then the curve* $P(t)$ *is variation diminishing.*

Proof. Let R be a plane, Q a point on R and N a vector normal to R. Then a point P lies on the plane R iff

$$(Q-P)\cdot N=0$$

and 2 points P_i, P_j lie on opposite sides of the plane R iff

$$\text{sign}\,[(Q-P_i)\cdot N]=-\text{sign}\,[(Q-P_j)\cdot N].$$

Let

$I=$ number of times $P(t)$ crosses R,

$J=$ number of times the polygon determined by $P_0,\cdots,P_n$ crosses R.

We must show that

$$I\leqq J.$$

Now since $P(t)$ is well-defined

$$Q-P(t)=Q-\sum B_k^n(t)P_k=\sum B_k^n(t)Q-\sum B_k^n(t)P_k=\sum B_k^n(t)(Q-P_k).$$

Therefore by Descartes' law of signs

$$I\leqq \text{zeros in } (0,1)[(Q-P(t))\cdot N]=\text{zeros in } (0,1)\,[\textstyle\sum B_k^n(t)(Q-P_k)\cdot N]$$
$$\leqq \text{sign alternations of } [(Q-P_0)\cdot N,\cdots,(Q-P_n)\cdot N]=J. \qquad \square$$

We have shown that

$$\text{Descartes' law of signs}\Rightarrow\text{variation diminishing property.}$$

Notice too that

$$\text{Descartes' law of signs}\Rightarrow\text{linearly independent blending functions}$$
$$\Rightarrow\text{nondegenerate curves.}$$

In computer aided geometric design curves must not oscillate too much; designers must be able to control the wiggle. Therefore the variation diminishing property is critical. Thus even though Lagrange polynomials exactly interpolate position, they have proved to be inappropriate for computer aided geometric design. They tend to oscillate uncontrollably precisely because they are not variation diminishing. On the other hand, even though Bezier curves do not faithfully interpolate position, they have proved to be quite useful for computer aided geometric design. Bezier curves give an accurate representation of a designer's intent because they are variation diminishing.

2.13. Local control. Given a curve

$$P(t) = \sum B_k^n(t) P_k, \qquad 0 \leqq t \leqq 1,$$

we are said to have local control iff changing any one control point P_k has only a local effect on the shape of $P(t)$. We can have local control iff the support of each blending function is only some fraction of the total domain of t. Thus

$$\text{we have local control} \Leftrightarrow B_k^n(t) \text{ has local support.}$$

Local control is important in computer aided geometric design because it allows a designer to alter a segment with which he is dissatisfied without ruining the shape of the remainder of the curve. It is for this reason that B-splines have become increasingly popular in computer aided geometric design.

For curves whose blending functions do not have local support, we can use subdivision algorithms to isolate unsatisfactory segments into separate curves. We can then alter these segments without affecting the remainder of the curve. This gives us a measure of local control at the cost of the loss of some derivatives at the joints.

For Bezier curves it is easy to show that [6]

$$P^{(j)}(0) = \frac{n!}{(n-j)!} \sum_{k=0}^{j} (-1)^{j-k} \binom{j}{k} P_k, \qquad P^{(j)}(1) = \frac{n!}{(n-j)!} \sum_{k=0}^{j} (-1)^{j-k} \binom{j}{k} P_{n-j+k}.$$

Thus for Bezier curves the jth derivative at each end point depends only on the adjacent j control points. This additional fact allows us to predict exactly which derivatives at the joints will be affected by moving any particular control point. For example, the general formulas for curvature and torsion are [13]

$$K(t) = \frac{|P'(t) \times P''(t)|}{|P'(t)|^3}, \qquad T(t) = \frac{P'(t) \cdot [P''(t) \times P'''(t)]}{|P'(t) \times P''(t)|^2}.$$

Therefore at the end points of a Bezier curve

$$K(t) = \begin{cases} \dfrac{(n-1)}{n} \dfrac{|(P_1 - P_0) \times (P_2 - P_1)|}{|P_1 - P_0|^3}, & t = 0, \\ \dfrac{(n-1)}{n} \dfrac{|(P_{n-1} - P_n) \times (P_{n-2} - P_{n-1})|}{|P_{n-1} - P_n|^3}, & t = 1, \end{cases}$$

$$T(t) = \begin{cases} \dfrac{(n-2)}{n} \dfrac{(P_1 - P_0) \cdot [(P_2 - P_1) \times (P_3 - P_2)]}{|(P_1 - P_0) \times (P_2 - P_1)|^2}, & t = 0, \\ \dfrac{(n-2)}{n} \dfrac{(P_{n-1} - P_n) \cdot [(P_{n-2} - P_{n-1}) \times (P_{n-3} - P_{n-2})]}{|(P_{n-1} - P_n) \times (P_{n-2} - P_{n-1})|^2}, & t = 1. \end{cases}$$

Thus only the first 2 adjacent control points have any effect on the curvature, and only the first 3 adjacent control points have any effect on the torsion, at the end points. Hence moving any other control point will have no effect on these critical values at the joints.

2.14. Summary. We summarize our results in Table 1.

TABLE 1

	Curve		*Blending functions*
1.	well-defined	$\Leftrightarrow$	$\sum B_k^n(t) = 1$
2.	convex hull	$\Leftrightarrow$	$B_k^n(t) \geqq 0$
3.	smooth	$\Leftrightarrow$	$B_k^n(t)$ differentiable
4.	interpolates end points	$\Leftrightarrow$	$B_k^n(0) = \begin{cases} 0, & k \neq 0, \\ 1, & k = 0, \end{cases}$ $B_k^n(1) = \begin{cases} 0, & k \neq n, \\ 1, & k = n. \end{cases}$
5.	extends to surfaces a. rectangular b. triangular	 $\Leftrightarrow$ $\Leftrightarrow$	 same as 4 $B_{ij}^n(0, v) = \begin{cases} 0, & i \neq 0, \\ B_j^n(v), & i = 0, \end{cases}$ $B_{ij}^n(u, 0) = \begin{cases} 0, & j \neq 0, \\ B_i^n(u), & j = 0, \end{cases}$ $B_{ij}^n(u, 1-u) = \begin{cases} 0, & i+j \neq n, \\ B_i^n(u), & i+j = n, \end{cases}$
6.	symmetry	$\Leftrightarrow$	$B_k^n(t) = B_{n-k}^n(1-t)$
7.	geometric construction algorithm	$\Leftarrow$	recursion formula
8.	exactly reproduces straight lines	$\Leftrightarrow$	$\sum k B_k^n(t) = nt$
9.	nondegenerate	$\Leftrightarrow$	linear independence
10.	subdivision algorithm	$\Leftarrow$	polynomial basis
11.	augmentation algorithm	$\Leftarrow$	polynomial basis
12.	variation diminishing	$\Leftarrow$	Descartes' law of signs
13.	local control	$\Leftrightarrow$	local support

The conditions on the blending functions in Table 1 are not all independent. For example, it is easy to show that

conditions 1, 2, 8$\Rightarrow$condition 4, condition 12$\Rightarrow$condition 9.

Notice too that not all the implications go in both directions. For example, if the blending functions are a polynomial basis, then the curve necessarily has a subdivision algorithm. However it may be that there are other conditions which could imply that a curve has a subdivision algorithm even if its blending functions are not a polynomial basis.

Finally, the reader should recognize that everything we have said in this section about curves has a analogue for surfaces.

3. Polya's urn model. The question remains: where can we find blending functions with all, or even just a few, of the properties listed in the preceding section? If we focus our attention on the first two properties

$$\sum B_k^n(t) = 1, \quad 0 \leqq t \leqq 1, \qquad B_k^n(t) \geqq 0, \quad 0 \leqq t \leqq 1,$$

they strike a familiar chord; these are just the defining characteristics of discrete probability distributions. Therefore to find appropriate blending functions for computer aided geometric design, we can look to classical discrete probability theory. Indeed we already know that the blending functions for the Bezier curves are the Bernstein polynomials and these polynomials represent the binomial distribution.

There are many classical discrete probability distributions which we could consider. However here we shall restrict our attention to a particularly propitious collection of distributions which arise from an urn model first introduced by G. Polya [3].

Polya's urn. Consider an urn initially containing w white balls and b black balls. One ball at a time is drawn at random from the urn and its color inspected. It is then returned to the urn and a constant number c of balls of the same color are added to the urn.

Let

$$t = \frac{w}{w+b} = \text{initial probability of drawing a white ball,}$$

$$a = \frac{c}{w+b} = \text{initial percentage of balls added to the urn.}$$

If we hold a constant and allow t to vary, we obtain a discrete probability distribution

$$D_k^n(t) = \text{probability of drawing exactly } k \text{ white balls in the first } n \text{ trials.}$$

Notice that we get a different probability distribution $D_k^n(t)$ for each distinct value of a (see below). In particular, if $a = 0$, then $D_k^n(t)$ is just the binomial distribution (sampling with replacement). From here on we shall assume that a is a fixed constant.

The functions $D_k^n(t)$ have many properties which are desirable for computer aided geometric design. Since they represent a probability distribution, it follows immediately that

$$\sum D_k^n(t) = 1, \quad 0 \leqq t \leqq 1, \qquad D_k^n(t) \geqq 0, \quad 0 \leqq t \leqq 1.$$

In addition, if initially there are no white balls in the urn, then we will never add any white balls to the urn. Similarly, if initially there are only white balls in the urn, then the urn will always contain only white balls. Therefore

$$D_k^n(0) = \begin{cases} 0, & k \neq 0, \\ 1, & k = 0, \end{cases} \qquad D_k^n(1) = \begin{cases} 0, & k \neq n, \\ 1, & k = n. \end{cases}$$

Also since white balls and black balls are treated identically, this urn model is symmetric with respect to white and black. Therefore

probability of drawing exactly k white balls in the first n trials	=	probability of drawing exactly $n-k$ black balls in the first n trials

so

$$D_k^n(t) = D_{n-k}^n(1-t).$$

By considering 2 Polya urns each with 2 colors, we get the 2-dimensional distribution

$$D_{jk}^{mn}(s, t) = \text{probability of drawing exactly } j \text{ white balls in the first } m \text{ trials from urn 1 and exactly } k \text{ white balls in the first } n \text{ trials from urn 2.}$$

This distribution is just the product of the distribution for each individual urn. That is,

$$D_{jk}^{mn}(s, t) = D_j^m(s) D_k^n(t).$$

This urn model can also be extended to urns containing balls of many different colors. After each pick, we still just return the selected ball along with c new balls of the same color. Consider an urn which initially holds r red, w white, and b black balls, and let

$$u = \frac{r}{r+w+b} = \text{initial probability of drawing a red ball,}$$

$$v = \frac{w}{r+w+b} = \text{initial probability of drawing a white ball,}$$

$$a = \frac{c}{r+w+b} = \text{initial perentage of balls added to the urn.}$$

Again if we hold a constant and allow u, v to vary, we obtain a discrete probability distribution in two variables

$$D_{ij}^n(u, v) = \text{probability of drawing exactly } i \text{ red balls and } j \text{ white balls in the first } n \text{ trials.}$$

Therefore

$$\sum_{i,j} D_{ij}^n(u, v) = 1, \quad 0 \leqq u+v \leqq 1, \qquad D_{ij}^n(u, v) \geqq 0, \quad 0 \leqq u+v \leqq 1.$$

Moreover if initially one color is absent from the urn, then this urn behaves exactly like an urn with balls of only two colors. Therefore

$$D_{ij}^n(0, v) = \begin{cases} 0, & i \neq 0, \\ D_j^n(v), & i = 0, \end{cases} \qquad D_{ij}^n(u, 0) = \begin{cases} 0, & j \neq 0, \\ D_i^n(u), & j = 0, \end{cases}$$

$$D_{ij}^n(u, 1-u) = \begin{cases} 0, & i+j \neq n, \\ D_i^n(u), & i+j = n. \end{cases}$$

There is a recursion formula relating $D_k^{n+1}(t)$ to $D_k^n(t)$ and $D_{k-1}^n(t)$. Indeed let

$$f_k^n(t) = \text{probability of failing to draw a white ball after drawing exactly } k \text{ white balls in the first } n \text{ trials,}$$

$$s_k^n(t) = \text{probability of succeeding to draw a white ball after drawing exactly } k \text{ white balls in the first } n \text{ trials.}$$

Then certainly

$$f_k^n(t) + s_k^n(t) = 1.$$

Moreover

$$\begin{aligned} D_k^{n+1}(t) &= \text{probability of drawing exactly } k \text{ white balls in the first } n+1 \text{ trials} \\ &= (\text{probability of drawing exactly } k \text{ white balls in the first } n \text{ trials}) \\ &\quad \times (\text{probability of failing to draw a white ball on the next trial}) \\ &\quad + (\text{probability of drawing exactly } k-1 \text{ white balls in the first } n \text{ trials}) \\ &\quad \times (\text{probability of succeeding to draw a white ball on the next trial}) \\ &= D_k^n(t) f_k^n(t) + D_{k-1}^n(t) s_{k-1}^n(t) \end{aligned}$$

so

$$D_k^{n+1}(t) = f_k^n(t) D_k^n(t) + s_{k-1}^n(t) D_{k-1}^n(t), \qquad f_k^n(t) + s_k^n(t) = 1.$$

We can compute $f_k^n(t)$, $s_k^n(t)$ explicitly. After exactly k successes in the first n trials there are $w+kc$ white balls and $w+b+nc$ total balls in the urn. Therefore

$$s_k^n(t) = \frac{\text{number of white balls}}{\text{total number of balls}} = \frac{w+kc}{w+b+nc}.$$

Dividing numerator and denominator by $w+b$, we get

$$s_k^n(t) = \frac{t+ka}{1+na}, \qquad f_k^n(t) = 1 - s_k^n(t) = \frac{(1-t)+(n-k)a}{1+na}.$$

Since by definition

$$D_0^1(t) = 1-t, \qquad D_1^1(t) = t,$$

it follows by induction on n that $D_k^n(t)$ depends only on a and on t. In particular for the binomial distribution

$$a = 0, \quad s_k^n(t) = t, \quad f_k^n(t) = 1-t, \quad B_k^{n+1}(t) = (1-t)B_k^n(t) + tB_{k-1}^n(t),$$

which is the standard recursion formula for the Bernstein polynomials.

Since $s_k^n(t)$, $f_k^n(t)$, $D_0^1(t)$, $D_1^1(t)$ are all first degree polynomials in t, it again follows easily by induction on n that $D_k^n(t)$ is an nth degree polynomial in t. Therefore certainly

$$D_k^n(t) \text{ is infinitely differentiable.}$$

We can even derive an explicit expression for $D_k^n(t)$. There are $\binom{n}{k}$ ways of selecting exactly k white balls in the first n trials. To compute the probability of just one such way, we must multiply together k success factors of type $s_i^L(t)$ and $n-k$ failure factors of type $f_j^L(t)$ where for each L either $s_i^L(t)$ or $f_j^L(t)$ appears but not both. Now the denominators of $s_i^L(t)$, $f_j^L(t)$ are identical. Moreover i must take on the values $0, \cdots, k-1$; $L-j$ must take on the values $0, \cdots, n-k-1$; and L must take on the values $0, \cdots, n-1$. Therefore

$$D_k^n(t) = \binom{n}{k} \frac{t \cdots [t+(k-1)a](1-t) \cdots [(1-t)+(n-k-1)a]}{(1+a) \cdots [1+(n-1)a]}.$$

(For further details see [2].) When $a=0$, this formula reduces to the binomial distribution

$$B_k^n(t)=\binom{n}{k}t^k(1-t)^{n-k}.$$

In our derivation of the explicit formula for $D_k^n(t)$ we observed that the probabilities of any 2 distinct ways of selecting exactly k white balls in the first n trials are identical. This critical observation has several important consequences. Let

$S_n(t)$ = a priori probability of selecting a white ball on the nth trial,

$E_n(t)$ = the expected number of white balls selected in the first n trials.

Then it is obvious from probabilistic considerations that

$$S_n(t)=\sum s_k^{n-1}(t)D_k^{n-1}(t),\quad E_n(t)=\sum kD_k^n(t),\quad E_n(t)=\sum_{k=1}^{n} S_k(t).$$

The first two formulas are just weighted averages, and the third formula just says that the expectation is the sum of the a priori probabilities.

PROPOSITION 3.1. $S_n(t)=t$, $n\geqq 1$.

Proof. We shall use a simple counting argument. Let

$A_k^n(t)$ = probability of selecting exactly k white balls in the next n trials after selecting a white ball on the first trial.

Since the probabilities of any 2 distinct ways of selecting exactly k white balls in the first n trials are identical, it follows that

$s_k^n(t)D_k^n(t)$ = probability of selecting exactly k white balls in the first n trials and then selecting a white ball on the $(n+1)$st trial

$$=\binom{n}{k}$$ probability of selecting, in one particular way, exactly $k+1$ white balls in the first $n+1$ trials

= probability of selecting a white ball on the first trial and then selecting exactly k white balls in the next n trials

$$=tA_k^n(t).$$

Therefore,

$$S_{n+1}(t)=\sum_k s_k^n(t)D_k^n(t)=t\sum_k A_k^n(t)=t.$$ □

COROLLARY 3.1. $E_n(t)=nt$.

COROLLARY 3.2. $\sum kD_k^n(t)=nt$.

Another consequence of the critical observation about the Polya distribution is that it allows us to derive an explicit formula for $D_k^n(t)$ in terms of $D_k^{n+1}(t)$ and $D_{k+1}^{n+1}(t)$.

LEMMA 3.1.

$$f_k^n(t)D_k^n(t) = \frac{(n+1-k)}{(n+1)} D_k^{n+1}(t).$$

Proof. As above

$f_k^n(t)D_k^n(t)$ = probability of selecting exactly k white balls in the first n trials and then selecting a black ball on the next trial.

Now there are a total of $\binom{n}{k}$ ways of selecting exactly k white balls in n trials, and a total of $\binom{n+1}{k}$ ways of selecting exactly k white balls in $n+1$ trials. Therefore since each distinct way has exactly the same probability of occurring, it follows that

$$f_k^n(t)D_k^n(t) = \frac{\binom{n}{k}}{\binom{n+1}{k}} D_k^{n+1}(t) = \frac{(n+1-k)}{(n+1)} D_k^{n+1}(t). \qquad \square$$

LEMMA 3.2.

$$s_k^n(t)D_k^n(t) = \frac{(k+1)}{(n+1)} D_{k+1}^{n+1}(t).$$

Proof. Again

$s_k^n(t)D_k^n(t)$ = probability of selecting exactly k white balls in the first n trials and then selecting a white ball on the next trial.

Now there are $\binom{n}{k}$ ways of selecting exactly k white balls in n trials, and $\binom{n+1}{k+1}$ ways of selecting exactly $k+1$ white balls in $n+1$ trials. Therefore, since each distinct way has exactly the same probability of occurring, it follows that

$$s_k^n(t)D_k^n(t) = \frac{\binom{n}{k}}{\binom{n+1}{k+1}} D_{k+1}^{n+1}(t) = \frac{(k+1)}{(n+1)} D_{k+1}^{n+1}(t). \qquad \square$$

COROLLARY 3.3.

$$D_k^n(t) = \frac{(n+1-k)}{(n+1)} D_k^n(t) + \frac{(k+1)}{(n+1)} D_{k+1}^{n+1}(t).$$

Proof. This result follows immediately by simple addition from Lemmas 3.1, 3.2. $\square$

The coefficients in the formula of the preceding corollary do not depend on the value of a. This means that the formula for raising the degree of the Polya distributions is identical to the formula for the binomial distribution. Geometrically this means that the augmentation algorithm for the Polya curves is identical to the augmentation algorithm for Bezier curves. Specifically if

$$Q_k = \frac{k}{(n+1)} P_{k-1} + \frac{(n+1-k)}{(n+1)} P_k$$

then for any Polya distribution the curve with control points $Q_0, \cdots, Q_{n+1}$ is identical to the curve with control points $P_0, \cdots, P_n$.

Because of their similarity to the Bernstein polynomials, the polynomials $D_0^n(t), \cdots, D_n^n(t)$ satisfy Descartes' law of signs in the interval (0, 1). However, since our proof of this result is not based on probability theory, we shall defer it to the Appendix. From the fact that these polynomials satisfy Descartes' law of signs, we also conclude that they are linearly independent and that they form a polynomial basis for all nth degree polynomials in t.

We summarize our results in Table 2.

TABLE 2

	Urn		*Formula*
1.	probability distribution	$\Rightarrow$	$\sum D_k^n(t) = 1, \quad 0 \leqq t \leqq 1,$ $D_k^n(t) \geqq 0, \quad 0 \leqq t \leqq 1$
2.	adding balls only of the selected color	$\Rightarrow$	$D_k^n(0) = \begin{cases} 0, & k \neq 0, \\ 1, & k = 0, \end{cases}$ $D_k^n(1) = \begin{cases} 0, & k \neq n, \\ 1, & k = n, \end{cases}$
3.	symmetry between white and black	$\Rightarrow$	$D_k^n(t) = D_{n-k}^n(1-t)$
4.	extensions to multiple urns	$\Rightarrow$	$D_{jk}^{mn}(s, t) = D_j^m(s) D_k^n(t)$
5.	extensions to urns with multiple colors	$\Rightarrow$	$D_{ij}^n(0, v) = \begin{cases} 0, & i \neq 0, \\ D_j^n(v), & i = 0, \end{cases}$ $D_{ij}^n(u, 0) = \begin{cases} 0, & j \neq 0, \\ D_i^n(u), & j = 0, \end{cases}$ $D_{ij}^n(u, 1-u) = \begin{cases} 0, & i+j \neq n, \\ D_i^n(u), & i+j = n. \end{cases}$
6.	relationship between first n and first $n+1$ picks (recursion)	$\Rightarrow$	$D_k^{n+1}(t) = f_k^n(t) D_k^n(t) + s_{k-1}^n(t) D_{k-1}^n(t),$ $f_k^n(t) + s_k^n(t) = 1$
7.	polynomial function	$\Rightarrow$	$D_k^n(t)$ infinitely differentiable
8.	expectation	$\Rightarrow$	$\sum k D_k^n(t) = nt$
9.	raising degree	$\Rightarrow$	$D_k^n(t) = [(n+1-k)/(n+1)] D_k^{n+1}(t)$ $+ [(k+1)/(n+1)] D_{k+1}^{n+1}(t)$
10.	similarity to Bernstein polynomials	$\Rightarrow$	$\{D_k^n(t)\}$ satisfy Descartes' law of signs in the interval (0, 1), $\{D_k^n(t)\}$ are a polynomial basis

Comparing Table 2 with Table 1, we see immediately that a curve

$$P(t) = \sum D_k^n(t) P_k,$$

which uses one of Polya's urn distributions $\{D_k^n(t)\}$ for its blending functions will automatically have all of the following geometric properties: 1. well-defined, 2. convex

hull, 3. smooth, 4. interpolates end points, 5. extends to surfaces (a. rectangular, b. triangular), 6. symmetry, 7. geometric construction algorithm, 8. exactly reproduces points and lines, 9. nondegenerate, 10. subdivision algorithm, 11. augmentation algorithm and 12. variation diminishing.

Missing are only local control and an explicit subdivision algorithm.

We cannot hope for local control in the sense of § 2.13 since the functions $D_k^n(t)$ do not have local support; even the classical Bezier curves fail to allow this kind of local control. However for curves which admit a subdivision algorithm, we can achieve a degree of local control by using the subdivision algorithm to isolate unsatisfactory segments. Changes to control points will then have only a local effect though we may lose some derivatives at the joints.

Since the functions $\{D_k^n(t)\}$ form a polynomial basis, we know that there always exist constants $\{D_{ik}^n(r)\}$ such that

$$D_i^n(rt) = \sum_k D_{ik}^n(r) D_k^n(t).$$

However to actually subdivide a specific curve, we need explicit formulas for the constants $\{D_{ik}^n(r)\}$. We shall now show that, for the binomial distribution,

$$B_{ik}^n(r) = B_i^k(r).$$

PROPOSITION 3.2. *The binomial distribution satisfies the identity*

$$B_i^n(rt) = \sum_k B_i^k(r) B_k^n(t).$$

Proof. This proof is based on Polya's urn model ($a = 0$) of the binomial distribution. Consider two binomial urns: one with red and blue balls, the other with white and black balls. Let

r = probability of selecting a red ball from urn 1,

t = probability of selecting a white ball from urn 2.

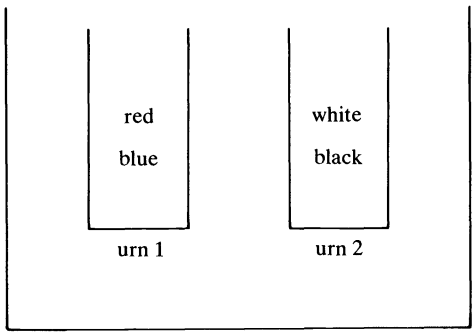


FIG. 4. *Super urn.*

Place these two urns into a super urn. A selection from the super urn consists of selecting one ball from each regular urn, inspecting the colors, and replacing the balls (see Fig. 4). The super urn is also a binomial urn and

rt = probability of selecting a red-white combination

Therefore,

$$\begin{aligned} B_i^n(rt) &= \text{probability of selecting exactly } i \text{ red-white combinations in } n \text{ trials} \\ &= \sum_k (\text{probability of selecting exactly } k \text{ white balls in } n \text{ trials}) \\ &\quad \times (\text{probability of selecting exactly } i \text{ red balls during the } k \text{ trials where the white balls were chosen}) \\ &= \sum_k B_i^k(r) B_k^n(t). \end{aligned}$$

□

When the small urns model the binomial distribution, then so does the large urn since both small and large urns employ sampling with replacement. However when the small urns model some other Polya distribution $\{D_k^n(t)\}$ with $a \neq 0$, then the large urn will no longer model this same distribution since the addition of new balls into the two small urns has a very different effect on the composition of the super urn. Therefore this identity is not generally valid for arbitrary Polya distributions. Indeed, for arbitrary Polya distributions, we do not yet have explicit expressions for $D_{ik}^n(r)$.

For Bezier curves the subdivision algorithm is intimately related to the geometric construction algorithm which generates the points $\{P_k^L(r)\}$. Indeed it is the points $\{P_0^k(r)\}$ which actually subdivide the Bezier curve at $P(r)$ [7]. Since arbitrary Polya distributions also give rise to a geometric construction algorithm, it may be that the points $\{P_0^k(r)\}$ also subdivide these curves at $P(r)$. As yet this is still an open question.

The Polya distribution has a free constant a. By varying this free constant, we can alter the shape of our curves without moving our control points. We would like to understand the geometric impact of increasing the value of a. Consider then what happens in the limit when a is actually infinite. In this case after the first pick, we must add an infinite number of balls of the selected color to the urn. Therefore, with probability 1, all the balls selected after the first trial will be of the same color as the ball selected on the first trial. Hence

$$\lim_{a\to\infty} D_k^n(t) = \begin{cases} D_0^1(t) = 1-t, & k = 0, \\ 0, & k \neq 0, n, \\ D_1^1(t) = t, & k = n. \end{cases}$$

Thus,

$$\lim_{a\to\infty} P(t) = (1-t)P_0 + tP_n.$$

Therefore the effect of increasing a from 0 to ∞ is simply to flatten a Bezier curve into a straight line (see Fig. 5).

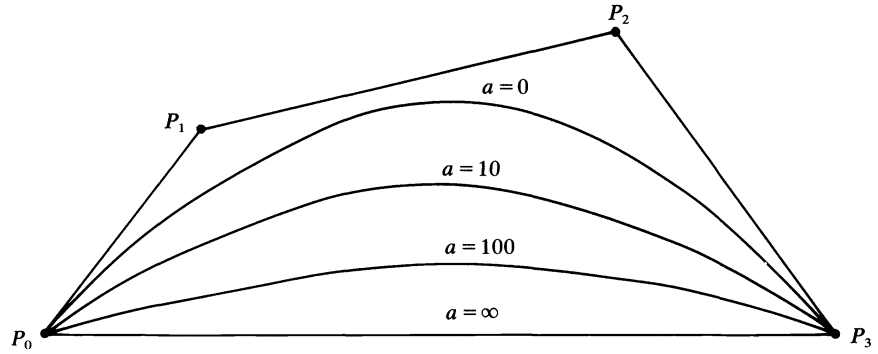


FIG. 5. *Polya curves for different values of* a $(n = 3)$.

4. Other probabilistic models. The main theme of this paper is that the blending functions of computer aided geometric design are discrete probability distributions. Therefore if we know what properties we wish to build into our curves, we can construct probabilistic models to generate the required blending functions.

Conversely, starting with a discrete probability distribution, we can study the geometric properties of the curves that it generates. If these properties are sufficiently interesting, then eventually applications may be found for these curves in computer aided geometric design. For example, we could begin with the following variation of Polya's urn model.

Generalized Polya's urn. Consider an urn containing w white balls and b black balls. One ball at a time is drawn at random from the urn and its color inspected. It is then returned to the urn and c_1 balls of the same color and c_2 balls of the opposite color are added to the urn.

Again this urn model gives rise to a collection of probability distributions $D_k^n(t)$. However while some properties of the original Polya urns, like the symmetry between white and black, are retained, other properties, like adding only balls of the selected color, are abandoned. This change implies that the corresponding curves will no longer pass through the designer's end points. If, for some reason, the user wishes to relax this end point condition, then this model may generate just the blending functions he requires.

Or consider the classical Poisson model.

Poisson model. Certain events occur at random times. Their occurrences are such that:

1. The number of events occurring in two disjoint time intervals is independent.
2. There is a fixed constant a such that when Δt is small, the probability of one event occurring in time Δt is approximately $a\Delta t$.
3. The probability of more than one event occurring in time Δt is negligible when Δt is small.

For each fixed value of a, the Poisson model gives rise to a probability distibution

$$D_k(t) = \text{probability of } k \text{ events occurring in the time interval } (0, t).$$

It is a well-known fact [1] that, explicitly,

$$D_k(t) = e^{-at}\frac{(at)^k}{k!}, \qquad t \geqq 0.$$

Moreover it is easy to show either by direct probabilistic arguments or from the explicit formula for $D_k(t)$ that the Poisson distribution has the following properties [1]:

1. $\sum D_k(t) = 1,\ t \geqq 0.$
2. $D_k(t) \geqq 0,\ t \geqq 0.$
3. $D_k(t)$ is infinitely differentiable.
4. $D_k(0) = \begin{cases} 0, & k \neq 0, \\ 1, & k = 0. \end{cases}$
5. $D_k(r+t) = \sum_{i+j=k} D_i(r) D_j(t).$
6. $\sum kD_k(t) = at$ (expectation).
7. $\{D_k(t)\}$ are linearly independent.
8. $\{D_k(t)\}$ satisfy Descartes' law of signs in the interval $(0, \infty)$.

Therefore curves

$$P(t) = \sum D_k(t) P_k,$$

which use a Poisson distribution as blending functions automatically have the following

geometric properties: 1. well-defined, 2. convex hull, 3. smooth, 4. interpolates initial point, 5. extends to surfaces, 6. subdivision algorithm, 7. exactly reproduces points and lines, 8. nondegenerate and 9. variation diminishing.

Since they require an infinite sequence of control points, Poisson curves can have neither symmetry, nor a geometric construction algorithm, nor an augmentation algorithm. Nor do the functions $D_k(t)$ have local support. Hence there is no local control in the sense of § 2.13. However for N large and t small, $D_N(t)$ is negligibly small. Therefore if the points $\{P_k\}$ are bounded, then the points $\{P_k\}$, $k > N$, have little effect on the curve near $t=0$. Also

$$D_k^{(j)}(0) = \begin{cases} 0, & k > j, \\ (-1)^{j-k}\binom{j}{k}a^j, & k \leqq j, \end{cases}$$

so

$$P^{(j)}(0) = a^j \sum (-1)^{j-k}\binom{j}{k} P_k.$$

Therefore just like Bezier curves (see § 2.13), the jth derivative of a Poisson curve at $t=0$ depends only on its first $j+1$ control points.

We can use these formulas to calculate the curvature and the torsion of a Poisson curve at $t=0$. For curvature we have

$$K(0) = \frac{|P'(0) \times P''(0)|}{|P'(0)|^3} = \frac{|(P_1 - P_0) \times (P_2 - P_1)|}{|(P_1 - P_0)|^3},$$

and for torsion

$$T(0) = \frac{P'(0) \cdot [P''(0) \times P'''(0)]}{|P'(0)P''(0)|^2} = \frac{(P_1 - P_0) \cdot [(P_2 - P_1) \times (P_3 - P_2)]}{|(P_1 - P_0) \times (P_2 - P_1)|^2}.$$

Notice that the curvature depends only on the first three control points and the torsion only on the first four; moreover both are independent of a. The comparable formulas for Bezier curves (see § 2.13) are

$$K(0) = \frac{(n-1)}{n} \frac{|(P_1 - P_0) \times (P_2 - P_1)|}{|(P_1 - P_0)|^3},$$

$$T(0) = \frac{(n-2)}{n} \frac{(P_1 - P_0) \cdot [(P_2 - P_1) \times (P_3 - P_2)]}{|(P_1 - P_0) \times (P_2 - P_1)|^2}.$$

Therefore

$$\lim_{n \to \infty} \text{Bezier curvature } (0) = \text{Poisson curvature } (0),$$

$$\lim_{n \to \infty} \text{Bezier torsion } (0) = \text{Poisson torsion } (0).$$

Poisson curves are actually limiting cases of Bezier curves. Indeed let $n \to \infty$ and $t \to 0$ in such a way that $\lim (nt)$ exists and is finite. Then it is a well-known fact [1] that

$$\lim_{\substack{n \to \infty \\ t \to 0}} \text{binomial distribution } (t) = \text{Poisson distribution } \left[\lim_{\substack{n \to \infty \\ t \to 0}} (nt) \right].$$

For this reason the Poisson distribution is often used to approximate the binomial

distribution when n is large. Therefore it follows that

$$\lim_{\substack{n\to\infty \\ t\to 0}} \text{Bezier}\,[P_0, \cdots, P_n](t) = \text{Poisson}\,[P_0, P_1, \cdots]\left[\lim_{\substack{n\to\infty \\ t\to 0}} (nt)\right].$$

Thus one possible application of Poisson curves could be as a quick approximation for Bezier curves of high degree.

Poisson curves are different from the classical curves of computer aided geometric design because they use an infinite number of control points. Clearly some truncation will be required before these curves can be effectively employed. Therefore the convergence properties of Poisson curves need to be carefully understood before they can be applied directly to problems in computer aided geometric design.

5. Conclusions and questions. Probability theory is the key to deeper insight into many of the curves and surfaces of computer aided geometric design. Many geometric properties of these curves and surfaces are just reflections of corresponding probabilistic properties of their blending functions. Thus rather than derive these geometric properties from explicit representations of the blending functions, we have tried to give arguments based on their probabilistic interpretations. These arguments are simpler, more general, more natural and more elegant. By adopting this high level perspective, we have realized a deeper level of understanding.

Still, many questions remain. We must clarify the relationship between geometric construction algorithms and subdivision algorithms. For curves $P(t)$ which use one of Polya's urn distributions as blending functions, do the points $\{P_0^k(r)\}$ always subdivide the curve at $P(r)$? If not, can we indeed construct simple, general, subdivision algorithms? How?

As yet, we have been unable to derive the variation diminishing property from purely probabilistic considerations. Can this be done? We believe that the answer is yes because Descartes' law of signs can be interpreted as a statement about the expectation of a sequence of scalars $c_0, \cdots, c_n$ with respect to a discrete probability distribution $\{D_k^n(t)\}$. However so far we have met with little success in this direction.

Differential conditions—tangents, curvature, torsion—still elude direct probabilistic interpretations. Is there anything that probability theory can tell us about these critical conditions?

We have shown that the classical expectation of a discrete distribution is related to the geometric property of exactly reproducing straight lines. What is the geometric significance of the variance, or of the standard deviation, cr of the higher order means of a discrete distribution?

Laplace and Fourier transforms play a fundamental role in probability theory. Do they also have an important role in computer aided geometric design?

Finally, we have looked only at discrete probability distributions. What precisely is the role of continuous probability distributions in computer aided geometric design?

Appendix: Polya's urn model and Descartes' law of signs. In this appendix we shall give an elementary proof of the fact that the discrete probability distributions $D_0^n(t), \cdots, D_n^n(t)$ generated by Polya's urn model satisfy Descartes' law of signs in the interval $(0, 1)$. It then follows automatically from Theorem 2.2 that curves which use these polynomials as blending functions are necessarily variation diminishing.

To begin, recall that a collection of functions $F_0(t), \cdots, F_n(t)$ is said to satisfy Descartes' law of signs in the interval (a, b) iff for every collection of constants $c_0, \cdots, c_n$

$$\text{zeros in } (a, b)\,[\textstyle\sum c_k F_k(t)] \leqq \text{sign alternations of } (c_0, \cdots, c_n).$$

THEOREM A.1 (Descartes). *The power functions* $1, t, \cdots, t^n$ *satisfy Descartes' law of signs in the interval* $(0, \infty)$.

Proof. See [12]. □

LEMMA A.1. *Let* $p_0, \cdots, p_n$ *be a collection of positive constants, and let*

$$F_k(t) = p_k E_k(t), \qquad k = 0, 1, \cdots, n.$$

Then $F_0(t), \cdots, F_n(t)$ *satisfy Descartes' law of signs in the interval* (a, b) *iff* $E_0(t), \cdots, E_n(t)$ *satisfy Descartes' law of signs in the interval* (a, b).

Proof. Suppose that the functions $E_0(t), \cdots, E_n(t)$ satisfy Descartes' law of signs in the interval (a, b). Then

$$\begin{aligned}\text{zeros in } (a, b)\,[\textstyle\sum c_k F_k(t)] &= \text{zeros in } (a, b)\,[\textstyle\sum c_k p_k E_k(t)] \\ &\leqq \text{sign alternations } (p_0 c_0, \cdots, p_n c_n) \\ &= \text{sign alternations } (c_0, \cdots, c_n).\end{aligned}$$

Therefore the functions $F_0(t), \cdots, F_n(t)$ satisfy Descartes' law of signs in the interval (a, b).

Conversely if the functions $F_0(t), \cdots, F_n(t)$ satisfy Descartes' law of signs in the interval (a, b), then by what we have just proved the functions $E_0(t), \cdots, E_n(t)$ must also satisfy Descartes' law of signs in the interval (a, b) since

$$E_k(t) = \frac{1}{p_k} F_k(t). \qquad \square$$

LEMMA A.2. *Let* p, q *be positive constants, and let*

$$F_j(t) = \begin{cases} E_j(t), & j \neq k, \\ pE_k(t) + qE_{k\pm 1}(t), & j = k. \end{cases}$$

If $E_0(t), \cdots, E_n(t)$ *satisfy Descartes' law of signs in the interval* (a, b), *then* $F_0(t), \cdots, F_n(t)$ *satisfy Descartes' law of signs in the interval* (a, b).

Proof. To be specific, suppose that

$$F_k(t) = pE_k(t) + qE_{k+1}(t).$$

Then by construction

$$\textstyle\sum c_j F_j(t) = \sum s_j E_j(t),$$

where

$$s_j = \begin{cases} c_j, & j \neq k, k+1, \\ pc_k, & j = k, \\ c_{k+1} + qc_k, & j = k+1. \end{cases}$$

Therefore since $p, q > 0$

$$\text{sgn } (s_j) = \text{sgn } (c_j), \quad j \neq k+1, \qquad \text{sgn } (s_{k+1}) = \text{sgn } (c_k) \text{ or sgn } (c_{k+1}).$$

Now if

$$\text{sgn } (s_{k+1}) = \text{sgn } (c_{k+1})$$

then

$$\text{sign alternations } (s_k, s_{k+1}, s_{k+2}) = \text{sign alternations } (c_k, c_{k+1}, c_{k+2}).$$

On the other hand, if

$$\text{sgn}\,(s_{k+1}) = \text{sgn}\,(c_k)$$

then

$$\begin{aligned}\text{sign alternations}\,(s_k, s_{k+1}, s_{k+2}) &= \text{sign alternations}\,(c_k, c_k, c_{k+2}) \\ &\leqq \text{sign alternations}\,(c_k, c_{k+1}, c_{k+2}).\end{aligned}$$

Therefore in general

$$\text{sign alternations}\,(s_0, \cdots, s_n) \leqq \text{sign alternations}\,(c_0, \cdots, c_n).$$

Now suppose that the functions $E_0(t), \cdots, E_n(t)$ satisfy Descartes' law of signs in the interval (a, b). Then

$$\begin{aligned}\text{zeros in}\,(a, b)\,[\textstyle\sum c_j F_j(t)] &= \text{zeros in}\,(a, b)\,[\textstyle\sum s_j E_j(t)] \\ &\leqq \text{sign alternations}\,(s_0, \cdots, s_n) \\ &\leqq \text{sign alternations}\,(c_0, \cdots, c_n).\end{aligned}$$

Therefore the functions $F_0(t), \cdots, F_n(t)$ satisfy Descartes' law of signs in the interval (a, b). An exactly analogous argument works if

$$F_k(t) = pE_k(t) + qE_{k-1}(t). \qquad \square$$

We now introduce the following notation:

$$B_k^n(t) = \binom{n}{k} t^k (1-t)^{n-k}, \qquad b_k^n(t) = t^k (1-t)^{n-k},$$

$$D_k^n(t) = \binom{n}{k} \frac{t \cdots [t+(k-1)a](1-t) \cdots [(1-t)+(n-k-1)a]}{(1+a) \cdots [1+(n-1)a]},$$

$$d_k^n(t) = t \cdots [t+(k-1)a](1-t) \cdots [(1-t)+(n-k-1)a].$$

The polynomials $B_k^n(t)$ are the Bernstein polynomials, and the polynomials $D_k^n(t)$ are the polynomials which define the discrete probability ditributions generated by Polya's urn model (see § 3). The functions $b_k^n(t), d_k^n(t)$ are just these same polynomials with their constant coefficients stripped off. By Lemma A.1, to prove that the polynomials $B_0^n(t), \cdots, B_n^n(t)$ $(D_0^n(t), \cdots, D_n^n(t))$ satisfy Descartes' law of signs in the interval $(0, 1)$, it is enough to prove that the polynomials $b_0^n(t), \cdots, b_n^n(t)$ $(d_0^n(t), \cdots, d_n^n(t))$ satisfy Descartes' law of signs in the interval $(0, 1)$. This we now proceed to do.

THEOREM A.2 (Polya and Schoenberg). *The Bernstein polynomials $B_0^n(t), \cdots, B_n^n(t)$ satisfy Descartes' law of signs in the interval $(0, 1)$.*

Proof. Since it is short, we repeat the proof given in [10]. By Lemma A.1, we need only prove this result for the polynomials $b_0^n(t), \cdots, b_n^n(t)$. Now let $u = t/(1-t)$. Then by Theorem A.1

$$\begin{aligned}\text{zeros in}\,(0, 1)\,[\textstyle\sum c_k b_k^n(t)] &= \text{zeros in}\,(0, 1)\,[\textstyle\sum c_k t^k (1-t)^{n-k}] \\ &= \text{zeros in}\,(0, 1)\,\frac{[\sum c_k t^k (1-t)^{n-k}]}{(1-t)^n} \\ &= \text{zeros in}\,(0, \infty)\,[\textstyle\sum c_k u^k] \\ &\leqq \text{sign alternations}\,(c_0, \cdots, c_n).\end{aligned}$$

Therefore the polynomials $b_0^n(t), \cdots, b_n^n(t)$ satisfy Descartes' law of signs in the interval $(0,1)$. □

THEOREM A.3. *The polynomials $D_0^n(t), \cdots, D_n^n(t)$ which define the discrete probability distributions generated by Polya's urn model satisfy Descartes' law of signs in the interval* $(0,1)$.

Proof. By Lemma A.1 we need only prove this result for the polynomials $d_0^n(t), \cdots, d_n^n(t)$. Now we have just proved a special case of this result since Theorem A.2 is the case where $a=0$. The idea of the general proof is to start with the collection of functions $b_0^n(t), \cdots, b_n^n(t)$ $(a=0)$, and step by step to transform these functions into the functions $d_0^n(t), \cdots, d_n^n(t)$ $(a>0)$ all the while retaining Descartes' law of signs. That is, we shall construct sequences of functions $F_{0k}(t), \cdots, F_{nk}(t)$ $0 \leqq k \leqq L$ such that:

a. $F_{00}(t), \cdots, F_{n0}(t) = b_0^n(t), \cdots, b_n^n(t)$;
b. $F_{0L}(t), \cdots, F_{nL}(t) = d_0^n(t), \cdots, d_n^n(t)$;
c. $F_{0k}(t), \cdots, F_{nk}(t)$ satisfy Descartes' law of signs in the interval $(0,1)$.

We proceed as follows:

1. The first sequence is obtained from $b_0^n(t), \cdots, b_n^n(t)$ by replacing one factor of t by $(t+a)$ in the function $b_n^n(t)$; thus

$$F_{j1}(t) = \begin{cases} b_j^n(t), & j \neq n, \\ \dfrac{(t+a)}{t} b_n^n(t), & j = n. \end{cases}$$

2. The second sequence is obtained from the first sequence by replacing one factor of t by $(t+a)$ in the function $b_{n-1}^n(t)$.
3. Continue in this fashion down to $b_2^n(t)$, each time replacing one factor of t by $(t+a)$; this procedure generates $(n-1)$ sequences of functions.
4. Now return to $[(t+a)/t]b_n^n(t)$ and change one factor of t to $(t+2a)$.
5. Continue in this fashion down to $[(t+a)/t]b_3^n(t)$, each time changing one factor of t to $(t+2a)$; this procedure generates $(n-2)$ new sequences of functions.
6. Repeat this procedure for the terms $(t+3a), \cdots, (t+[n-1]a)$.
7. The last step generates the sequences of functions $E_0(t), \cdots, E_n(t)$ where $E_k(t) = t(t+a)\cdots(t+[k-1]a)(1-t)^{n-k}$.

Let us stop here for a moment and show that every sequence of functions which we have generated so far satisfies Descartes' law of signs in the interval $(0,1)$. The proof is by induction. Certainly by Theorem A.2 the 0th sequence satisfies Descartes' law of signs in the interval $(0,1)$ since the 0th sequence is just $b_0^n(t), \cdots, b_n^n(t)$. Now suppose that a sequence $G_0(t), \cdots, G_n(t)$ satisfies Descartes' law of signs in the interval $(0,1)$, and consider the very next sequence $H_0(t), \cdots, H_n(t)$. By construction if

$$G_{k-1}(t) = t(t+a)\cdots(t+ja)t^{k-j-2}(1-t)^{n-k+1},$$
$$G_k(t) = t(t+a)\cdots(t+ja)t^{k-j-1}(1-t)^{n-k},$$
$$G_{k+1}(t) = t(t+a)\cdots(t+ja)(t+[j+1]a)t^{k-j-1}(1-t)^{n-k-1},$$

then

$$H_i(t) = G_i(t), \qquad i \neq k,$$
$$H_k(t) = t(t+a)\cdots(t+ja)(t+[j+1]a)t^{k-j-2}(1-t)^{n-k}$$
$$= G_k(t) + (j+1)at(t+a)\cdots(t+ja)t^{k-j-2}(1-t)^{n-k}.$$

But notice that

$$G_k(t)+G_{k-1}(t)=t(t+a)\cdots(t+ja)t^{k-j-2}(1-t)^{n-k}.$$

Therefore

$$H_k(t)=[1+(j+1)a]G_k(t)+(j+1)aG_{k-1}(t).$$

Since by assumption $a>0$ and $G_0(t),\cdots,G_n(t)$ satisfy Descartes' law of signs in the interval $(0,1)$, it follows by Lemma A.2 that the functions $H_0(t),\cdots,H_n(t)$ also satisfy Descartes' law of signs in the interval $(0,1)$. Thus the property of satisfying Descartes' law of signs in the interval $(0,1)$ propagates down to the last sequence $E_0(t),\cdots,E_n(t)$.

Now apply the same construction to the factors $(1-t)$. That is, starting with $E_0(t)$ replace one factor of $(1-t)$ by $(1-t+a)$. Continue this procedure down to $E_{n-2}(t)$ generating $(n-1)$ new sequences. Then return to $[(1-t+a)/(1-t)]E_0(t)$ and repeat the preceeding construction for the terms $(1-t+2a),\cdots,(1-t+[n-1]a)$. The same argument as before shows that each sequence along the way must satisfy Descartes' law of signs in the interval $(0,1)$. But the last sequence is exactly $d_0^n(t),\cdots,d_n^n(t)$. This completes the proof. □

COROLLARY A.1. *Every curve $P(t)=\sum D_k^n(t)P_k$, which uses a distribution generated by Polya's urn model for blending functions, is variation diminishing.*

Proof. This result is an immediate consequence of Theorem A.3 and Theorem 2.2.

REFERENCES

[1] H. D. BRUNK, *An Introduction to Mathematical Statistics*, Ginn and Company, New York, 1960.

[2] K. L. CHUNG, *Elementary Probability Theory with Stochastic Processes*, Springer-Verlag, New York, 1975.

[3] F. EGGENBERGER AND G. POLYA, *Uber die Statistik Verketteter Vorgange*, Z. Angew. Math. Mech., 1 (1923), pp. 279–289.

[4] G. FARIN, *Subsplines uber Dreiecken*, Dissertation, Braunschweig, 1979.

[5] ———, *Bezier polynomials over triangles and the construction of piecewise C^r polynomials*, TR/91, Department of Mathematics, Brunel University, Uxbridge, Middlesex, U.K.

[6] A. R. FORREST, *Interactive interpolation and approximation by Bezier Polynomials*, Computer J., 15 (1972), pp. 71–79.

[7] R. GOLDMAN, *Using degenerate Bezier triangles and tetrahedra to subdivide Bezier curves*, Computer-Aided Design, 14 (1982), pp. 307–311.

[8] ———, *Subdivision algorithms for Bezier triangles*, Computer-Aided Design, 15 (1983), pp. 159–166.

[9] J. LANE AND R. RIESENFELD, *A theoretical development for the computer generation and display of piecewise polynomial surfaces*, IEEE Trans. PAMI, 2 (1980), pp. 35–46.

[10] G. POLYA AND I. J. SCHOENBERG, *Remarks on De La Vallée Poussin means and convex conformal maps of the circle*, Pacific J. Math., 8 (1958), pp. 296–334.

[11] M. A. SABIN, *The use of piecewise forms for the numerical representation of shape*, Ph.D. dissertation, Budapest, 1977.

[12] L. L. SCHUMAKER, *Spline Functions: Basic Theory*, John Wiley and Sons, New York, 1981.

[13] J. J. STOKER, *Differential Geometry*, John Wiley and Sons, New York, 1969.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 1, January 1985

A SIMPLE GAME WITH NO SYMMETRIC SOLUTION*

MOHAMED A. RABIE†

Abstract. This paper presents an n-person simple game in characteristic function form for which no von Neumann–Morgenstern solution (stable set) exhibits the symmetry of the characteristic function, for $n = 9$.

AMS 1970 subject classification. Primary 90D12
IAOR 1973 subject classification. Main: Games

Key words. game theory, solution, coalitions, stable set, simple games, voting games, symmetric solutions

1. Introduction. The following basic question was raised by L. S. Shapley at the Fourth International Workshop on Game Theory in 1978 [5]. Does every simple game have a solution that retains the symmetry of the game?

In this paper we answer this question in the negative by exhibiting simple games that have no symmetric solutions. These are either nonproper or nonstrong. It remains an open question whether every simple, strong and proper game has a symmetric solution.

Section 2 contains definitions. Section 3 describes the games and outlines the proof of the nonsymmetry of their solutions.

2. Definitions. We describe a *game* by its *characteristic function* v, which is a mapping from the set of subsets of the player set to the real numbers. A *simple game* (see [8] and [9]) has a characteristic function that takes on only the values 0 (on *losing* coalitions) and 1 (on *winning* ones).

We will consider games in which adding a player to a winning coalition preserves the winning property, so that the characteristic function is *monotone* increasing. A *proper* game does not have two disjoint winning coalitions. A *strong* game is one with no two losing coalitions whose union is the entire player set.

The *symmetry group* of a game is the set of permutations of players that preserve the characteristic function. An *imputation* is a nonnegative valued vector that sums to one, whose components correspond to the players. An imputation X *dominates* another imputation Y, if X's components are strictly greater than Y's on a winning coalition. A *solution* of a game is a set of imputations, no one dominating another, that among them dominate all other imputations. The condition that no imputation of a solution dominates another is called *internal stability*; that every other imputation is dominated is called *external stability*. Some games have many solutions ([7], [9]) and some have few ([2], [4]) or none at all ([3], [6]).

Every simple game has at least one solution, which we can obtain by taking all imputations whose support is contained in some minimal winning coalition.

A *symmetric solution* is a solution that is fixed by the symmetry group of the game.

3. The example. We first present a simple and strong, but not proper nine-person game that has no symmetric solution. This corresponds directly to a ten-person proper but not strong game with the same properties.

* Received by the editors March 31, 1981, and in revised form September 15, 1983. This research was supported in part by the Office of Naval Research under contract N00014-75-C-0678 NR 047-094 at Cornell University.

† Sana'a University, Department of Mathematics, Faculty of Sciences, Sana'a, Yemen Arab Republic.

We number the players $1, \cdots, 9$. In this game, all coalitions of three or more players win except $(1, 2, 3)$, $(4, 5, 6)$ and $(7, 8, 9)$.

If we add a tenth player and insist that the winning coalitions include him along with the sets of three or more players as indicated among the nine, we obtain a proper but not strong game with the same properties. (See Gillies [1].)

We now prove the nonexistence of a symmetric solution to these games.

THEOREM 1. *The nine-person game just described has no symmetric solution.*

Proof. It is obvious that no imputation that dominates an image of itself under a symmetry operation can belong to a symmetric solution without violating the internal stability of that solution. This severely limits the form of "allowable imputations" that can belong to such a solution, as follows:

1. No allowable imputation can take on two distinct values on each of $(1, 2, 3)$, $(4, 5, 6)$ and $(7, 8, 9)$. (Otherwise on the winning coalition consisting of the inverse image of the larger of these values it dominates any imputation obtained by switching those players with the inverse images of the smaller values.)

2. An allowable imputation that takes on two distinct values on two of $(1, 2, 3)$, $(4, 5, 6)$ and $(7, 8, 9)$ (say the first two) must have the form $(a, b, b, c, b, b, b, b, b)$ up to permutations.

From the previous step the allowed values are $(a, b, c, d, e, f, g, g, g)$.

If the first three values were all distinct we would arrange to have $a > b > c$, $d > e$ and by permuting $3 \to 2 \to 1 \to 3$, and $5 \to 4 \to 5$, arrange to obtain an imputation dominated by the original one on $(1, 2, 4)$. This limits the allowed form to $(a, b, b, c, d, d, e, e, e)$.

If $b \neq e$, say $b < e$, we may permute $(1, 2, 3)$ with $(7, 8, 9)$ and 4 with 5, again obtaining dominance on a winning coalition $(8, 9, 4)$ or $(8, 9, 5)$.

3. An allowed imputation that is constant on two of $(1, 2, 3)$, $(4, 5, 6)$ and $(7, 8, 9)$ takes the same value on both and therefore has the form $(a, b, c, d, d, d, d, d, d)$ up to permutations. Moreover all of a, b, c must be greater than d, or all less than d, or one must equal d. These follow by arguments similar to those above, which we leave to the reader.

A symmetric solution must therefore consist of imputations that, up to permutation, have the form $(a, b, c, d, d, d, d, d, d)$, $(a, b, b, c, b, b, b, b, b)$ or $(a, b, c, b, b, b, b, b, b)$.

Moreover, such a solution can contain imputations having only exactly one value of d in the former form or b in the latter form, and if both forms were present these would have to have the same value. Otherwise the solution would obviously lack internal stability.

It is necessary that each of the following imputations be in or be dominated by an imputation in such symmetric solution:

$$\mathrm{I} = (\tfrac{1}{6}, \tfrac{1}{6}, \tfrac{1}{6}, \tfrac{1}{6}, \tfrac{1}{6}, \tfrac{1}{6}, 0, 0, 0)$$

$$\mathrm{II} = (\tfrac{1}{6}, \tfrac{1}{6}, 0, \tfrac{1}{6}, \tfrac{1}{6}, 0, \tfrac{1}{6}, \tfrac{1}{6}, 0)$$

$$\mathrm{III} = (\tfrac{1}{9}, \tfrac{1}{9}, \tfrac{1}{9}, \tfrac{1}{9}, \tfrac{1}{9}, \tfrac{1}{9}, \tfrac{1}{9}, \tfrac{1}{9}, \tfrac{1}{9})$$

$$\mathrm{IV} = (\tfrac{1}{8}, \tfrac{1}{8}, \tfrac{1}{8}, \tfrac{1}{8}, \tfrac{1}{8}, \tfrac{1}{8}, \tfrac{1}{8}, \tfrac{1}{8}, 0).$$

The choice $\frac{1}{6}$ (imputation I) fails to be a solution since it does not dominate imputation II. In order to dominate I then we must have $a > \frac{1}{6}$ for some imputation in the solution. To dominate III a solution of the first form would have to have $\frac{1}{6} > d > \frac{1}{9}$; since, by the remarks above, a, b and c would all have to be larger than d the entries

in this imputation could not possibly sum to one, so that the first form cannot occur.

In either of the other two forms, domination of I requires $a > \frac{1}{6}$, of III requires $b > \frac{1}{9}$, and of IV requires b or $c > \frac{1}{8}$. These requirements are incompatible with the condition that $a + c + 7b = 1$, which completes the proof.

Acknowledgments. The author would like to thank William Lucas for suggestions and valuable comments, as well as Daniel Kleitman and W. T. Trotter for major editorial assistance.

REFERENCES

[2] D. B. GILLIES, *Solutions to general non-zero-sum games*, Contributions to the Theory of Games, Vol. IV, Annals of Math. Studies, No. 40, A. W. Tucker and R. D. Luce, eds., Princeton Univ. Press, Princeton, NJ, 1959 pp. 47–85.

[2] W. F. LUCAS, *On solutions for n-person games*, RM-5567-PR, The Rand Corp., Santa Monica, CA, 1968.

[3] ———, *A game with no solution*, Bull. Amer. Math. Soc., 74 (1968), pp. 273–279.

[4] ———, *Games with unique solutions that are nonconvex*, Pacific J. of Math., 28 (1969), pp. 599–602.

[5] ———, *Report on the fourth international workshop in game theory*, Tech. Rept. No. 392, School of O.R. & I.E. Cornell University, Ithaca, NY, 1978.

[6] W. F. LUCAS AND M. A. RABIE, *Games with no solutions and empty cores*, Math. Oper. Res., 7 (1982), pp. 491–500.

[7] L. S. SHAPLEY, *Solutions of compound simple games*, Advances in Game Theory, Annals of Math. Studies, No. 52, M. Dresher, L. S. Shapley and A. W. Tucker, eds., Princeton Univ. Press, Princeton, NJ, 1964, pp. 443–476.

[8] ———, *Simple games: an outline of descriptive theory*, Behavioral Science, 7 (1966), pp. 59–66.

[9] J. VON NEUMANN AND O. MORGENSTERN, *Theory of Games and Economic Behavior*, Princeton Univ. Press, Princeton, NJ, 1944 (3rd ed., 1953).

SIAM J. ALG. DISC. METH.
Vol. 6, No. 1, January 1985

003

DECOMPOSITION OF FUZZY MATRICES*

HIROSHI HASHIMOTO†

Abstract. A problem of decomposition of fuzzy rectangular matrices is examined and some properties of decomposition are shown. Any fuzzy matrix can be factored into a product of a square matrix and a rectangular matrix of the same dimension. This square matrix has reflexivity and transitivity. The decomposition of fuzzy matrices is closely related to fuzzy databases and fuzzy retrieval models.

1. Introduction. We consider decomposition of fuzzy rectangular matrices. It is shown that any fuzzy matrix is factored into a product of a square matrix and a rectangular matrix of the same dimension. This square matrix has reflexivity and transitivity, so that it is a matrix which represents a preorder [2], [8]. The decomposition of fuzzy matrices is closely related to fuzzy databases and fuzzy retrieval models.

2. Definitions. Some operations and notation are defined. For x, y in the interval $[0, 1]$, $x+y$, xy, $x-y$, $x*y$ are defined as follows.

$$x+y=\max(x, y),$$

$$xy=\min(x, y),$$

$$x-y=\begin{cases} x & \text{if } x>y, \\ 0 & \text{if } x\leqq y, \end{cases}$$

$$x*y=\begin{cases} 1 & \text{if } x\geqq y, \\ x & \text{if } x<y. \end{cases}$$

Next we define some matrix operations on fuzzy matrices whose elements exist in the interval $[0, 1]$. Let $A=[a_{ij}]$ $(m\times n)$, $B=[b_{ij}]$ $(m\times n)$, $F=[f_{ij}]$ $(n\times l)$, and $R=[r_{ij}]$ $(n\times n)$. Then the following operations are defined.

$$AF=\left[\sum_{k=1}^{n} a_{ik}f_{kj}\right].$$

$$A*F=\left[\prod_{k=1}^{n} (a_{ik}*f_{kj})\right].$$

$A'=[a_{ji}]$ (the transpose of A).

$A\leqq B$ if and only if $a_{ij}\leqq b_{i3}$ for all i, j.

$\Delta R=R-R'$.

Furthermore some special types of fuzzy matrices are defined [2], [8]. A matrix R is said to be transitive if $R^2\leqq R$. A matrix R, all of whose diagonal elements are one, is called reflexive. Conversely a matrix R, all of whose diagonal elements are zero, is called irreflexive. A matrix R is nilpotent if $R^n=0$ (0 is the zero matrix). We deal only with fuzzy matrices.

3. Results. Using the operations defined above we construct a square matrix which represents a hierarchy of rows of a given rectangular matrix. This matrix has

* Received by the editors November 24, 1982, and in revised form October 11, 1983.
† Faculty of Economics, Yamaguchi University, Yamaguchi City, 753 Japan.

reflexivity and transitivity, and plays an important role in decomposition of the matrix. We show some properties of the square matrix. Then we prove some theorems and propositions on the decomposition of fuzzy matrices.

LEMMA 1. *If $A=[a_{ij}]$ is an $m\times n$ fuzzy matrix, then $A*A'$ is reflexive and transitive.*

Proof. Let $S=[s_{ij}]=A*A'$. That is,

$$s_{ij}=\prod_{k=1}^{n}(a_{ik}*a_{jk}).$$

Clearly

$$s_{ii}=\prod_{k=1}^{n}(a_{ik}*a_{ik})=1.$$

Thus S is reflexive.

Suppose that $s_{il}s_{lj}=c>0$ for some l. Then

$$s_{il}=\prod_{k=1}^{n}(a_{ik}*a_{lk})\geqq c,$$

$$s_{lj}=\prod_{k=1}^{n}(a_{lk}*a_{jk})\geqq c.$$

If $s_{ij}<c$, then

$$a_{ih}<a_{jh} \quad \text{and} \quad a_{ih}<c$$

for some h. Therefore since $s_{il}\geqq c$ and $s_{lj}\geqq c$ we have

$$c>a_{ih}\geqq a_{lh}\geqq a_{jh},$$

which is a contradiction. Hence $s_{ij}\geqq c$, so that S is transitive. □

Letting A_i be the ith row of A, if $A_i\geqq A_j$, then $s_{ij}=1$, where s_{ij} is the (i,j) entry of $S=A*A'$. Hence the matrix S represents inclusion among the rows of A. In other words, S gives the hierarchy of the rows of A. The hierarchy is reflexive and transitive. This fact becomes clearer if A is Boolean [3].

A reflexive and transitive relation is called a preorder, which has some interesting properties [2], [8]. If an $n\times n$ fuzzy matrix R is reflexive and transitive, then as is well-known R is idempotent, that is, $R^2=R$.

PROPOSITION 1. *Let $S=[s_{ij}]$ be an $m\times m$ fuzzy matrix. Then the following conditions are equivalent.*

(1) *The matrix S is reflexive and transitive.*

(2) $S*S'=S$.

Proof. (1) implies (2). Suppose that

$$\prod_{k=1}^{m}(s_{ik}*s_{jk})=c>0.$$

Then by setting $k=j$ we have $s_{ij}\geqq c$. Next we show that $S\leqq S*S'$. Suppose that $s_{ij}=c>0$. If $s_{il}<c$ and $s_{il}<s_{jl}$ for some l, then

$$s_{il}\geqq s_{ij}s_{jl}=cs_{jl}\geqq cs_{il}=s_{il},$$

so that $s_{il} = s_{jl}$, which is a contradiction. Hence

$$\prod_{k=1}^{m} (s_{ik} * s_{jk}) \geqq c,$$

so that $S \leqq S * S'$.

Equation (2) implies (1). This is clear from Lemma 1. □

LEMMA 2. *If $A = [a_{ij}]$ is an $m \times n$ fuzzy matrix, then*

$$(A * A')A = A.$$

Proof. Let $B = [b_{ij}] = (A * A')A$. That is,

$$b_{ij} = \sum_{k=1}^{m} \prod_{l=1}^{n} (a_{il} * a_{kl}) a_{kj}.$$

By Lemma 1, $A * A'$ is reflexive, so that $A \leqq B$. We show that $B \leqq A$. Suppose that $b_{ij} > a_{ij}$. Then

$$\prod_{l=1}^{n} (a_{il} * a_{hl}) > a_{ij}, \; a_{hj} > a_{ij},$$

for some h. For $l = j$ we have

$$a_{ij} * a_{hj} > a_{ij},$$

so that $a_{ij} \geqq a_{hj}$, which is a contradiction. Hence $b_{ij} \leqq a_{ij}$. □

In the language of information retrieval [5], [7], A is called a fuzzy term-document matrix. Then $A * A'$ is considered to be a fuzzy term-term matrix which represents a hierarchy of terms. However since $A * A'$ is obtained by using A, if we multiply $A * A'$ by A, any information is not added to A. That is, the product $(A * A')A$ is equal to A.

LEMMA 3. *If $A = [a_{ij}]$ is an $m \times n$ fuzzy matrix and $S = [s_{ij}]$ is an $m \times m$ transitive matrix, then*

$$SA = S(A - QA),$$

where $Q = [q_{ij}]$ is an $m \times m$ nilpotent matrix such that $Q \leqq S$.

Proof. Let

$$B = [b_{ij}] = SA, \qquad C = [c_{ij}] = S(A - QA).$$

That is

$$b_{ij} = \sum_{k=1}^{m} s_{ik} a_{kj}, \qquad c_{ij} = \sum_{k=1}^{m} s_{ik} \left(a_{kj} - \sum_{l=1}^{m} q_{kl} a_{lj} \right).$$

Since it is clear that $c_{ij} \leqq b_{ij}$, we show that $b_{ij} \leqq c_{ij}$. Suppose that $b_{ij} = b > 0$ and $c_{ij} < b$. Then

$$s_{il(0)} \geqq b, \qquad a_{l(0)j} \geqq b$$

for some $k = l(0)$. Since $b_{ij} > c_{ij}$, we have

$$\sum_{l=1}^{m} q_{l(0)l} a_{lj} \geqq a_{l(0)j} \geqq b.$$

Thus

$$q_{l(0)l(1)} \geqq b, \quad a_{l(1)j} \geqq b, \quad s_{l(0)l(1)} \geqq b$$

for some $l(1)$. Therefore

$$s_{il(1)} \geqq b, \quad a_{l(1)j} \geqq b, \quad q^{(1)}_{l(0)l(1)} \geqq b.$$

Since $b_{ij} > c_{ij}$, we have

$$\sum_{l=1}^{m} q_{l(1)l} a_{lj} \geqq a_{l(1)j} \geqq b.$$

Thus

$$q_{l(1)l(2)} \geqq b, \quad a_{l(2)j} \geqq b, \quad s_{l(1)l(2)} \geqq b$$

for some $l(2)$. Therefore

$$s_{il(2)} \geqq b, \quad a_{l(2)j} \geqq b, \quad q^{(2)}_{l(0)l(2)} \geqq b.$$

By repeating the same argument

$$s_{il(m)} \geqq b, \quad a_{l(m)j} \geqq b, \quad q^{(m)}_{l(0)l(m)} \geqq b.$$

This contradicts the fact that Q is nilpotent. Hence $b_{ij} \leqq c_{ij}$. □

Similarly, we obtain the following lemma.

LEMMA 4. *If A is an $m \times n$ fuzzy matrix and R is an $n \times n$ transitive matrix, then*

$$AR = (A - AP)R,$$

where P is an $n \times n$ nilpotent matrix such that $P \leqq R$.

The above lemma is very useful for retrieval models [5], [7]. That is, A is a document-keyword matrix and the matrix R plays a role of a fuzzy thesaurus.

THEOREM 1. *If A is an $m \times n$ fuzzy matrix, then*

$$A = (A * A')(A - QA),$$

where Q is an $m \times m$ nilpotent matrix such that $Q \leqq A * A'$.

Proof. By Lemma 1, $A * A'$ is transitive. Therefore by Lemma 3

$$(A * A')A = (A * A')(A - QA).$$

Using Lemma 2, we have

$$A = (A * A')(A - QA). \qquad \square$$

The above theorem shows that any fuzzy matrix can be factored into a product of two matrices. Decomposition of matrices is important to simplification of various systems.

Similarly we obtain the following theorem.

THEOREM 2. *If A is an $m \times n$ fuzzy matrix, then*

$$A = (A - AP)(A' * A)',$$

where P is an $n \times n$ nilpotent matrix such that $P \leqq (A' * A)'$.

Since an irreflexive and transitive matrix is nilpotent, the following two corollaries are obtained.

COROLLARY 1. *If A is an $m \times n$ fuzzy matrix, then*

$$A = (A * A')(A - QA),$$

where Q is an $m \times m$ irreflexive and transitive matrix such that $Q \leqq A * A'$.

COROLLARY 2. *If A is an $m \times n$ fuzzy matrix, then*

$$A = (A - AP)(A' * A)',$$

where P is an $n \times n$ irreflexive and transitive matrix such that $P \leqq (A' * A)'$.

LEMMA 5. *Let $S = [s_{ij}]$ and $Q = [q_{ij}]$ be $m \times m$ transitive matrices. If $S \leqq Q$, then $S - Q'$ is irreflexive and transitive.*

Proof. Let $H = [h_{ij}] = S - Q'$. That is,

$$h_{ij} = s_{ij} - q_{ji}.$$

Then

$$h_{ii} = s_{ii} - q_{ii} = 0,$$

so that H is irreflexive. Next suppose that

$$h_{ik} h_{kj} = c > 0.$$

Then there are two cases.

Case 1. $s_{ik} = c$, $s_{ik} > q_{ki}$, $s_{kj} \geqq c$.

Case 2. $s_{ik} \geqq c$, $s_{kj} = c$, $s_{kj} > q_{jk}$.

Clearly $s_{ij} \geqq c$. Suppose that $q_{ji} \geqq c$. In the first case

$$q_{ki} \geqq q_{kj} q_{ji} \geqq c,$$

which is a contradiction. Furthermore, in the second case

$$q_{jk} \geqq q_{ji} q_{ik} \geqq c,$$

which is a contradiction. Hence $q_{ji} < c$, so that $h_{ij} \geqq c$. That is, H is transitive. □

By Lemma 5 we obtain the following lemma.

LEMMA 6. *If S is an $m \times m$ transitive matrices, then ΔS is irreflexive and transitive.*

The operation Δ is useful for a discussion of preferences and has some important properties [4]. Now we obtain the following two corollaries by Theorem 1, Theorem 2, and Lemma 6.

COROLLARY 3. *If A is an $m \times n$ fuzzy matrix, then*

$$A = (A * A')(A - \Delta SA),$$

where $S = A * A'$.

COROLLARY 4. *If A is an $m \times n$ fuzzy matrix, then*

$$A = (A - A\Delta R)(A' * A)',$$

where $R = (A' * A)'$.

PROPOSITION 2. *If A is an $m \times n$ fuzzy matrix and F is an $n \times l$ fuzzy matrix, then*

$$AF = (A - AP)F,$$

where P is nilpotent and $P \leqq F * F'$.

Proof. By Lemma 4

$$(A - AP)(F * F') = A(F * F').$$

By Lemma 2

$$(F * F')F = F.$$

Hence

$$(A - AP)(F * F')F = A(F * F')F,$$

so that

$$(A - AP)F = AF. \qquad \square$$

PROPOSITION 3. *If A is an $m \times n$ fuzzy matrix and F is an $n \times l$ fuzzy matrix, then*

$$AF = A(F - PF),$$

where P is nilpotent and $P \leqq (A' * A)'$.

Using Proposition 2 and Proposition 3 we have the following two propositions, respectively.

PROPOSITION 4. *If A is an $m \times n$ fuzzy matrix and F is an $n \times l$ fuzzy matrix, then*

$$AF = (A - A\Delta R)F,$$

where $R = F * F'$.

PROPOSITION 5. *If A is an $m \times n$ fuzzy matrix and F is an $n \times l$ fuzzy matrix, then*

$$AF = A(F - \Delta RF),$$

where $R = (A' * A)'$.

The following proposition is obvious, but it is useful for the decomposition of fuzzy matrices.

PROPOSITON 6. *Let $A = [a_{ij}]$ be an $m \times n$ fuzzy matrix and let $F = [f_{ij}]$ be an $n \times l$ fuzzy matrix ($n \geqq 2$). If*

$$\sum_{k \neq p}^{n} a_{ik} f_{kj} \geqq a_{ip} f_{pj}$$

for all i, j, then deleting both the pth column of A and the pth row of F does not change AF.

Example 1. Let

$$A = \begin{bmatrix} 0.1 & 0.6 & 0.5 & 0.1 \\ 0.1 & 0.2 & 0.2 & 0.1 \\ 0.2 & 0.4 & 0.6 & 0.3 \end{bmatrix}.$$

We decompose A by Corollary 3. Then

$$S = A * A' = \begin{bmatrix} 1 & 1 & 0.1 \\ 0.2 & 1 & 0.1 \\ 0.4 & 1 & 1 \end{bmatrix},$$

$$\Delta S = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0.4 & 1 & 0 \end{bmatrix},$$

$$\Delta SA = \begin{bmatrix} 0.1 & 0.2 & 0.2 & 0.1 \\ 0 & 0 & 0 & 0 \\ 0.1 & 0.4 & 0.4 & 0.1 \end{bmatrix},$$

$$A - \Delta SA = \begin{bmatrix} 0 & 0.6 & 0.5 & 0 \\ 0.1 & 0.2 & 0.2 & 0.1 \\ 0.2 & 0 & 0.6 & 0.3 \end{bmatrix}.$$

Thus A is decomposed as follows:

$$A = S(A - \Delta SA) = \begin{bmatrix} 1 & 1 & 0.1 \\ 0.2 & 1 & 0.1 \\ 0.4 & 1 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0.6 & 0.5 & 0 \\ 0.1 & 0.2 & 0.2 & 0.1 \\ 0.2 & 0 & 0.6 & 0.3 \end{bmatrix}.$$

Using Proposition 6 ($p = 2$) we have

$$A = \begin{bmatrix} 1 & 0.1 \\ 0.2 & 0.1 \\ 0.4 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0.6 & 0.5 & 0 \\ 0.2 & 0 & 0.6 & 0.3 \end{bmatrix}.$$

4. Concluding remarks. Sanchez [6] introduced a matrix operation equivalent to $*$ in order to solve fuzzy equations. He showed some interesting properties of the operation. It is an important matter to solve fuzzy equations in the fields such as fuzzy control [1].

Decomposition of rectangular fuzzy matrices may be useful for decomposition of fuzzy databases. By the decomposition we can know a hierarchy of keys or attributes. Decomposition of fuzzy matrices is closely related to reduction of fuzzy retrieval models.

REFERENCES

[1] E. CZOGAŁA AND W. PEDRYCZ, *On identification in fuzzy systems and its applications in control problems*, Fuzzy Sets and Systems, 6 (1981), pp. 73–83.
[2] D. DUBOIS AND H. PRADE, *Fuzzy Sets and Systems: Theory and Applications*, Academic Press, New York, 1980.
[3] K. H. KIM, *Boolean Matrix Theory and Applications*, Marcel Dekker, New York, 1982.
[4] S. V. OVCHINNIKOV, *Structure of fuzzy binary relations*, Fuzzy Sets and Systems, 6 (1981), pp. 169–195.
[5] G. SALTON, *Automatic Information Organization and Retrieval*, McGraw-Hill, New York, 1968.
[6] E. SANCHEZ, *Resolution of composite fuzzy relation equations*, Inform. and Control, 30 (1976), pp. 38–48.
[7] V. TAHANI, *A fuzzy model of document retrieval systems*, Information Processing & Management, 12 (1976), pp. 177–187.
[8] L. A. ZADEH, *Similarity relations and fuzzy orderings*, Information Sciences, 3 (1971), pp. 177–200.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 1, January 1985

004

COMPLEXITY AND STABILITY IN COMPARTMENTAL MODELS*

GILBERT G. WALTER†

Abstract. Compartmental models, by which flows through various systems can be studied, have a dual aspect: one structural and the other dynamic. The structural leads to a directed graph and may be analyzed by means of graph theory. The dynamic leads to a system of differential equations which are usually linear. The coefficient matrix in this case has a special form, that of the negative transpose of an M-matrix in which the off diagonal elements are nonnegative and the columns add up to zero. Hence the eigenvalues have a nonpositive real part.

If the digraph is weakly connected, the differential equation has a stable equilibrium solution; if it is unilaterally connected, the solution is unique; if it is strongly connected, the solution is feasible as well. It is possible to define various indices of stability which may then be shown to be related to indices of complexity of the structure. However, it is also possible, by redirecting the flows, to show that a given model can be reduced to a mammillary system with the same equilibrium solution. Hence any index of stability based on the equilibrium solution has no relation to a complexity index based on the number of arcs per vertex. Other stability indices, however, increase with increasing complexity.

1. Introduction. Compartmental models are used for the analysis and simulation of systems arising in a number of diverse disciplines such as ecology, economics, physiology, genetics, psychology, and chemistry. We shall be motivated mainly by applications to ecology although our results could be used in these other disciplines as well. While they are not the only models used in ecology, many of the others, such as Lotka–Volterra equations, Markov chains, Leslie matrices and even logistic equations can be interpreted as special cases of compartmental models.

The question with which we shall be concerned is the relation between the complexity and stability of ecosystems. The conventional wisdom has been that more complex systems are more stable. However, May [8] showed that for certain models, the opposite can be true. We shall interpret this question in terms of compartmental models with linear donor controlled flows. Some of the results which we present have appeared in different form elsewhere [11], [12], [13], [14], [15], [16], [17].

The construction of a compartmental model is straightforward and highly intuitive. The ecosystem (or any system) is partitioned into homogeneous compartments and the flow of nutrients or energy (or of money, goods, electrons, radioactive tracers, etc.) traced between them.

In order to keep the discussion general we shall refer to the *flow* of *material* between compartments with the understanding that it could be any of those. Similarly we shall be concerned as well with the *level* of *material* in each compartment. The compartments are represented by boxes and the flow by arrows. (See Fig. 1.)

This representation is the graphical or structural aspect of the compartmental model. Some information can already by gleaned from it at this stage. The theory of *directed graphs* may be applied by interpreting the compartments as vertices and the flows as arcs. However, in order to simulate or analyze a system, the model must be quantified. This is done by studying the rate of change of the level x_i of the ith compartment in time. The flow between the ith and the jth compartment, designated f_{ij}, is a rate, measured in quantity of material per unit time.

* Received by the editors July 1, 1983, and in revised form September 27, 1983. This paper was presented at SIAM 1983 National Meeting, June 1983, Denver, Colorado under the title, *Stability and structure of compartmental models.*

† Department of Mathematical Sciences, University of Wisconsin-Milwaukee, Milwaukee, Wisconsin 53201.

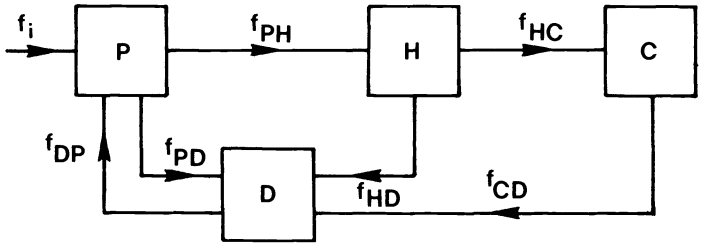


FIG. 1. *The flow of nutrients through a simple ecosystem. The nutrient input (f_i) is taken up by plants (P) some of which are eaten by herbivores (H) which in turn are eaten by carnivores (C). Part of the nutrients from each compartment flows to the detritivores (D) which in turn make the nutrient available to the plants.*

A differential equation describing the behavior of x_i may be obtained by equating the time rate of change of x_i to the difference between the flow rates coming in and those going out of the ith compartment, i.e.

$$\frac{dx_i}{dt} = \sum_{k=0}^{n} f_{ki} - \sum_{j=0}^{n} f_{ij}, \qquad i = 1, 2, \cdots, n. \tag{1.1}$$

The subscript 0 denotes flows coming from or going to the outside of the system. The f_{ij} may be constant or variable in time and they may be and usually are, functionally dependent on the x_i's.

The most widely used assumption, in particular in physiology and medicine, is that the functional form of the f_{ij}, the flow rate, is

$$f_{ij} = a_{ij}x_i, \qquad i = 1, 2, \cdots, n, \quad j = 0, 1, \cdots, n. \tag{1.2}$$

Very often, particularly in ecosystems, the use of the derivative is inappropriate and a finite difference should be used instead. This happens, e.g. if diurnal data are used. Then the equation (1.1) is replaced by

$$\frac{\Delta x_i}{h} = \sum_{k=0}^{n} \bar{f}_{ki} - \sum_{j=0}^{n} \bar{f}_{ij}, \qquad i = 1, 2, \cdots, n, \tag{1.3}$$

where h is the time step and $\bar{f}_{ij}$ are the flow rates averaged over the time h. If the equation (1.1) is written with the flow rates of (1.2), and if there are no flows to or from the outside (i.e. the system is closed), then it can be expressed in matrix form as

$$\frac{dX}{dt} = AX \tag{1.4}$$

where $-A^T$ is a singular M matrix. Similarly if the time step h in (1.3) is sufficiently small, and the $\bar{f}_{ij}$ given by (1.2), it becomes

$$\Delta X = hAX. \tag{1.5}$$

If, furthermore, the levels are normalized and $X_k = X(kh)$, then (1.5) becomes

$$X_{k+1} = PX_k = (I + hA)X_k \tag{1.6}$$

where P is a stochastic matrix. Thus we obtain a Markov chain version of the model.

The structure of a closed compartmental model can be represented by a directed graph in which the compartments are vertices and the flows arcs. For example, Fig. 1, without the flow in, has the directed graph given in Fig. 2.

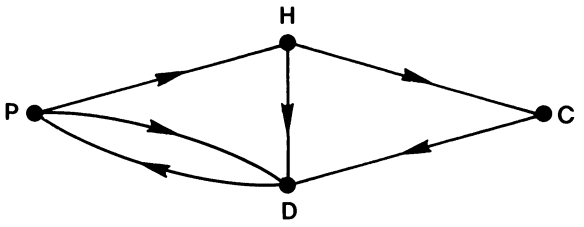


FIG. 2. *The digraph of the compartmental model of Fig.* 1.

In this work we first (§ 2) present some results that are straightforward or well known regarding the relation between the structure and the stability of the model. In § 3, we study various measures of complexity and in § 4 some measures of stability. Finally in § 5 we compare them to each other and study the effect on them of transformations of the model.

2. Some basic properties. The results presented in this section are either well known or not very complicated. We first observe that because of the nature of the matrix A in (1.4), namely that the columns sum to zero, that the main diagonal elements are nonpositive and that the off diagonal elements nonnegative, its eigenvalues, if not zero, must have a negative real part. This is a consequence of Gershgorin's theorem (see [7, p. 146] or [3]). Hence the solution to the differential equation approaches an equilibrium solution X_∞ as $t \to \infty$, at least if the rank of $A = n-1$. Even if the rank $< n-1$, the same conclusion follows (see [3, p. 45]). Moreover the equilibrium solution $X_\infty \geqq 0$ if the initial vector is.

From the standpoint of ecosystems an important question is the determination of which compartments will be zero and which will be positive ultimately. The answer depends on the structure of the digraph.

A digraph (V, A) is classified as weakly connected, unilaterally connected, or strongly connected if there exists respectively a complete semi-path, a complete path, or a complete closed path (see [9, Chap. 2]). These are illustrated in Fig. 3.

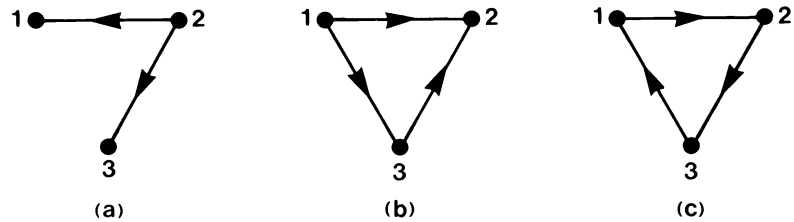


FIG. 3. *Digraphs which are weakly connected* (a), *unilaterally connected* (b), *and strongly connected* (c).

In the weakly connected case the equilibrium solution X_∞ depends on X_0 and will always have at least one empty compartment. This is clear in the case of Fig. 3(a) since compartment 2 will ultimately empty out. In this case the matrix has the form

$$A = \begin{bmatrix} 0 & a & 0 \\ 0 & -a-b & 0 \\ 0 & b & 0 \end{bmatrix} \tag{2.1a}$$

and there are two linearly independent equilibrium solutions.

In the unilaterally connected case the equilibrium solution depends only on the magnitude of X_0. It will again have at least one compartment which ultimately will

empty out. The matrix for the example in Fig. 3(b) has the form

$$A=\begin{bmatrix} -a-b & 0 & 0 \\ a & 0 & c \\ b & 0 & -c \end{bmatrix} \tag{2.1b}$$

whence it follows that $X_\infty = \alpha[0,1,0]^T$.

For strongly connected digraphs the X_∞ is positive and again depends only on the magnitude of X_0. The matrix looks like

$$A=\begin{bmatrix} -a & 0 & c \\ a & -b & 0 \\ 0 & b & -c \end{bmatrix}.$$

The statements in the general cases are most easily proved using the Markov chain version of the equations (1.6). The three cases lead to respectively, (a) an absorbing Markov chain with multiple absorbing states, (b) an absorbing chain with one absorbing state, and (c) a regular chain. The conclusions are then straightforward. (See [9], [11], [1].)

One could also use properties of M-matrices to reach the same conclusion. (See [3].)

3. Complexity. A recurring problem in ecology is the relation between the complexity of an ecosystem and its stability. Most ecologists assumed the two concepts went together, i.e. greater complexity was associated with greater stability. However May [8] in his 1973 monograph challenged this assumption and indeed showed that for Lotka–Volterra models of ecosystems, the opposite is sometimes true. However he did not consider compartmental models and used only the number of nonzero flows as an indicator of complexity. For compartmental models another approach, which appears in [16], is possible.

This alternate approach uses a family of complexity indices Γ_β which are similar to diversity indices and some of which are based on information theory [10]. It uses the Markov chain model (1.6) but takes the limit as $h \to 0$ to avoid dependence on the time step.

DEFINITION 3.1 [16]. Let p be an element of the transition matrix $P = I + hA$. The *weakness* of p will be a monotone function of the form

$$\begin{aligned} &w_\alpha(p) = (p^{-\alpha}-1)/\alpha, && 0<\alpha\leqq 1, \quad 0<p\leqq 1, \\ &w_0(p) = -\log p, && 0<p\leqq 1. \end{aligned} \tag{3.1}$$

DEFINITION 3.2 [16]. The *α-complexity index*, Γ_α, of a compartmental model, is given by

$$\Gamma_\alpha = \sum_{j=1}^{n} \lim_{h\to 0} \sum_i{}' \frac{p_{ij} w_\alpha(p_{ij})}{h w_\alpha(h)} \tag{3.2}$$

where the inner sum is taken over all i such that $p_{ij} \neq 0$.

Most of the properties of these indices are easily derived and may be found in [16]. They are:

(i) $\Gamma_0 = -T_r A$,

(ii) Γ_1 = number of arcs in the digraph,

(iii) $\Gamma_\alpha = \sum_j \sum_{i\neq j} a_{ij}^{(1-\alpha)}$, $0<\alpha<1$,

(iv) $\Gamma_\alpha \leqq \Gamma_0^{1-\alpha}\Gamma_1^\alpha$, $0<\alpha<1$.

(v) $\Gamma_\alpha \leqq -m^{-\alpha}\sum \lambda_i$ where $\{\lambda_i\}$ are the eigenvalues of A and m is the minimum nonzero flow rate.

3.1. Two reduction algorithms [15] [16]. It is possible to simplify a given compartmental model by reducing it to a mammillary system in a number of ways. We mention two of them here, one of which leaves Γ_0 invariant and the other of which leaves the equilibrium solution X_∞ invariant. We shall assume that the digraph of our model is in the form of an *advanced rosette* initially, i.e. is strongly connected and has a central vertex lying on all cycles (simple closed paths). Most strongly connected ecosystem models have this form. See Fig. 4.

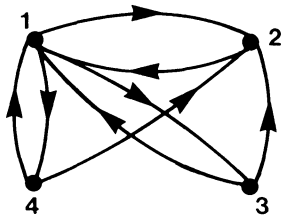


FIG. 4. *An advanced rosette digraph.*

We number the vertices such that the central vertex is numbered 1 and the others follow in such a way that the matrix A has the form

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & a_{24} & \cdots & a_{2n} \\ a_{31} & 0 & a_{33} & a_{34} & \cdots & a_{3n} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{n1} & 0 & 0 & 0 & \cdots & a_{nn} \end{bmatrix}. \tag{3.3}$$

That is, if we cross off the first row and column, the remaining matrix is upper triangular.

DEFINITION 3.3. Let A be the matrix of a compartmental model whose digraph is an advanced rosette. Let the digraph be modified by redirecting the arc (i, j) from j to $1, 1 \neq i \neq j \neq 1$. If B is the matrix corresponding to a sequence of such modifications, then B is a 0-*reduction* of A (or of the digraph or of the compartmental model).

For example, Fig. 4 may be changed to Fig. 5 by two such modifications.

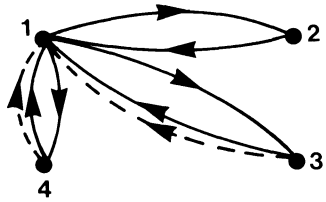


FIG. 5. *Modification to Fig.* 4 *by redirecting arc* (3, 2) *to* (3, 1) *and* (4, 2) *to* (4, 1).

The effect on the matrix A is to replace a_{ji}, $j < i$ by 0 and to add a_{ji} to a_{1i} in the latter's position. Clearly the trace of A is invariant under this procedure and hence Γ_0 remains the same.

The other reduction procedure involves the augmented matrix used to find the equilibrium solution whose (l') length is 1:

$$A_a = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 & \cdot & 1 \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} & \cdot & 0 \\ a_{31} & 0 & a_{33} & \cdots & a_{3n} & \cdot & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdot & \cdots \\ a_{n1} & 0 & 0 & \cdots & a_{nn} & \cdot & 0 \end{bmatrix}. \tag{3.4}$$

The effect of a single 0-reduction on A_a is to replace element a_{ij}, $1 < i < j$ by 0.

DEFINITION 3.4. An *∞-reduction* of A_a (or of the digraph or the compartmental model) is a matrix B_a obtained from A_a by sequence of elementary row operations each of which eliminates an element a_{ij}, $1 < i < j \leqq n$.

This reduction corresponds to the same redirection of an arc of the digraph with the additional change of a flow rate from vertex 1 to the vertex from which the arc was removed.

The ultimate form obtained by either reduction is an advanced rosette all of whose cycles are of length 2. Such a digraph corresponds to a mammillary system. The digraph of Fig. 5 corresponds to such a system. In both types of reduction, we obtain a matrix of the form

$$B_a = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 & \cdot & 1 \\ a'_{21} & a_{22} & 0 & \cdots & 0 & \cdot & 0 \\ a'_{31} & 0 & a_{33} & \cdots & 0 & \cdot & 0 \\ \cdots & \cdots & \cdots & \cdots & 0 & \cdot & 0 \\ a_{n1} & 0 & 0 & \cdots & a_{nn} & \cdot & 0 \end{bmatrix} \tag{3.5}$$

where the first column changes for ∞-reduction but not for 0-reduction. Clearly the equilibrium solution is invariant under ∞-reduction. We summarize in

PROPOSITION 3.1. *Let B_a be obtained from A_a by a 0-reduction and C_a by an ∞-reduction. Let w_1 be the first component of the equilibrium solution. Then*

(i) $w_1(B_a) \leqq w_1(A_a) = w_1(C_a)$,

(ii) $\Gamma_0(B_a) = \Gamma_0(A_a) \leqq \Gamma_0(C_a)$,

(iii) $\Gamma_1(B_a) \leqq \Gamma_1(A_0) \geqq \Gamma_1(C_a)$,

with equality holding in each case only if A_a is the matrix of mammillary system.

Another operation on the system is a *redistribution* which consists of transferring part of a higher flow rate to a lower rate from the same vertex. That is, if $a_{ij} < a_{kj}$, it replaces the former by $a_{ij} + h$ and the latter by $a_{kj} - h$ for h sufficiently small.

PROPOSITION 3.2. *Let B be obtained from A by a sequence of redistributions. Then*

$$\Gamma_\alpha(B) \geqq \Gamma_\alpha(A)$$

for any $0 \leqq \alpha \leqq 1$.

4. Stability indices. All of our compartmental models are stable as we observed earlier. However some will recover more rapidly from a perturbation than others. A stability index should measure this rapidity in some way, but should be independent of the initial values X_0. We shall restrict ourselves to strongly connected models and consider three indices:

$$\text{(i)} \qquad \sigma_\alpha = \sum_{i=2}^{n} \alpha_i \lambda_i, \quad \alpha_i > 0, \quad \sum \alpha_i = 1,$$

$$\text{(ii)} \qquad \rho^{-1} = \lim_{h \to 0} hT_r \sum_{t=1}^{\infty} (P^t - W), \tag{4.1}$$

$$\text{(iii)} \qquad m = w_1^{-1} = \text{mean first passage time.}$$

The first is merely a convex combination of the nonzero eigenvalues $\lambda_2, \lambda_3, \cdots, \lambda_n$. The second, a resilience index, is based on the total deviation from equilibrium throughout the history of the regular Markov chain:

$$R = \sum_{t=1}^{\infty} (P^t - W). \tag{4.2}$$

Here W is the matrix all of whose columns are X_∞ normalized to length 1, and $P = I + hA$ is the transition matrix. The index ρ combines the elements of R and gets rid of the time step by taking the limit.

The third index, the mean first passage from a central compartment as in an advanced rosette, back to itself again is also related to R ([6, p. 79]). Indeed,

$$M = (ER_d + E - R)W_d^{-1} \tag{4.3}$$

is the matrix of mean first passage times. Here R_d and W_d are the diagonal matrices which agree with R and W on the main diagonal and E is composed of 1's. The index m is just the element in the upper left-hand corner of M.

It can be shown [12] that

$$\rho = \left(-\sum_{i=2}^{n} \lambda_i^{-1} \right)^{-1}. \tag{4.4}$$

The index m, on the other hand, does not have such a simple relation to the eigenvalues. For an advanced rosette it is given by [14]

$$m = \prod_{i=2}^{n} \lambda_i / a_{ii}. \tag{4.5}$$

It can also be expressed in units of the turnover time of the first compartment. In these units m is exactly the mean residence time [2], [4], [15].

5. Complexity vs. stability. If m is taken as the measure of stability, then m is invariant under the ∞-reduction algorithm by Proposition 3.1. However Γ_0 decreases. Similarly if the 0-reduction algorithm is used, m increases but Γ_0 remains the same. Hence it appears that *complexity and stability are independent* if the former is interpreted as m and the latter as Γ_0.

If σ_β is taken to be the measure of stability, then for each Γ_α we have

$$\Gamma_\alpha \leqq (-\sum \lambda_i)^{1-\alpha} N^\alpha = \left(-\sum \frac{\lambda_i \beta_i}{\beta^i} \right)^{1-\alpha} N^\alpha \leqq \frac{N^\alpha}{(\min \beta_i)^{1-\alpha}} \sigma_\beta. \tag{5.1}$$

Hence *greater complexity leads to greater stability* under this interpretation.

The same is true for the index ρ provided the spread of the eigenvalues is not too great. Both Γ_0 and Γ_1 are proportional to minus the sum of the eigenvalues and hence the greatest contribution to them is from the eigenvalue with the largest negative real part. ρ^{-1} on the other hand depends primarily on the eigenvalue with smallest negative real part. Thus Γ_0 or Γ_1 and ρ vary together provided the ratio between the smallest and largest eigenvalue does not change. However it can change considerably for models with the same complexity index. Consider the two digraphs (weighted) of Fig. 6. Both have the same complexity indices and are advanced rosettes. Their matrices are

$$A = \begin{bmatrix} -2 & 1 & 3 \\ 0 & -1 & 1 \\ 2 & 0 & -4 \end{bmatrix}, \qquad B = \begin{bmatrix} -2 & 1 & 1 \\ 0 & -1 & 3 \\ 2 & 0 & -4 \end{bmatrix}.$$

Their nonzero eigenvalues are respectively

$$\lambda = -\frac{7}{2} \pm \frac{\sqrt{17}}{2} \text{ for } A \quad \text{and} \quad \lambda = -\frac{7}{2} \pm \frac{1}{2} \text{ for } B.$$

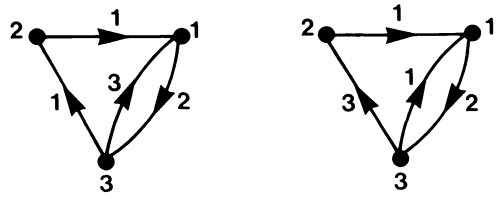


FIG. 6. *Two models with the same complexity indices but different eigenvalues.*

Hence the spread of the eigenvalues of A is $\sqrt{17}$ while that of B is 1. The resilience indices are

$$\rho(A) = \tfrac{8}{7}, \qquad \rho(B) = \tfrac{12}{7}.$$

Thus no conclusion in general about the relation between complexity and ρ is possible. What is needed is some structural criterion for the spread of the eigenvalues. This appears as yet not to have been done.

REFERENCES

[1] J. EISENFELD, *Relationship between stochastic and differential models of compartmental systems*, Math. Biosci., 43 (1979), pp. 289–305.
[2] ———, *On mean residence times in compartments*, Math. Biosci., 57 (1981), pp. 265–278.
[3] J. Z. HEARON, *Theorems on linear systems*, Ann. NY Acad. Sci., 108 (1963), pp. 36–91.
[4] ———, *Residence times in compartmental systems and the moments of a certain distribution*, Math. Biosci., 15 (1972), pp. 69–77.
[5] J. A. JACQUEZ, *Compartmental Analysis in Biology and Medicine*, Elsevier, Amsterdam, 1972.
[6] J. J. KEMENY AND J. L. SNELL, *Finite Markov Chains*, Van Nostrand, Princeton, NJ, 1960.
[7] M. MARCUS AND H. MINC, *A Survey of Matrix Theory and Matrix Inequalities*, Prindle, Weber and Schmidt, Boston, 1964.
[8] R. M. MAY, *Stability and Complexity in Model Ecosystems*, Princeton Univ. Press, Princeton, NJ, 1973.
[9] FRED S. ROBERTS, *Discrete Mathematical Models*, Prentice-Hall, Englewood Cliffs, NJ, 1976.
[10] C. TAILLIE, *The mathematical statistics of diversity and abundance*, Ph.D. Thesis, Penn State Univ., University Park, PA, 1977.
[11] G. G. WALTER, *Compartmental models, digraphs, and Markov chains*, in Compartmental Analysis of Ecosystem Models, Matis, Patten & White, eds., Intern. Co-op Pub. Fairland, MD, 1979, pp. 295–310.
[12] ———, *Stability and structure of compartmental models of ecosystems*, Math. Biosci., 51 (1980), pp. 1–10.
[13] ———, *Passage time, resilience, and structure of compartmental models*, Math. Biosci., 63 (1983), pp. 199–213.
[14] ———, *Stability in compartmental models*, in Population Biology, Freedman and Strobeck, eds., Springer-Verlag, Berlin, 1983, pp. 372–578.
[15] ———, *Some equivalent compartmental models*, Math. Biosci., 64 (1983), pp. 273–293.
[16] ———, *Complexity of compartmental models*, submitted.
[17] ———, *A compartmental model of a marine ecosystem*, in Compartmental Analysis of Ecosystem Models, Matis, Pattern and White, eds., Int. Co-op Pub., Fairland, MD, 1979, pp. 29–42.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 1, January 1985

NONNEGATIVE SOLUTIONS OF A QUADRATIC MATRIX EQUATION ARISING FROM COMPARISON THEOREMS IN ORDINARY DIFFERENTIAL EQUATIONS*

G. J. BUTLER†, CHARLES R. JOHNSON‡ AND H. WOLKOWICZ§

Abstract. We study the quadratic matrix equation

$$X^2+\beta X+\gamma A=0,$$

where A is a given elementwise nonnegative (resp. positive semi-definite) matrix and the solution X is required to be an elementwise nonnegative (resp. positive semi-definite) matrix. When $\beta=-1$ and $\gamma=1$, our results may be used, for example, to obtain a simple nonoscillation criterion for the matrix differential equation

$$Y''(t)+Q(t)Y(t)=0,$$

where Y and Q are matrix-valued functions and denotes differentiation. This generalizes a result of Hille for the scalar case. Extensions are given when A and X are nonnegative with respect to more general cone orderings.

AMS(MOS) subject classification. 15A24

1. Introduction. In this paper we characterize the existence of solutions of the quadratic matrix equation

$$X^2+\beta X+\gamma A=0, \tag{1.1}$$

where γ and β are given real scalars and A is a given "nonnegative" $n\times n$ matrix. We first consider the case when $\gamma>0$, $\beta<0$ and A is either Hermitian positive semi-definite or elementwise nonnegative. The solution X is then restricted to be Hermitian or elementwise nonnegative, respectively. In these cases we completely characterize the existence of a solution in terms of the spectrum of A; see § 2.

In § 3 we use the notion of a positivity cone K, see [9], to unify and extend the results of § 2. Thus, in the case that $\gamma>0$, we characterize the existence of nonnegative or M-matrix (with respect to K) solutions of (1.1) when A is nonnegative (with respect to K).

The problem of the existence of solutions of (1.1) arises in the context of comparison theorems for two matrix-valued ordinary differential equations. Consider the equation

$$Y''(t)+Q(t)Y(t)=0. \tag{1.2}$$

Here Y and Q are continuous $n\times n$ matrix-valued functions and $'$ denotes differentiation. Such equations arise both in the self-adjoint case (in the study of Hamiltonian

* Received by the editors February 22, 1983, and in revised form October 13, 1983.

† Department of Mathematics, University of Alberta, Edmonton, Alberta, Canada T6G 2G1. The research of this author was supported in part by the Natural Sciences and Engineering Research Council of Canada under grant A-8130.

‡ Institute for Physical Science and Technology, University of Maryland, College Park, Maryland 20742. The work of this author was supported by Air Force Wright Aeronautical Laboratories contract F-33615-81-K-3224.

§ Department of Mathematics, University of Alberta, Edmonton, Alberta, Canada T6G 2G1. The research of this author was supported in part by the Natural Sciences and Engineering Research Council of Canada under grant A-3388. Some of this research was carried out while this author was on leave at the Institute for Physical Science and Technology, University of Maryland, College Park, Maryland 20742.

systems, for example [7], [8]) and in the nonself-adjoint case [1], [5]. See also the references in [5]. A solution $Y(t)$ of (1.2) is said to be nonoscillatory if for some t_0 it is nonsingular for all $t \geqq t_0$. In that case we may form the so-called Riccati equation

$$Z'(t)+Z^2(t)+Q(t)=0, \qquad t \geqq t_0 \tag{1.3}$$

where $Z(t)=Y'(t)Y^{-1}(t)$.

Of interest are comparison theorems between two equations of the form (1.2) with different coefficients. Thus we consider also the equations

$$Y''(t)+Q_1(t)Y(t)=0, \tag{1.2}_1$$

$$Z'(t)+Z^2(t)+Q_1(t)=0. \tag{1.3}_1$$

In the scalar case ($n=1$), the classical Sturm comparison theorem yields the result that if (1.2) has a nonoscillatory solution (and therefore (1.3) has a solution on some interval $[t_0, \infty)$) and if $Q(t) \geqq Q_1(t)$ for all t, then $(1.2)_1$ will have a nonoscillatory solution (and $(1.3)_1$ will have a solution on $[t_0, \infty)$). There are many other comparison theorems in the scalar case (see [12], for example).

The extension of comparison theorems to the general matrix case requires some kind of ordering on the coefficient matrices $Q(t)$, $Q_1(t)$; hence some form of positivity must be defined. Positive semi-definite is the appropriate concept for studying self-adjoint equations; positive cone versions of positivity are a suitable choice for nonself-adjoint equations.

The idea behind comparison theorems is that the oscillatory or nonoscillatory character of an equation $(1.2)_1$ may be determined by comparison with some equation (1.2) whose behavior is known.

Here we shall confine ourselves to obtaining a simple nonoscillation criterion for $(1.2)_1$, which is a generalization of a well-known result of Hille [10] in the scalar case.

Suppose that

$$P(t)=\lim_{T\to\infty}\int_t^T Q(s)\,ds$$

and

$$P_1(t)=\lim_{T\to\infty}\int_t^T Q_1(s)\,ds$$

both exist, and are finite, and that

$$P(t) \geqq |P_1(t)| \geqq 0 \quad \text{for all } t, \tag{1.4}$$

in the sense that $P(t)-|P_1(t)|$ has nonnegative elements, and where $|P_1(t)|$ is the matrix whose elements are the absolute values of those of $P_1(t)$.

Under these assumptions, it was shown in [5] that if (1.3) has a positive solution $Z(t)$ on $[t_0, \infty)$, then $(1.3)_1$ has a positive solution $Z_1(t)$, where $0 \leqq Z_1(t) \leqq Z(t)$, $t \geqq t_0$. (This is a generalization of the Hille–Wintner theorem in the scalar case [10], [12]).

To apply this result, we look for a suitable candidate for $Q(t)$.

If $Q(t)=t^{-2}A$, where A is a constant $n \times n$ matrix, we can try to find a solution of (1.3) of the form $Z(t)=t^{-1}X$, where X is a constant $n \times n$ matrix. This leads to the quadratic matrix equation

$$X^2-X+A=0. \tag{1.5}$$

To use the comparison theorem quoted above we require that A and X are positive.

Then the solvability of (1.5) reduces to that of (1.1) with $\beta=-1$, $\gamma=1$. Let $\rho(A)$ be the spectral radius of A. Theorem 2.3 of § 2 will show that (1.5) has a nonnegative solution X if and only if

$$\text{(1.6)} \qquad \begin{array}{l}\rho(A)<\frac{1}{4}, \text{ or } \rho(A)=\frac{1}{4} \text{ and the eigenvalues}\\ \text{of } A \text{ which have modulus } \frac{1}{4} \text{ have degree equal to 1,}\end{array}$$

where the degree is the size of the largest Jordan block. Denoting the set of nonnegative matrices A satisfying (1.6) by $\mathscr{A}$, we have:

THEOREM 1.1. *Let $Q_1(t)$ be continuous, such that*

$$t\left|\int_t^\infty Q_1(s)\,ds\right| \leqq A$$

for all sufficiently large t, for some $A \in \mathscr{A}$.

Then $(1.2)_1$ has a nonoscillatory solution Y_1 whose associated Riccati variable Z_1 satisfies $|Z_1(t)| \leqq t^{-1}X$, t sufficiently large, where X is the unique positive solution of (1.5).

In the scalar case, A can be any constant $\leqq\frac{1}{4}$, and we have Hille's result.

2. Existence of solutions. By using the substitution $X=-\beta Y$, we may consider the equation

$$\text{(2.1)} \qquad X^2-X+A=0$$

rather than (1.1), and this we choose to do.

We answer the following two questions concerning existence of solutions:

1. A is given Hermitian, positive semi-definite (psd) and we require X to be Hermitian;

2. A is given real and nonnegative (elementwise) and we require X to be real and nonnegative.

The Hermitian case essentially reduces to a scalar problem, and we have:

THEOREM 2.1. *Suppose that A is a given Hermitian matrix. Then* (2.1) *has a Hermitian solution X if and only if*

$$\text{(2.2)} \qquad \sigma(A) \subset (-\infty, \tfrac{1}{4}]$$

where $\sigma(A)$ is the spectrum of A.

Proof. Since $A=X-X^2$ is a polynomial in X, A commutes with any solution X and so A and X can be simultaneously diagonalized by some unitary matrix U. Thus X is a Hermitian solution of (2.1) if and only if

$$\text{(2.3)} \qquad D^2-D+\Lambda=0$$

has a solution, where $D=UXU^*$ and $\Lambda=UAU^*$ are the diagonal matrices of eigenvalues of X and A, respectively. Thus the diagonal elements satisfy

$$d_i^2-d_i+\lambda_i=0, \qquad i=1,\cdots,n.$$

Since $d_i=\frac{1}{2}(1\pm\sqrt{1-4\lambda_i})$ is real if and only if $1-4\lambda_i \geqq 0$, the result follows.

COROLLARY 2.1. *Let A be psd. Then* (2.1) *has a Hermitian solution X if and only if $\sigma(A) \subset [0, \frac{1}{4}]$, and in this case $\sigma(X) \subset [0, 1]$, i.e. all Hermitian solutions are psd.*

Proof. The result follows since we need $1+\sqrt{1-4\lambda_i} \geqq 0$ for all i.

Now we consider the case that $0 \neq A \geqq 0$ elementwise, and we seek $X \geqq 0$ (elementwise) to solve (2.1). The solution of this problem again rests upon the spectrum of A.

If X solves (2.1), then

$$0 = X^2 - X + A = (X - \tfrac{1}{2}I)^2 - \tfrac{1}{4}I + A,$$

so

$$X = \tfrac{1}{2}(I \pm S), \tag{2.4}$$

where

$$S = (I - 4A)^{1/2}. \tag{2.5}$$

If S should admit a series expansion, then

$$X = \frac{1}{2}I \pm \frac{1}{2}\sum_{i=0}^{\infty}(-1)^i\binom{\frac{1}{2}}{i}(4A)^i, \tag{2.6}$$

so

$$X = -\frac{1}{2}\sum_{i=1}^{\infty}(-1)^i\binom{\frac{1}{2}}{i}(4A)^i, \tag{2.7}$$

choosing the negative sign in (2.6), so that $X \geqq 0$. This series will converge if $4\rho < 1$ and diverge if $4\rho > 1$.

Now consider the following iterative scheme:

$$X_1 = A, \qquad X_{n+1} = A + X_n^2, \quad n = 1, 2, \cdots. \tag{2.8}$$

If X_n converges to X as $n \to \infty$, we shall have $X = A + X^2$; clearly $X \geqq 0$, and so will be a nonnegative solution of (2.1). The iterative scheme has the following properties.

LEMMA 2.1. *Suppose that $X \geqq 0$ solves* (2.1). *Then the sequence of iterates in* (2.8) *satisfies*

$$0 \leqq X_n \leqq X_{n+1} \leqq X, \qquad n = 1, 2, \cdots, \tag{2.9}$$

and

$$S_n \leqq X_n \leqq S_{2^{n-1}}, \qquad n = 1, 2, \cdots, \tag{2.10}$$

where S_k denotes the partial sum of degree k of the series in (2.7).

Proof. $X_1 = A \leqq A + X^2 = X$, and

$$X_1 = A \leqq A + A^2 = X_2 = A + X_1^2 \leqq A + X^2 = X,$$

i.e. (2.9) holds for $n = 1$. Assume that (2.9) holds for a particular value of n. Then

$$X_{n+1} - X_n = (X_n^2 - X_{n-1}^2) \geqq 0,$$

and similarly, $X - X_{n+1} \geqq 0$. Thus (2.9) follows by induction.

To obtain (2.10), observe that the power series X defined by (2.7) formally satisfies

$$X = X^2 + A. \tag{2.11}$$

Denote the partial sum of degree k of the formal series for X^2 by T_k. Since X has no constant term, formally squaring the power series shows that $T_{n+1} \leqq S_n^2$, $n = 1, 2, \cdots$.

From (2.11), we have $S_{n+1} = T_{n+1} + A$, and so

$$S_{n+1} \leqq S_n^2 + A, \qquad n = 1, 2, \cdots. \tag{2.12}$$

Again, we see that $T_{2^n} \geqq S_{2^{n-1}}^2$, and so

$$S_{2^n} \geqq S_{2^{n-1}}^2 + A, \qquad n = 1, 2, \cdots. \tag{2.13}$$

Since $S_1 = S_{2^0} = X_1 = A$, a simple induction argument with (2.12) and (2.13) gives (2.10), which completes the proof of the lemma.

In fact, by considering the case when A is a scalar, we see that the infinite series, obtained by expanding the iteration (2.8), must be the same as (2.7).

Now we can obtain the following existence result.

THEOREM 2.3. (i) $4\rho < 1$ *implies that there is a nonnegative solution to* (2.1).

(ii) $4\rho > 1$ *implies that there is no nonnegative solution to* (2.1).

(iii) *If* $4\rho = 1$, *then* (2.1) *has a nonnegative solution if and only if the eigenvalues of* A *which are equal to the spectral radius in modulus, have degree* 1, *that is,*

$$|\lambda_i| = \rho \Rightarrow \lambda_i \text{ has degree } 1. \tag{2.14}$$

Proof. If $4\rho < 1$, the nonnegative solution X is given explicitly by (2.7).

Now suppose that $4\rho > 1$ and that $X \geqq 0$ is a solution of (2.1). By (2.9) of Lemma 2.1, the iterates of (2.8) are monotone increasing and bounded above by X. Without loss of generality, we may assume that $X_n \to X$, X a positive solution of (2.1). But then (2.10) of Lemma 2.1 shows that X satisfies (2.6), which will be a divergent power series when $4\rho < 1$. This is a contradiction and gives (ii).

Finally suppose that $4\rho = 1$. Suppose that (2.14) holds, and let

$$A = PJP^{-1} \tag{2.15}$$

where J is the Jordan canonical form of A. Convergence of the power series in (2.7) depends only on the individual blocks of J. Since these blocks have spectral radius less than or equal to $\frac{1}{4}$, with equality only if they have degree 1, the power series converges and yields a nonnegative solution to (2.1).

Conversely, suppose that $X \geqq 0$ is a solution of (2.1) and that (2.14) fails to hold. First assume that there is exactly one defective Jordan block corresponding to an eigenvalue equal to ρ. X satisfies (2.4) and S satisfies (2.5). This contradicts the criterion in [2] for the existence of a square root of a singular matrix, which states that the defective Jordan blocks must come in pairs. This then implies that the series in (2.7) diverges if J is replaced by a single defective Jordan block $\tilde{J}$. Since the convergence of the series in (2.7) depends only on the individual Jordan blocks, it follows that A cannot have any defective blocks corresponding to an eigenvalue equal to ρ. (We have already seen that the existence of a positive solution of (2.1) implies convergence of the series in (2.7) as the limit of the iterates X_n of (2.8).)

The result now follows, since $|\lambda_i| = \rho$ implies that the degree of λ_i, i.e. the size of the largest block in the Jordan canonical form of A that contains λ_i, is not larger than the degree of the eigenvalue equal to ρ, see e.g. [6]. Thus there can be no defective blocks, and (2.14) must hold.

The above results are related to the notion of an M-matrix. Recall that A is an M-matrix if $A = rI - P$, where $P \geqq 0$ and $\rho(P) \leqq r$. If $\rho(P) = r$, then A is a singular M-matrix. Note that if A is an M-matrix then A has the Z-matrix sign pattern, i.e. $a_{ij} \leqq 0$ if $i \neq j$. If A is an invertible M-matrix, then $A^{-1} \geqq 0$ and moreover, A has a square root $A^{1/2}$ which is also an M-matrix. See e.g. [3]. The M-matrix property arises in (2.5), for if $4\rho < 1$, then S^2 is an invertible M-matrix and so has a square root S which is also an M-matrix. This implies that $X = \frac{1}{2}(-\beta I + S) \geqq 0$. Our proofs yield the following for singular M-matrices.

COROLLARY 2.1. *The (singular)* M*-matrix* $\rho I - A$ *has a square root if and only if* (2.4) *holds.*

The series (2.6) yields two solutions to (2.1). Choosing the negative sign yields

$$X_1 = -\frac{1}{2}\left(\sum_{i=1}^{\infty} (-1)^i \binom{\frac{1}{2}}{i} (4A)^i\right) \geqq 0.$$

The second solution is

$$X_2 = -I + \frac{1}{2}\left(\sum_{i=1}^{\infty} (-1)^i \binom{\frac{1}{2}}{i} (4A)^i\right).$$

Thus $X_2 = I - P$, where $P \geqq 0$, and so is a Z-matrix. But, if $\rho < \frac{1}{4}$, then $\rho(P) < 1$ which implies that X_2 is in fact an M-matrix. The case $\rho = \frac{1}{4}$ is similar. In fact, we have a nonnegative solution if and only if we have an M-matrix solution. For if X is an M-matrix solution, then $X = \frac{1}{2}(I - S)$ with $\rho(S) \leqq 1$, see (2.4). But then $\frac{1}{2}(I - S)$ is a nonnegative solution.

3. Extension to positivity cones. The notion of a positivity cone was introduced in [9] to give a unified treatment of results on M-matrices and positive definite matrices. We now extend our results to such cones. Following [9], we define $\mathbf{K}$ to be a positivity cone of matrices if $\mathbf{K}$ is a pointed, closed, convex cone, i.e. if $K \cap -K = \{0\}$, $\mathbf{K} + \mathbf{K} \subset \mathbf{K}$ and $\lambda \mathbf{K} \subset \mathbf{K}$, for all $\lambda \geqq 0$, and if

$$P \in \mathbf{K} \text{ implies } P^i \in \mathbf{K}, \qquad i = 0, 1, 2, \cdots. \tag{3.1}$$

The cones $\mathbf{K}_1$, of all nonnegative (elementwise) matrices, and $\mathbf{K}_2$, the cone of positive semi-definite Hermitian matrices to which we addressed ourselves in § 2, are examples of positivity cones, as is $\mathbf{K}_1 \cap \mathbf{K}_2$. Additional examples are given in [9].

We let $\mathbf{K}$ denote a positivity cone and partially order $\mathbf{C}^{nn}$ with respect to $\mathbf{K}$, i.e. $P \geqq 0$ if $P \in \mathbf{K}$. Associated with $\mathbf{K}$ are the sets

$$\mathbf{Z} = \{A \in \mathbf{C}^{nn} \colon A = sI - P, s \in R, P \in K\}, \tag{3.2}$$

$$\mathbf{M} = \{A \in \mathbf{Z} \colon \operatorname{Re} \lambda \geqq 0, \text{ for all eigenvalues } \lambda \text{ of } A\}. \tag{3.3}$$

Corresponding to $\mathbf{K}_1$ and $\mathbf{K}_2$ above, $\mathbf{Z} = \mathbf{Z}_1$ is the set of Z-matrices, $\mathbf{M} = \mathbf{M}_1$ is the set of M-matrices, $\mathbf{Z} = \mathbf{Z}_2$ is the set of Hermitian matrices and $\mathbf{M} = \mathbf{M}_2$ is the set of positive semi-definite matrices.

We would like to unify our results from § 2 as well as extend them to general positivity cones. We shall require the series solution defined by (2.6) and a result corresponding to Lemma 2.1 concerning the iterative scheme (2.8). For the lemma to hold in the new partial order, we need an additional condition, (3.4) below.

LEMMA 3.1. *Lemma* 2.1 *holds if the partial order induced by a positivity cone* $\mathbf{K}$ *is closed under commuting products, i.e.* $\mathbf{K}$ *satisfies*

$$B_1, B_2 \in \mathbf{K}, \qquad B_1 B_2 = B_2 B_1 \Rightarrow B_1 B_2 \in \mathbf{K}. \tag{3.4}$$

(*this is condition* (2.4) *in* [9]).

Proof. Since $A \in \mathbf{K}$ and (3.1) holds for a positivity cone, it follows inductively that the iterates X_n of (2.8) are in $\mathbf{K}$ and are polynomials in A with nonnegative coefficients. Thus we have

$$X_{n+1} - X_n = (X_n^2 - X_{n-1}^2) = (X_n - X_{n-1})(X_n + X_{n-1}), \tag{3.5}$$

since the two factors on the right-hand side commute. It follows inductively from (3.5) that $0 \leqq X_n \leqq X_{n+1}$, $n = 1, 2, \cdots$.

Now suppose that $X \geqq 0$ solves (2.1). Then $X = X^2 + A$, so

$$X^2 = X^3 + AX = X^3 + XA,$$

so X commutes with A. Since the X_n are polynomials in A, it follows that X commutes with each X_n. It is now easy to show that $X_n \leqq X$ for all n, and we have (2.9) of Lemma 2.1.

The proof of (2.10) proceeds as before.

We remark that $\mathbf{K}_1$ and $\mathbf{K}_2$ are both positivity cones that satisfy (3.4).

Next we prove the following result which includes a generalization of Theorem 2.3 to positivity cones satisfying (3.4).

THEOREM 3.1. *Let* $\mathbf{K}$ *be a positivity cone satisfying* (3.4) *and let* $A \geqq 0$ (*with respect to* $\mathbf{K}$). *Then* (2.1) *has a solution* $X \in \mathbf{K}$ *if and only if*

$$4\rho \leqq 1, \tag{3.6}$$

with (2.13) *holding if* $4\rho = 1$.

Proof. If $4\rho < 1$, then the series in (2.7) converges to X, which is a solution to (2.1). From the definition of the positivity cone, $\sum_{i=0}^{\infty} (-1)^i \binom{1/2}{i} (4A)^i \in -\mathbf{K}$. Thus $X \geqq 0$. If $4\rho = 1$ and (2.3) holds, then we still obtain convergence. (See the argument in the proof of Theorem 2.3.) Conversely, suppose that X solves (2.1) and $X \geqq 0$. To complete the proof we need only show that the existence of a solution $X \geqq 0$ of (2.1) implies that the series in (2.7) converges. First we show that the order interval $[0, X] = \{Y: 0 \leqq Y \leqq X\}$ is compact. Suppose not. Then there is a sequence $\{Y_n\} \subset [0, X]$ with $\| Y_n \| \to \infty$. We may assume that $Y_n / \| Y_n \| \to Y \in \mathbf{K}$. But then $(X - Y_n)/\| Y_n \| \in \mathbf{K}$, and upon taking the limit as $n \to \infty$, we find that $-Y \in \mathbf{K}$, a contradiction, since $\mathbf{K}$ is pointed. It follows that $[0, X]$ is compact. Using Lemma 3.1, we deduce that $X_n \to Y$, a solution of (2.1), which implies that the series in (2.7) converges.

Note that an M-matrix solution (with respect to $\mathbf{K}$) is obtained by using the positive sign in the expansion (2.6).

REFERENCES

[1] S. AHMAD AND A. C. LAZER, *An n-dimensional extension of the Sturm separation and comparison theory to a class of nonselfadjoint systems*, SIAM J. Math. Anal., 9 (1978), pp. 1137–1150.

[2] G. ALEFELD AND N. SCHNEIDER, *On square roots of M-matrices*, Linear Algebra and Appl., 42 (1982), pp. 119–132.

[3] A. BERMAN AND R. J. PLEMMONS, *Nonnegative Matrices in the Mathematical Sciences*, Academic Press, New York, 1979.

[4] J. M. BORWEIN AND B. RICHMOND, *When does a matrix have a root?*, Preprint, Dalhousie University.

[5] G. J. BUTLER AND L. H. ERBE, *Comparison theorems for second-order operator-valued linear differential equations*, Pacific J. Math., to appear.

[6] G. W. CROSS AND P. LANCASTER, *Square roots of complex matrices*, Linear and Multilinear Algebra, 1 (1974), pp. 289–293.

[7] G. J. ETGEN AND R. T. LEWIS, *Positive functionals and oscillation criteria for second-order differential systems*, Proc. Edinburgh Math. Soc., 22 (1979), pp. 277–290.

[8] G. J. ETGEN AND J. F. PAWLOWSKI, *Oscillation criteria for second-order selfadjoint differential systems*, Pacific J. Math., 66 (1976), pp. 99–110.

[9] M. FIEDLER AND H. SCHNEIDER, *Analytic functions of M-matrices and generalizations*, Linear and Multilinear Algebra, to appear.

[10] E. HILLE, *Nonoscillation theorems*, Trans. Amer. Math. Soc., 64 (1948), pp. 234–252.

[11] C. R. JOHNSON, *Inverses of M-matrices*, Linear Algebra and Appl., to appear.

[12] C. A. SWANSON, *Comparison and Oscillation Theory of Linear Differential Equations*, Academic Press, New York, 1968.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 1, January 1985

006

ON KERNELS OF GRAPHS AND SOLUTIONS OF GAMES: A SYNOPSIS BASED ON RELATIONS AND FIXPOINTS*

GUNTHER SCHMIDT† AND THOMAS STRÖHLEIN†

Dedicated to F. L. Bauer on the occasion of his 60th birthday.

Abstract. We aim at a uniform approach to results concerning the existence of kernels of graphs and introduce new results in the bipartite case. The Galois connection based on the function which assigns to a vertex set the set of its nonpredecessors is investigated using a special fixpoint theorem; it is illustrated by the notions of retardation and expansiveness. The related topic of solutions of games is mentioned, and an analysis of some chess endings is included as an application. The paper contains an extended bibliography.

Key words. kernel of a graph, solution of a game, relational algebra, lattices and fixpoints, chess endings

AMS subject classifications. 68E10, 05C20, 05C50, 04A05, 06A15, 90D05, 90D45

1. Introduction. Research on kernels originated from the theory of games and economic behaviour. Among those who contributed in the early years were Zermelo, König, Kalmár, Max Euwe, former world chess champion, and von Neumann.

Our approach is to introduce relational algebra into the study of kernels and to apply the by now well developed theory of lattice antimorphisms. As our first tool we recall basic concepts of relational algebra which are easily understood if they are interpreted in terms of Boolean $(n \times n)$-matrices. The basis is formed by a complete atomistic Boolean algebra with respect to $\vee$ (join), $\wedge$ (meet), $\bar{}$ (complement) and $\subset$ (inclusion). If we additionally define composition of relations (Boolean matrix multiplication), identity relation I, transposition ^{T}, zero relation 0, and universal relation L, we arrive at a relational algebra. In the sequel we proceed by algebraic methods, i.e., we rely on lattice formulas and identities for relations like

$$P(Q \vee R) = PQ \vee PR \qquad (\vee\text{-distributivity}),$$

$$P(Q \wedge R) \subset PQ \wedge PR \qquad (\wedge\text{-subdistributivity}),$$

$$PQ \subset R \Leftrightarrow P^T\bar{R} \subset \bar{Q} \qquad (\text{Schröder rule}),$$

$$PQ \wedge R \subset (P \wedge RQ^T)(Q \wedge P^TR) \qquad (\text{Dedekind rule}),$$

$$R \neq 0 \Rightarrow LRL = L \qquad (\text{Tarski rule}).$$

The transitive closure of a relation R is defined as the union of its powers $R^+ := R \vee R^2 \vee R^3 \vee \cdots$, whereas $R^* := I \vee R^+$ is called the reflexive transitive closure. Familiar notions are symmetry $(R \subset R^T)$, irreflexivity $(R \subset \bar{I})$ and transitivity $(R^2 \subset R$ or $R^+ \subset R)$.

Now we pass to graph terminology: An arbitrary relation B on a set V gives rise to a graph, more precisely, a directed 1-graph with associated relation B and vertex set V, denoted by (V, B). Boolean vectors are considered as representing sets of vertices. In particular, vertices $x, y \in V$ are treated as special Boolean vectors. Note that in this case $x \subset By$ means that there is an arc from x to y. The zero vector, representing the empty subset of vertices, is always denoted by the same symbol 0, as

* Received by the editors May 26, 1982, and in revised form October 5, 1983.

† Institut für Informatik, Technische Universität München, D-8000 München 2, Bundesrepublik Deutschland.

is the zero relation; similarly we use L for the full subset V of vertices and the universal relation.

If the rules of the following theorem are considered as matrix identities, the assertion is obvious. Our formal proof, see [48], depends on injectivity ($xx^T \subset I$) and row-constancy ($x = xL$) of the relation $x \neq 0$ which represents a single vertex.

PROPOSITION 1. *For arbitrary relations R, S and vertices x, y the following holds*:

i) $\bar{R}x = \overline{Rx}$; ii) $x \subset Ry \Leftrightarrow xy^T \subset R$; iii) $(R \wedge S)x = Rx \wedge Sx$.

Proof. i) Applying monotonicity and Schröder's rule, we get

$$xx^T \subset R^T \Rightarrow xx^T R^T \subset I \Leftrightarrow x^T \overline{R^T} \subset \overline{x^T R^T} \Leftrightarrow \bar{R}x \subset \overline{Rx}.$$

For the opposite inclusion, we show $L = LxL = Lx = (R \vee \bar{R})x = Rx \vee \bar{R}x \Leftrightarrow \overline{Rx} \subset \bar{R}x$.

ii) From left to right: $x \subset Ry \Rightarrow xy^T \subset Ryy^T \subset R$. Conversely,

$$xy^T \subset R \Leftrightarrow x^T \bar{R} \subset \overline{y^T} \Leftrightarrow \bar{R}^T x \subset \bar{y} \Leftrightarrow \bar{R}y \subset \bar{x},$$

and therefore by (i) $\overline{Ry} \subset \bar{x}$, i.e., $x \subset Ry$.

iii) Applying Dedekind's rule, $\wedge$-subdistributivity is tightened for an injective x:

$$Rx \wedge Sx \subset (R \wedge Sxx^T)(x \wedge R^T Sx) \subset (R \wedge S)x. \qquad \square$$

In a graph $G = (V, B)$, the vector $\overline{BL}$ corresponds to the set of *terminal vertices*. The relation B^* comprises all the pairs of vertices for which there exists a path (repetition of arcs and vertices not excluded) from the first vertex to the second one; it is called the *reachability* of G: The relation B represents the arcs of G, and B^n represents reachability along paths of length n. This makes related notions precisely definable: *loopfreeness* ($B \subset \bar{I}$), *circuitfreeness* (acyclicity) ($B^+ \subset \bar{I}$) and *strong connectedness* ($B^* = L$).

Finiteness (or finite order) of a graph is declared via V. A graph is of *finite outdegree* if every vertex has a finite number of successors $B^T x$. More subtle notions are defined by calling

$$G \text{ progressively bounded} :\Leftrightarrow \sup_{i \geqq 1} \overline{B^i L} = L,$$

$$G \text{ progressively finite} :\Leftrightarrow (x \subset Bx \Rightarrow x = 0).$$

The bounded progress condition means that for every vertex x there is a natural number $h(x)$ which bounds the lengths of all paths starting from x. The condition of finite progress was studied by von Neumann [32, Chap. XII] in the transposed form of finite regress.

From every vertex of a set x fulfilling $x \subset Bx$ there starts a path of infinite length because $x \subset Bx$ guarantees a successor in x for every vertex of x. Therefore G is progressively finite iff there are no infinite paths.

For further illustrations regarding a loopfree relation B, we consider the implication for finite progress in the equivalent form $x \neq 0 \Rightarrow x \wedge \overline{Bx} \neq 0$. Call a vertex m a maximum of the vertex set x if m belongs to x and if no vertex y exists in x with $m \subset By$. Then the set $x \wedge \overline{Bx}$ describes the maxima of x. Therefore, G is progressively finite iff every nonempty set of vertices possesses maxima.

Obviously, in a progressively finite graph $B^* \overline{BL} = L$ is valid, i.e., from every vertex a terminal vertex can be reached. The concepts of bounded and finite progress coincide in graphs of finite outdegree ([27], König's lemma).

As a second tool, we introduce a special fixpoint consideration exhibiting the lattice-theoretical facet of the problem of existence of kernels. Consider an antitone

function f with $f(0)=L$ on a complete lattice with least and greatest elements 0 and L. Antitony is equivalent to either of the "antimorphism properties"

$$f\left(\sup_{i\geqq 1} x_i\right) \subset \inf_{i\geqq 1} f(x_i), \qquad \sup_{i\geqq 1} f(x_i) \subset f\left(\inf_{i\geqq 1} x_i\right).$$

A fixpoint of f is a fixpoint of f^2. Due to Tarski's theorem the isotone function f^2 has indeed fixpoints in a complete lattice. Note that 0 is a fixpoint of f^2 iff $f(L)=0$.

We introduce the sets of pre- and post-fixpoints of f by $V_s := \{x: x \subset f(x)\}$, $V_a := \{x: f(x) \subset x\}$. (The subscripts mnemonically refer to "stable" and "absorbant".) Because of antitony of f

$$f(V_s) \subset V_a, \quad f(V_a) \subset V_s \quad \text{and} \quad f^2(V_s) \subset V_s, \quad f^2(V_a) \subset V_a$$

hold. If x is a fixpoint of f and z is an element with $x \subset z \subset f(z)$ then $z \subset f(z) \subset f(x) = x$ follows, i.e. $z = x$. Analogously, $f(z) \subset z \subset x$ implies $z = x$. This means that a fixpoint of f is maximal in V_s and is minimal in V_a. From $y \subset x \in V_s$ follows $y \in V_s$ since $y \subset x \subset f(x) \subset f(y)$. Similarly, $y \supset x \in V_a$ implies $y \in V_a$.

The determination of a fixpoint x of an antitone function f with $f(0)=L$ is supported by the fundamental iterations

$$s_0 := 0, \qquad z_0 := L,$$
$$s_{i+1} := f^2(s_i), \qquad z_{i+1} := f^2(z_i)$$

for which a straightforward induction reveals the inclusions ($i \geqq 0$)

$$s_i \subset s_{i+1} = f(z_i) \subset x \subset f(s_{i+1}) = z_{i+1} \subset z_i.$$

This gives rise to defining the *iterative bounds* of the set of fixpoints of f

$$S := \sup_{i\geqq 1} s_i; \qquad Z := \inf_{i\geqq 1} z_i,$$

fulfilling $S \subset Z$ and $f(Z) \subset f(S)$.

This process has been considered in different notations by many authors of graph and game theory. Formulations using pseudo-Boolean programming or Boolean matrices or a remoteness function are known; sometimes [23] this process is traced back even to Steinhaus. The first precise presentation, however, can be found in von Neumann–Morgenstern [32].

The effectiveness of the iteration heavily depends on the value $f(L)$; in particular, it is completely useless if $f(L)=0$, since $f(L)=0 \Leftrightarrow S=0$, $Z=L$.

PROPOSITION 2i. *Let f be a function on a complete lattice fulfilling $f(0)=L$ and the antimorphism properties. Then the iterative bounds of fixpoints satisfy*

$$S \subset f(Z) \subset f(S) \subset Z.$$

If equality holds in the first antimorphism property we have: $f(S)=Z$. If the lattice is finite we have: $S=f(Z)$, $f(S)=Z$.

Proof. We apply, e.g., the second antimorphism property

$$S = \sup_{i\geqq 1} s_i = \sup_{i\geqq 0} f(z_i) \subset f\left(\inf_{i\geqq 0} z_i\right) = f(Z).$$

In a finite lattice there exists an integer n with $S = s_n = s_{n+1}$, $Z = z_n = z_{n+1}$ and therefore $f(Z) = f(z_n) = s_{n+1} = S$ holds. □

If the gap between the sequences s_i and z_i closes, the coinciding bound $S = Z$ is a uniquely determined fixpoint.

We now introduce similar constructions concerning f^2, $V_r := \{x: x \subset f^2(x)\}$, $V_e := \{x: f^2(x) \subset x\}$ where the subscripts associate with "retarding" and "expansive." From isotony of f^2 we deduce that V_e and V_r are closed under inf, sup respectively: Consider $Y \subset V_e$. On the one hand $\inf\{f^2(y): y \in Y\} \subset \inf Y$ follows from $f^2(y) \subset y$ for all $y \in Y$, and on the other hand $f^2(\inf Y) \subset \inf\{f^2(y): y \in Y\}$ holds, because $\inf Y \subset y$ for all $y \in Y$ implies $f^2(\inf Y) \subset f^2(y)$ for all $y \in Y$. Both inclusions result in $\inf Y \in V_e$. Analogously, $Y \subset V_r \Rightarrow \sup Y \in V_r$ is shown.

A transfinite generalization of the iteration for s_i and z_i suggests the definition of the *descriptive bounds* of the set of fixpoints of f

$$\zeta := \inf V_e, \qquad \sigma := \sup V_r.$$

The preceding remark reveals $\zeta \in V_e$, $\sigma \in V_r$. Because of $f^2(V_e) \subset V_e$, $f^2(V_r) \subset V_r$, even

$$\zeta = f^2(\zeta), \qquad \sigma = f^2(\sigma)$$

hold. More precisely, as a consequence of Tarski's fixpoint theorem ζ is the least fixpoint of f^2 and σ the greatest one. The following result is closely related to ideas of Roth [40], [41] and Blair–Roth [7].

PROPOSITION 2ii. *Let f be an antitone function with $f(0) = L$ on a complete lattice. Then the iterative and descriptive bounds of fixpoints satisfy*

$$S \subset f(\sigma) = \zeta \subset \sigma = f(\zeta) \subset Z.$$

If the lattice is finite we have: $S = \zeta$, $\sigma = Z$.

Proof. The extremal properties reveal $\zeta \subset \sigma$, and in connection with $f(V_r) \subset V_e$ and $f(V_e) \subset V_r$, the inclusions $\zeta \subset f(\sigma)$, $f(\zeta) \subset \sigma$ follow. From $\zeta \subset f(\sigma)$ we deduce $f(\zeta) \supset f^2(\sigma) = \sigma$, i.e. $f(\zeta) = \sigma$. Applying f we get $f^2(\zeta) = \zeta = f(\sigma)$. For every $x \in V_r$ an induction yields $x \subset z_i$ for every i and therefore $x \subset Z$. In particular, $\sigma \subset Z$ holds. Following the same pattern, $S \subset x$ for every $x \in V_e$ implies $S \subset \zeta$. In a finite lattice $S = f(Z)$, $f(S) = Z$ hold. Applying f we get $f^2(S) = S$, $f^2(Z) = Z$ establishing that S, Z are fixpoints of f^2; therefore $\sigma = Z$, $\zeta = S$. □

2. Kernels of graphs. A kernel of a graph combines the divergent properties of stability and absorption, i.e. a set x of vertices is a kernel if there are no arcs inside the set x but from each vertex not belonging to x there exists an arc leading into x.

DEFINITION 3. In a graph with associated relation B we call a set of vertices

$$x \textit{ stable} \quad :\Leftrightarrow Bx \subset \bar{x},$$

$$x \textit{ absorbant} :\Leftrightarrow \bar{x} \subset Bx,$$

$$x \textit{ kernel} \quad :\Leftrightarrow \bar{x} = Bx. \qquad \square$$

Often, the transposed notions "dominance" ($\bar{x} \subset B^T x$) and "solution" ($\bar{x} = B^T x$) are used. The concept of a kernel occurs even in König [28] where it is termed "pointbasis of the second kind." Then "basis" is a kernel of B^*. To indicate related phenomena, see [24], [29], [60], [33], we mention further: "semi-kernel" ($B^T x \subset Bx \subset \bar{x}$) and "quasi-kernel" ($Bx \subset \bar{x} \subset Bx \vee B^2 x$). Chvátal and Lovász have shown that every loopfree graph has a quasi-kernel. A "k-kernel" or "k-basis" is a kernel of $B \vee \cdots \vee B^k$. In [57], [8] "mini-maximal kernels" are studied: kernels which are in addition stable sets of maximum cardinality and absorbant sets of minimum cardinality. In [61] the concept is extended to the field of hypergraphs.

Stability does not depend on the "direction" of edges. This is obvious; formally it corresponds to an application of Schröder's law: $Bx \subset \bar{x} \Leftrightarrow B^T x \subset \bar{x}$. Absorption of a set is independent from the existence of loops.

The lattice-theoretical background of the problem of the existence of a kernel is the question of fixpoints of the function $f(x) = \overline{Bx}$. Since f is antitone, we may apply the theorems of the preceding section. Specific properties of the function defined here will then reveal further results.

A reformulation of a result of § 1 reads

$$x \text{ kernel} \Rightarrow x \text{ maximal stable and minimal absorbant set.}$$

The statements of the following theorem give several well-known conditions to ensure that this implication holds in the reverse direction. Nevertheless we have included a formal proof.

THEOREM 4. i) *Every symmetric and loopfree graph has a kernel. Kernels are exactly the maximal stable sets.* ii) *Every transitive graph with* $B^*\overline{BL} = L$ (*which condition is fulfilled e.g. by finite loopfree graphs or progressively finite graphs*) *has exactly one kernel. The kernel is the set* $\overline{BL}$ *of terminal vertices which in addition is the least absorbant set.*

Proof. i) Let x be a maximal stable set. Assume $Bx \subsetneq \bar{x}$ and let $v \subset \overline{Bx} \wedge \bar{x}$ be an arbitrary vertex. We show $B(x \vee v) \subset \overline{x \vee v}$ in contradiction to the maximality of x, thus refuting the assumption $Bx \neq \bar{x}$: (1) $Bx \subset \bar{x}$; (2) $Bx \subset \bar{v} \Leftrightarrow v \subset \overline{Bx}$; (3) $Bv \subset \bar{x}$, applying Schröder's rule and symmetry of B; (4) $Bv \subset \bar{v}$, may be deduced from $vv^T \subset I \subset \bar{B}$ by Proposition 1 and loopfreeness.

ii) Transitivity implies $L = B^*\overline{BL} = (I \vee B)\overline{BL} = \overline{BL} \vee B\overline{BL}$, i.e. $BL \subset B\overline{BL}$, and therefore $BL = B\overline{BL}$ because $BL \supset B\overline{BL}$ generally holds. Finally, an arbitrary absorbant set y necessarily fulfills $\overline{BL} \subset \overline{By} \subset y$. □

Part (ii) assures the existence of a basis of a finite graph without loops or a graph with finite progress. This has been the starting point in the original proof of Richardson's theorem [36].

Applying the results of Proposition 2i and ii concerning iterative and descriptive bounds for the function $f(x) = \overline{Bx}$, we obtain

$$S \subset \overline{BZ} \subset \overline{B\sigma} = \zeta \subset \sigma = \overline{B\zeta} \subset \overline{BS} = Z.$$

If a kernel x exists, we may additionally insert $\zeta \subset x \subset \sigma$. Note, that the asymmetric occurrence of the equality $\overline{BS} = Z$ is a result of $\vee$-distributivity which gives the strengthened antimorphism property

$$f(\sup_{i \geq 1} x_i) = \overline{B \sup_{i \geq 1} x_i} = \overline{\sup_{i \geq 1} Bx_i} = \inf_{i \geq 1} \overline{Bx_i} = \inf_{i \geq 1} f(x_i).$$

Fig. 1 shows an example with $S \neq \overline{BZ}$.

PROPOSITION 5. *In an arbitrary graph with associated relation B and iterative bounds S and Z we have*

$$\sup_{i \geq 0} \overline{B^i L} \subset \bar{Z} \vee S.$$

In particular $S = Z$ *holds for progressively bounded graphs.*

Proof. We remember the sequences s_i and z_i and use an inductive argument. The assumption $z_i \subset s_i \vee B^{2i}L$ implies $\overline{s_{i+1}} \subset \overline{z_i} \vee B^{2i+1}L \Leftrightarrow z_i \subset s_{i+1} \vee B^{2i+1}L \Rightarrow \overline{s_{i+1}} \subset \overline{z_{i+1}} \vee$

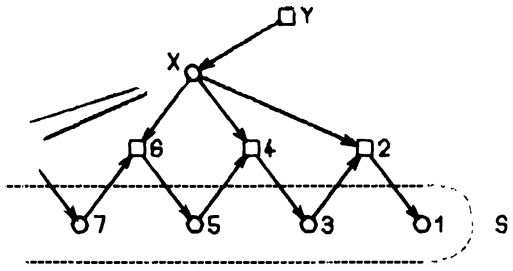
$B^{2i+2}L \Leftrightarrow z_{i+1} \subset s_{i+1} \vee B^{2(i+1)}L$. Using the equivalent inclusions $\overline{B^{2i}L} \subset \overline{z_i} \vee s_i$, we get

$$\sup_{h \geqq 0} \overline{B^h L} = \sup_{i \geqq 0} \overline{B^{2i} L} \subset \sup_{i \geqq 0} \overline{z^i} \vee \sup_{i \geqq 0} s_i = \bar{Z} \vee S. \qquad \square$$

Combining this with the result of Proposition 2i we get Corollary 6.

COROLLARY 6. *A progressively bounded graph has exactly one kernel. The kernel is determined by the iterative bounds* $S = Z$.

This argument does not apply to the progressively finite case as is shown in Fig. 1.



$$B = \begin{array}{c} \\ x \\ y \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ \vdots \end{array} \begin{array}{c} \begin{array}{ccccccccc} x & y & 1 & 2 & 3 & 4 & 5 & \cdot & \cdot \end{array} \\ \begin{pmatrix} 0 & 0 & 0 & L & 0 & L & 0 & \cdot & \cdot \\ L & 0 & 0 & 0 & 0 & 0 & 0 & \cdot & \cdot \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdot & \cdot \\ 0 & 0 & L & 0 & 0 & 0 & 0 & \cdot & \cdot \\ 0 & 0 & 0 & L & 0 & 0 & 0 & \cdot & \cdot \\ 0 & 0 & 0 & 0 & L & 0 & 0 & \cdot & \cdot \\ 0 & 0 & 0 & 0 & 0 & L & 0 & \cdot & \cdot \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{pmatrix} \end{array} \quad \begin{array}{cccc} \overline{BL} & S & Z & \overline{BZ} \\ \begin{pmatrix} 0 \\ 0 \\ L \\ 0 \\ 0 \\ 0 \\ 0 \\ \vdots \end{pmatrix} & \begin{pmatrix} 0 \\ 0 \\ L \\ 0 \\ L \\ 0 \\ L \\ \vdots \end{pmatrix} & \begin{pmatrix} L \\ L \\ L \\ 0 \\ L \\ 0 \\ L \\ \vdots \end{pmatrix} & \begin{pmatrix} L \\ 0 \\ L \\ 0 \\ L \\ 0 \\ L \\ \vdots \end{pmatrix} \end{array}$$

FIG. 1. *Progressively finite, but not progressively bounded graph.*

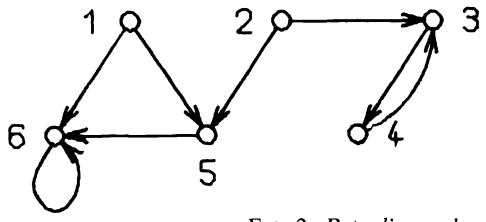
The existence of kernels can also be shown in the case of progressively finite graphs; however, by inherently nonconstructive methods. We are looking for a graph-theoretical interpretation of the properties $f^2(x) \subset x$ and $x \subset f^2(x)$ in the special case of the function $f(x) = \overline{Bx}$.

DEFINITION 7. In a graph with associated relation B we call a set of vertices

x *retarding* $:\Leftrightarrow B\overline{Bx} \subset \bar{x} \Leftrightarrow B^T x \subset Bx$;

x *expansive* $:\Leftrightarrow \bar{x} \subset B\overline{Bx}$.

The two definitions of retardation are equivalent because of Schröder's rule. Notice: Expansiveness does not allow similar equivalences. A set is retarding if its successor set is contained in its predecessor set; it is expansive if its complement consists only of predecessors of the complement of its predecessors. A "subsolution" [40], [42] is a stable, retarding and expansive set with respect to B^T. Figure 2 compares the phenomenology of these notions with stability and absorption. The set 0 is always retarding, while L is always expansive.



{1, 2} stable, not retarding
{3, 4, 5, 6} absorbant, not expansive
{1, 2, 4, 6} retarding, not stable
{1, 5, 6} expansive, not absorbant

FIG. 2. *Retarding and expansive sets.*

The sets s_i and S of the iteration are retarding as is σ, similarly z_i, Z and ζ are expansive. A kernel has all of the four properties. Using the second characterization of retardation, we will derive an implication.

THEOREM 8. *In a progressively finite graph we have*

$$x \text{ retarding} \Rightarrow x \text{ stable.}$$

Proof. Consider $y := x \wedge Bx$. Applying the Dedekind rule, we get

$$y \subset (B \wedge xx^T)(x \wedge B^T x) \subset B(x \wedge B^T x) \subset B(x \wedge Bx) = By,$$

and therefore $y = 0$, i.e. $Bx \subset \bar{x}$. □

Now, the following result is established by combining Proposition 2ii ($\overline{B\sigma} \subset \sigma$) and Theorem 8 ($B\sigma \subset \bar{\sigma}$); this is the only—but decisive!—use of Theorem 8.

COROLLARY 9. *A progressively finite graph has exactly one kernel. The kernel is determined by the descriptive bounds $\zeta = \sigma$.*

The work of von Neumann comprehends most of the preceding results about kernels (solutions) and stable sets.

Next, we try to get rid of the assumption of progressive finiteness by taking circuits into consideration. As a first attempt, we exclude odd circuits, i.e. circuits of odd length, and for simplicity we restrict ourselves to a strongly connected graph G with set of vertices V, $|V| > 1$. Obviously, G is a bipartite graph, i.e. $V = w \vee \bar{w}$ and w, $\bar{w}$ are two complementary kernels. This result was mentioned in [1], [30]; we give a formal proof as presented in [47].

THEOREM 10. *A strongly connected graph without odd circuits and with more than one vertex has at least two kernels which are complementary to each other.*

Proof. Let $G = (V, B)$. With the abbreviation $D := (B^2)^*$ we write

$$G \text{ strongly connected} \Leftrightarrow L = B^* = D \vee BD \Leftrightarrow \bar{D} \subset BD,$$

$$G \text{ without odd circuits} \Leftrightarrow BD = BDD \subset \bar{I} \Leftrightarrow (BD)^T \subset \bar{D}.$$

Combining both assumptions, we get $(BD)^T \subset \bar{D} \subset BD$, and $BD = \bar{D}$. Each column of D is a kernel. If $|V| > 1$ there are two kinds of columns corresponding to the vertices of a bipartition.

Richardson who carried on the work of von Neumann succeeded, see [36], without assuming the condition of strong connectedness, however, introducing a restriction to some finiteness conditions.

THEOREM 11. *A finite graph (or a graph of finite progress or of finite outdegree) without odd circuits has at least one kernel.*

An idea of a proof is contained in the subsequent considerations.

Figure 3 indicates a backward procedure supporting the construction of a kernel. A graph (odd circuits not necessarily excluded!) is drawn above the irreflexive part of

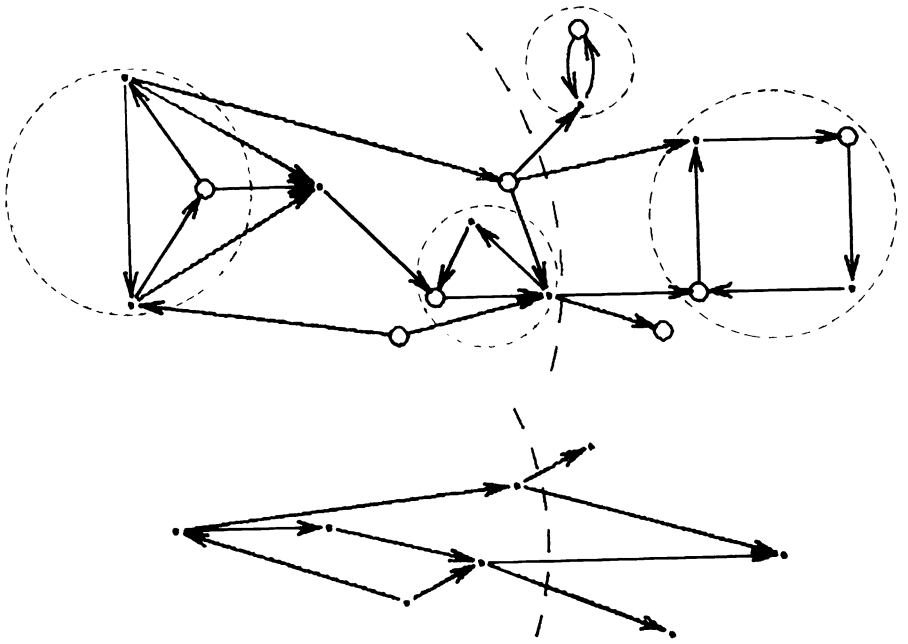


FIG. 3. *Backward construction searching for a kernel.*

its reduction; the terminal components, vertices resp. are separated. Consider the following iterative algorithm:

1. Find kernels in the terminal strong components of G and mark the vertices of these kernels;

2. $G :=$ subgraph of G generated from the vertices that neither belong to a terminal strong component of G nor have an arc leading into the kernels just established by 1;

3. If G no longer contains vertices then stop else goto 1.

If it is successful, it ends with a kernel whose vertices are marked. Clearly, the algorithm is not straightforward (because of multiple choice of kernels in the strong components), and, in general the search is not successful. But in the special case of graphs without odd circuits success is more likely, because Theorem 10 guarantees that whatever strong component we may encounter, a kernel in that component does exist. However, the algorithm might still not exhaust the whole graph. If we assume the irreflexive part of the reduced graph to be progressively bounded, the algorithm will succeed in exhausting.

The computational complexity of finding a kernel has been investigated by Chvátal and Fraenkel. In [19] NP-completeness has been shown even if the problem is restricted to planar graphs.

Richardson examines his proof and reveals special conditions under which the occurrence of odd circuits does not invalidate success. A step in this direction is due to Romanowicz who gives an algorithm-independent device: he admits odd circuits in which at least one arc has an arc in opposite direction which, in turn, does not lie on an odd circuit. Dually, one may admit odd circuits each of which contains at least one vertex x such that $Bx = B^T x$. This means that the set of predecessors of x coincides with the set of successors of x.

In particular, Theorem 11 applies to bipartite graphs. However, we show a stronger result without any assumption concerning finiteness.

THEOREM 12. *If $G=(V,B)$ is a bipartited graph with bipartition $V=w\vee\bar{w}$, then the following holds*:

i) *G has two distinguished (not necessarily distinct) kernels determined by the descriptive bounds*:

$$\sigma' := (w\wedge\sigma)\vee(\bar{w}\wedge\zeta),\qquad \zeta' := (w\wedge\zeta)\vee(\bar{w}\wedge\sigma).$$

ii) *An arbitrary kernel x is limited by these kernels as*:

$$\sigma'\wedge\zeta'\subset x\subset\sigma'\vee\zeta'.$$

Proof. i) We use the formula $B(w\wedge x)=(\bar{w}\wedge Bx)$, generally valid in a bipartition, and the equations in Proposition 2ii

$$\begin{aligned}B\sigma' &= B(w\wedge\sigma)\vee B(\bar{w}\wedge\zeta)=(\bar{w}\wedge B\sigma)\vee(w\wedge B\zeta)\\ &=(\bar{w}\wedge\bar{\zeta})\vee(w\wedge\bar{\sigma})=(\bar{w}\vee\bar{\sigma})\wedge(w\vee\bar{\zeta})=\overline{\sigma'}.\end{aligned}$$

By analogy we get $B\zeta'=\overline{\zeta'}$.

ii) From $\zeta\subset\sigma$ follows

$$\begin{aligned}\sigma'\wedge\zeta' &= [(w\wedge\sigma)\vee(\bar{w}\wedge\zeta)]\wedge[(w\wedge\zeta)\vee(\bar{w}\wedge\sigma)]\\ &=[(w\wedge\sigma)\wedge(w\wedge\zeta)]\vee[(\bar{w}\wedge\zeta)\wedge(\bar{w}\wedge\sigma)]\\ &=(w\wedge\sigma\wedge\zeta)\vee(\bar{w}\wedge\zeta\wedge\sigma)=\sigma\wedge\zeta=\zeta.\end{aligned}$$

$\sigma'\vee\zeta'=\sigma$ is derived analogously. □

Note that in finite or progressively bounded graphs descriptive and iterative limits coincide. The proof of (ii) also works if σ', ζ' are defined involving the iterative limits S and Z instead of σ and ζ, thus establishing different (weaker, but constructive) limits of an arbitrary kernel. Clearly, these constructions need not be kernels in general. An example is supplied by a slight modification of the graph of Fig. 1. The early versions of this result go back to Ströhlein ([53], also [55]) where finite bipartite graphs in connection with combinatorial games are studied. Roth [44] calls the bipartite case "asymmetric."

3. Solutions of games. The games we are now going to discuss, are finite two-player games with perfect information with alternate moves and with the last player losing. This essentially restricts us to games played on a board, like chess, and Nim-like games. Zermelo, König [27], Kalmár and Euwe have contributed to the investigation of chess. The game of Nim was fitted into mathematics by Bouton and Wythoff. Nim has a progressively bounded game graph. Every combinatorial game possesses at least a description by a bipartite game graph.

Subsequently, qualifications of loss, win, or draw for a player are used only in connection with positions in which it is this player's turn to move. In this sense, the knowledge of a kernel provides a player in positions outside the kernel at least with a strategy for avoiding loss: *Move into the kernel*! In the case of a progressively finite game graph the kernel is additionally good enough to assure win. If no kernels exist the advice *Move into the set S*! may lead to success in specific positions.

If the graph is not necessarily progressively finite we consider a bipartite description. For describing chess we apply Theorem 12 (w: white, $\bar{w}$: black) with distinguished kernels

$$S'=(w\wedge S)\vee(\bar{w}\wedge Z),\qquad S''=(w\wedge Z)\vee(\bar{w}\wedge S).$$

In terms of games we can say that S' determines loss of w or nonwin of $\bar{w}$ and that S'' determines loss of $\bar{w}$ or nonwin of w. The intersection of these kernels comprises loss, the symmetric difference draw. The knowledge of these kernels enables w to avoid loss, but only knowledge of their partition by the sequences s_i and z_i enables w to achieve win by the strategy: *Move from s_i into z_i!*

In the past, refined versions of the fundamental iteration were applied to practical problems. A survey of the activities in chess is given, e.g., in [12]. The table in Fig. 4 concentrates the complete analysis of some chess endings carried out in Munich. The results concerning the first 5 games have been evaluated in the years 1967–1969 in connection with the doctoral thesis [53] on an AEG-Telefunken computer TR4 while the last two games have been investigated later with adequate machine power. The win prediction number has been defined differently according to different games: in (3)–(5) as number of moves to force win by the capture of the black king (mate+1) or piece; in (1), (2), (6) and (7) as number of moves to force win by mate or the capture of a black man. The first and third endgames are exhibited in [56].

Endgame	Greatest win prediction number	Number (reduced by symmetries) of positions with greatest win prediction number	Example
1. *wR*	16	121	*w Ka1 Rb2, b Kc3.*
2. *wQ*	10	1	*w Ka1 Qb2, b Ke6.*
3. *wR : bB*	18	28	*w Ka4 Rc3, b Ka7 Ba6.*
4. *wR : bN*	27	2	*w Kd1 Rh1, b Kb1 Ng4.*
5. *wQ : bR*	31	4	*w Ka1 Qa4, b Kf3 Re1.*
6. *wQ : bQ*	10	5	*w Ke1 Qg1, b Kb1 Qa1.*
7. *wQ : bR + bP(d2)*	29	10	*w Kh8 Qa4, b Kf8 Rf2 Pd2.*

FIG. 4. *Some chess endings with no more than 5 men.*

Acknowledgments. We enjoyed detailed comments of the unknown referees and discussions with our colleagues, among which L. Zagler made considerable contributions. We are grateful to R. Berghammer, E. Hangel and B. Möller for carefully reading the manuscripts and to H. Gruschka, C. Halfar, M. Krämer and F. X. Winter for technical support.

REFERENCES

[1] M. ANCIAUX-MUNDELEER AND P. HANSEN, *On kernels in strongly connected graphs*, Networks, 7 (1977) pp. 263–266.

[2] G. AUMANN, *Über autogene Folgen und die Konstruktion des Kerns eines Graphen*, Bayer. Akad. Wiss. Math.-Natur. Kl. Sitzungsber, 1966 (1967), pp. 53–63.

[3] M. BEHZAD AND F. HARARY, *Which directed graphs have a solution?* Math. Slovaca, 27 (1977), pp. 37–42.

[4] C. BERGE, *Sur l'inversion des transformateurs*, C. R. Acad. Sci. Paris Sér. A-B, 232 (1951), pp. 134–136.

[5] ———, *Théorie générale des jeux à n personnes*, Mémor. Sci. Math. No. 138 (1957).

[6] ———, *Nouvelles extensions du noyau d'un graphe et ses applications en théorie des jeux*, Publ. Econométriques, 6 (1973), pp. 6–11.

[7] C. BLAIR AND A. E. ROTH, *An extension and simple proof of a constrained lattice fixed point theorem*, Algebra Universalis, 9 (1979), pp. 131–132.

[8] M. BOROWIECKI, *On the graphs with minimaximal kernels*, Prace Nauk. Inst. Mat. Politech. Wrocław Ser. Stud. Materialy No. 13 (1977), pp. 3–7. MR 57 12296.

[9] C. L. BOUTON, *Nim, a game with a complete mathematical theory*, Ann. Math., (2) 3 (1902), pp. 35–39.

[10] C.-Y. CHAO, *On a problem of C. Berge*, Proc. Amer. Math. Soc., 14 (1963), p. 80.
[11] V. CHVÁTAL AND L. LOVÁSZ, *Every directed graph has a semi-kernel*, in Hypergraph Seminar, Ohio, Aug. 10–Sept. 9, 1972, C. Berge, D. Ray-Chauduri, eds., Lecture Notes in Mathematics 411, Springer, Berlin, 1974, p. 75.
[12] M. R. B. CLARKE, *Advances in computer chess* 1, Edinburgh Univ. Press, Edinburgh, 1977.
[13] H. S. M. COXETER, *The golden section, phyllotaxis, and Wythoff's game*, Scripta Math., 19 (1953), pp. 135–143.
[14] P. DUCHET, *Graphes noyau-parfaits*, Ann. Discrete Math., 9 (1980), pp. 93–101.
[15] ———, *Two problems in kernel theory*, Ann. Discrete Math., 9 (1980), p. 302.
[16] P. DUCHET AND H. MEYNIEL, *A note on kernel-critical graphs*, Discrete Math., 33 (1981), pp. 103–105.
[17] ———, *Une généralisation du théorème de Richardson sur l'existence de noyaux dans les graphes orientés*, Discrete Math., 43 (1983), pp. 21–27.
[18] M. EUWE, *Mengentheoretische Betrachtungen über das Schachspiel*, Proc. Section of Sciences. Koninklijke Akad. van Wetensch., 32 (1929), pp. 633–642.
[19] A. S. FRAENKEL, *Planar kernel and Grundy with* $d \leqq 3$, $d_{out} \leqq 2$, $d_{in} \leqq 2$ *are* NP-*complete*, Discrete Appl. Math., 3 (1981), pp. 257–262.
[20] H. GALEANA-SÁNCHEZ, *A counterexample to a conjecture of Meyniel on kernel-perfect graphs*, Discrete Math., 41 (1982), pp. 105–107.
[21] P. M. GRUNDY AND C. A. B. SMITH, *Disjunctive games with the last player losing*, Proc. Cambridge Philos. Soc., 52 (1956), pp. 527–533.
[22] R. K. GUY AND C. A. B. SMITH, *The G-values of various games*, Proc. Cambridge Philos. Soc., 52 (1956), pp. 514–526.
[23] R. K. GUY, Reviews MR 58 33000, 80*c*: 90159.
[24] F. HARARY AND M. RICHARDSON, *A matrix algorithm for solutions and r-bases of a finite irreflexive relation*, Naval Res. Logist. Quart., 6 (1959), pp. 307–314.
[25] L. KALMÁR, *Zur Theorie der abstrakten Spiele*, Acta Sci. Math. (Szeged), 4 (1928/29), pp. 65–85.
[26] B. KNASTER, *Un théorème sur les fonctions d'ensembles*, Ann. Société Polonaise de Mathématique, 6 (1927), pp. 133–134.
[27] D. KÖNIG, *Über eine Schlussweise aus dem Endlichen ins Unendliche*, Acta Sci. Math. (Szeged), 3 (1927), pp. 121–130.
[28] ———, *Theorie der endlichen und unendlichen Graphen*, Akademische Verlagsgesellschaft, Leipzig, 1936.
[29] M. KWAŚNIK, *Über einen k-Antikern des gerichteten Graphen*, Zeszyty Naukowe, Wyższa Szkoła Inżynierska w Zielonej Górze, Matematyka-Fizyka 55 (1980), pp. 33–36.
[30] D. MARCU, *A method for finding the kernel of a digraph*, An. Univ. Bucureşti Mat. 27 (1978), pp. 41–43, MR 80a:5102.
[31] J. VON NEUMANN, *Zur Theorie der Gesellschaftsspiele*, Math. Ann., 100 (1928), pp. 295–320.
[32] J. VON NEUMANN AND O. MORGENSTERN, *Theory of Games and Economic Behavior*, Princeton Univ. Press, Princeton, NJ, 1944.
[33] V. NEUMANN LARA, *Seminuclei of a digraph*, An. Inst. Mat. Univ. Nac. Autónoma México, 11 (1971), pp. 55–62, MR 47 6536. (In Spanish.)
[34] R. G. NIGMATULLIN, *The largest number of kernels in graphs with n vertices*, Kazan. Gos. Univ. Ucen. Zap., 130 (1970), pp. 75–82, MR 44 3915. (In Russian.)
[35] M. RICHARDSON, *On weakly ordered systems*, Bull. Amer. Math. Soc., 52 (1946), pp. 113–116.
[36] ———, *Solutions of irreflexive relations*, Ann. of Math., (2) 58 (1953), pp. 573–590.
[37] ———, *Extension theorems for solutions of irreflexive relations*, Proc. Nat. Acad. Sci. USA, 39 (1953), pp. 649–655.
[38] ———, *Relativization and extension of solutions of irreflexive relations*, Pacific J. Math., 5 (1955), pp. 551–584.
[39] Z. ROMANOWICZ, *A note on Richardson's theorem*, Prace Nauk. Inst. Mat. Fiz. Teoret. Politech. Wrocław Ser. Stud. Materialy No. 4 (1971), pp. 43–46, MR 49 8896. (In Polish.)
[40] A. E. ROTH, *A fixed point approach to stability in cooperative games*, in S. Karamardian and C. B. Garcia (eds.): Fixed Points—Algorithms and Applications, Proc. of a Conf. Clemson, SC, June 26–28, 1974, Academic Press, New York, 1977, pp. 165–180.
[41] ———, *A lattice fixed-point theorem with constraints*, Bull. Amer. Math. Soc., 81 (1975), pp. 136–138.
[42] ———, *Subsolutions and the supercore of cooperative games*, Math. Oper. Res., 1 (1976), pp. 43–49.
[43] ———, *Two-person games on graphs*, J. Combin. Theory Ser. B, 24 (1978), pp. 238–241.
[44] ———, *A note concerning asymmetric games on graphs*, Naval Res. Logist. Quart., 25 (1978), pp. 365–367.

[45] S. RUDEANU, *Notes sur l'existence et l'unicité du noyau d'un graphe*, Rev. Française Recherche Opérationelle, 8 (1964), pp. 345–352.

[46] ———, *Notes sur l'existence et l'unicité du noyau d'un graphe II*, Applications des équations booléennes, Rev. Française Recherche Opérationelle, 10 (1966), pp. 301–310.

[47] G. SCHMIDT AND T. STRÖHLEIN, *Relationen, Graphen und Strukturen*, Vorlesungsskriptum 1974/75, Interner Bericht. Institut für Informatik der Techn. Univ. München.

[48] ———, *Relations, graphs and programs*, (to appear).

[49] C. A. B. SMITH, *Graphs and composite games*, J. Combin. Theory, 1 (1966), pp. 51–81.

[50] R. SPRAGUE, *Über mathematische Kampfspiele*, Tohoku Math. J., 41 (1935/36), pp. 438–444.

[51] ———, *Über zwei Abarten von Nim*. Tohoku Math. J., 43 (1937), pp. 351–354.

[52] H. STEINHAUS, *Definitions for a theory of games and pursuit* (in Polish, 1925), Engl. Reprint: Naval Res. Logist. Quart., 7 (1960), pp. 105–108.

[53] T. STRÖHLEIN, *Untersuchungen über kombinatorische Spiele*, Diss. Techn. Univ. München, 1970.

[54] ———, *Iterative Berechnung der Kerne eines Graphen und der Lösung eines Spiels*, in Graphen, Algorithmen, Datenstrukturen, H. Noltemeier, ed., Ergebnisse des Workshop WG 76, 2. Fachtagung über Graphentheoretische Konzepte der Informatik, Göttingen, 16–18 Juni, 1976, p. 326–336. München: Hanser 1976.

[55] T. STRÖHLEIN AND L. ZAGLER, *Analyzing games by Boolean matrix iteration*, Discrete Math., 19 (1977), pp. 183–193.

[56] ———, *Ergebnisse einer vollständigen Analyse von Schachendspielen-König und Turm gegen König-König und Turm gegen König und Läufer*, TUM-INFO-09-78-00-FBMA, Institut für Informatik der Techn. Univ. München, 1978.

[57] L. SZAMKOŁOWICZ, *Sur la classification des graphes en vue des propriétés de leurs noyaux*, Prace Nauk. Inst. Mat. Fiz. Teoret. Politech. Wrocław. Ser. Stud. Materialy 3 (1970), pp. 15–22, MR 52 # 5493. ZbL Mat 264 # 5125.

[58] A. TARSKI, *A lattice-theoretical fixpoint theorem, and its applications*, Pacific J. Math., 5 (1955), pp. 285–309.

[59] G. TINHOFER, *Über die Bestimmung von Kernen in endlichen Graphen*, Computing, 9 (1972), pp. 139–147.

[60] L. P. VARVAK, *Generalization of a kernel of a graph*, Ukrainian Math. J., 25 (1973), pp. 78–81, MR # 2005.

[61] P. VINCKE, *Hypergraphes orientés*. Cahiers Centre Etudes Rech. Opér., 17 (1975), pp. 407–416.

[62] ———, *Quasi-kernels of minimum weakness in a graph*. Discrete Math., 20 (1977), pp. 187–192.

[63] W. A. WYTHOFF, *A modification of the game of Nim*, Nieuw Arch. Wisk., (2) 7 (1905/07), pp. 199–202.

[64] E. ZERMELO, *Über eine Anwendung der Mengenlehre auf die Theorie des Schachspiels*, In Proc. 5th International Congress Mathematics, Vol. II, Cambridge, 1912, pp. 501–504.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 1, January 1985

A PERTURBATION RESULT FOR LINEAR CONTROL PROBLEMS*

DANIEL BOLEY†

Abstract. In this paper we will discuss some problems in computing the controllable (reachable) space for a linear system and give some perturbation analysis results that are significant for a popular algorithm used to compute that space, herein called the Staircase Algorithm.

AMS subject classifications. 65F99, 93B05, 15A03

1. Introduction. In this paper we will discuss some numerical problems in computing the controllable (reachable) space for a linear system

$$\dot{x} = Ax + Bu. \tag{1}$$

In this paper we confine our attention to the case where A, B are constant matrices, and x, u are vector functions of time. Under these conditions controllable and reachable are equivalent. One classic algebraic definition of the controllable space S_c is

$$S_c = \text{span}\,[B \quad AB \quad A^2B \quad \cdots \quad A^{n-1}B] \tag{2}$$

[1]. But numerical methods based on computing this matrix are very unstable. In [2], [9] and [4] it is pointed out that using (2) directly to compute the space S_c can be much more unstable than using a method based on orthogonal similarity transformations of A. To illustrate the pitfalls in (2) consider the system of the form (1) with

$$A = \text{diag}\,(32 \quad 16 \quad 8 \quad -64 \quad -32 \quad 4 \quad 2 \quad -2 \quad -4 \quad -8 \quad 16),$$

and B equal to the vector,

$$\begin{bmatrix} 10^{-14} \\ 2.321 \\ 0.385 \\ 10^{-14} \\ 1.161 \\ 0.187 \\ 0.500 \\ 3.674 \\ -1.119 \\ 0.070 \\ 10^{-14} \end{bmatrix}.$$

* Received by the editors August 8, 1982, and in revised form October 31, 1983. This paper was presented at the Applied Linear Algebra Conference held at Raleigh, North Carolina, April 1982. This research was supported by the National Science Foundation under grant ECS-8204468 and by the U.S. Army Research Office under grant DAHCO4-75-G-0185.

† Computer Science Department, University of Minnesota, Minneapolis, Minnesota 55455.

We see that dim $S_c = 8$, but if we compute the singular values of the matrix in (2) we obtain

$$10^{+15}\times\begin{bmatrix} 1.31\times10^{0} \\ 2.39\times10^{-4} \\ 4.28\times10^{-9} \\ 3.12\times10^{-10} \\ 3.88\times10^{-12} \\ 2.44\times10^{-13} \\ 4.91\times10^{-14} \\ 4.95\times10^{-15} \\ 3.75\times10^{-18} \\ 5.36\times10^{-28} \\ 0.0 \end{bmatrix}$$

which, depending on the choice of relative zero tolerance, would imply that dim S_c could have any value in the range 2 to 9. Further examples are given in [9] and [4].

In this paper we will indicate a more robust approach (§ 2) and give a limited sensitivity analysis of the problem (§§ 3–4). We then show how to apply the sensitivity analysis to this approach (§ 5), give some bounds for the numerical errors committed by the algorithm (§ 6), and indicate where the limitations of the analysis may lie (§ 7). We finally give the results of some numerical experiments (§ 8) and some concluding remarks (§ 9).

2. The method. A more stable method than using (2) is the so-called Staircase Algorithm using orthogonal transformations. It has been described in detail in many different places (e.g. [2], [4], [9], [10]) so that it suffices to give here only a short description. Briefly, this method consists of applying a series of orthogonal similarity transformations Q to (1) to obtain a system

$$\dot{z} = Q^T A Q z + Q^T B u \tag{3}$$

where we have $z = Q^T x$, $Q^T Q = I$, and the matrices $A' = Q^T A Q$, $B' = Q^T B$ are in a special reduced form:

$$A' = \begin{bmatrix} A'_{11} & A'_{12} \\ 0 & A'_{22} \end{bmatrix}, \qquad B' = \begin{bmatrix} B'_1 \\ 0 \end{bmatrix} \tag{3a}$$

with A'_{11} block upper Hessenberg [2], [4]. The transformation $Q = [Q_1 \; Q_2]$ can be partitioned as in (3a), where Q_1 is an orthogonal basis for the controllable space S_c. In the single input case, the method just reduces B to a multiple of $e_1 = (1, 0, \cdots, 0)^T$, and A'_{11} is upper Hessenberg. The first subdiagonal element $A'_{r+1,r}$ of the resulting A', which is zero, indicates the dimension r of the space S_c. The multiple input case proceeds in an analogous manner; in this case A'_{11} is reduced to a block upper Hessenberg form.

3. Sensitivity analysis, preliminaries. To understand the numerical properties of the Staircase Algorithm, one must observe that any random perturbation to the coefficients of (1) will tend to make (1) completely controllable [5, p. 100]. Hence we must be very careful in how we define a reasonable or *robust* answer. Our aim in this paper is two-fold: 1. to say what a *robust* answer might be and 2. to give a bound on the numerical errors committed by the Staircase Algorithm.

Since we are applying only orthogonal transformations to our original system (1), the resulting computed system

$$\dot{z} = Q^T(A+\varepsilon E)Qz + Q^T(B+\varepsilon F)u \tag{4a}$$

will be exactly equivalent to a slightly perturbed system (1):

$$\dot{x} = (A+\varepsilon E)x + (B+\varepsilon F)u \tag{4b}$$

where E, F are matrices with 2-norm of order $\min(1, \|A\|+\|B\|)$, n is the order of the system, and ε is the appropriate computer precision [6]. In other words, our algorithm does the equivalent of perturbing the coefficients by small multiples of ε; these perturbations we will call ε-perturbations.

By *robust*, we mean that the computed rank of the controllable space is insensitive to ε-perturbations in A, B, for some appropriate ε. To be *robust* in this sense, for a completely controllable system, the system must remain controllable after any such ε-perturbations to the original coefficients. For a system only partially controllable, the controllable part must remain controllable after any such perturbations. To be precise, we define the robust controllable space to be that space achieving the minimum in

$$\varepsilon\text{-rank} = \min \dim S_c(A+\Delta A, B+\Delta B),$$

where the minimum is taken over $\|\Delta A\| \leqq \varepsilon\|A\|$, and $\|\Delta B\| \leqq \varepsilon\|B\|$.

The first question to address is to determine whether or not the prospective controllable space obtained after application of a computational procedure is *robust* in that sense. We will address that question for the particular case of the Staircase Algorithm, giving a posteriori bounds. We will defer for the moment the question of finding a bound for the numerical errors in the computed basis for the controllable space S_c from ε-perturbations.

4. Sensitivity theorem. Given the previous discussion, we have the following limited sensitivity theorem for the single input case.

THEOREM 1. *Let*

A be upper Hessenberg, $n \times n$,

B be of the form $(b_1, 0, \cdots, 0)^T$, an n-vector.

If

$$\|A\|_2 + \|B\|_2 \leqq \tfrac{1}{4} \quad \textit{and} \quad |b_1 a_{21} a_{32} \cdots a_{n,n-1}| > 4\varepsilon \quad \textit{for } \varepsilon < \tfrac{1}{4},$$

then

the system (1) is completely controllable (i.e. $S_c = R^n$) and there is no system of the form (4b) with $\|E\|_2 < 1$, $\|F\|_2 < 1$ which yields an S_c of smaller dimension.

Proof. We have the following bound on the eigenvalues of A:

$$|\lambda| \leqq \|A\|_2 \leqq \tfrac{1}{4},$$

for any eigenvalue λ of A. If we perturb A slightly to obtain $A+\varepsilon E$, $\|E\|_2 < 1$, then we have the following bound:

$$|\lambda| \leqq \|A+\varepsilon E\|_2 \leqq \tfrac{1}{4} + \varepsilon,$$

for any eigenvalue λ of $A+\varepsilon E$.

Let

$$G = \begin{bmatrix} B & & \lambda I - A \\ 0 & \cdots\ 0 & \frac{1}{4} \end{bmatrix} \tag{5}$$

be an $(n+1)\times(n+1)$ upper triangular matrix. It suffices to show that G is nonsingular and remains so under any such ε-perturbation of A, B, and $\lambda \leqq \frac{1}{4}+\varepsilon$ [8, Thm 2.4–9]. With a suitable use of the triangular inequality, we see (with $\varepsilon \leqq \frac{1}{4}$)

$$\|G\|_2 \leqq 1 \tag{6}$$

for any eigenvalue λ of $A+\varepsilon E$. Since G is upper triangular, we may obtain the inequality

$$\varepsilon < |b_1 a_{21} \cdots a_{n,n-1}\tfrac{1}{4}| = |\det(G)| = |\det(\Sigma)| \tag{7}$$

where $G = U\Sigma V^T$ is the singular value decomposition of G. Here U, V are orthogonal, $\Sigma = \operatorname{diag}(\sigma_1, \cdots, \sigma_{n+1})$ where

$$1 \geqq \sigma_1 \geqq \sigma_2 \geqq \cdots \geqq \sigma_n \geqq \sigma_{n+1} \geqq 0.$$

From (7) we have a bound on the smallest singular value of G:

$$\sigma_{n+1} \geqq (\sigma_1 \cdots \sigma_n)\sigma_{n+1} = |\det(G)| > \varepsilon. \tag{8}$$

Hence one would have to perturb G by a matrix of norm at least ε to make it singular.

5. Applying the theorem. To see how to apply the theorem to a given system that may not be completely controllable, consider the computed transformed single input system (4a). We apply the theorem to the controllable part of (4a)

$$\dot{z}_1 = (A'_{11}+\varepsilon E'_{11})z_1 + (B'_1+\varepsilon F'_1)u, \tag{9a}$$

where $A'_{11}+\varepsilon E'_{11}$ is $r\times r$, z_1 is an r-vector, and r is the dimension of the computed controllable space S_c. Denote

$$\xi = \|A'_{11}+\varepsilon E'_{11}\|_2 + \|B'_1+\varepsilon F'_1\|_2. \tag{9b}$$

If $\xi < \frac{1}{4}$, then the quantity

$$\mu_s = |b_1 a_{21} \cdots a_{r,r-1}| \tag{9c}$$

is a lower bound on the size of the perturbations needed to obtain a space S_c of smaller dimension. If the system (9a) does not satisfy $\xi < \frac{1}{4}$, then we must scale (9a) by a factor of $\frac{1}{4}\xi$ in order to apply the theorem, so that the measure becomes

$$\mu_s = \frac{1}{4\xi^n}|b_1 a_{21} \cdots a_{r,r-1}|. \tag{9d}$$

In the block case, we define μ_s to be the product of the smallest singular values for each sub-diagonal block corresponding to the elements $b_1, a_{21}, \cdots, a_{r,r-1}$ that appear in (9c). These blocks will be rectangular of nonincreasing sizes. Detailed description of the block Staircase Algorithm can be found in [2], [4], [9], or [10]. This definition is not based on a specific proven result, but was used as an experimental extension to the definition for the single input case. The experimental behavior of the two definitions were very similar.

6. Error bounds. Once we have an indication that the computed orthogonal basis Q_1 for the controllable space S_c is *robust*, we may consider only perturbations that do not change the dimension of the controllable space. We consider the transformations

$Q=[Q_1\ Q_2]$, $Q+\Delta Q=[Q_1+\Delta Q_1\ Q_2+\Delta Q_2]$ obtained using the Staircase Algorithm from the original system (1) and the perturbed system (4b), where we assume the perturbation applied is such that $\dim Q_1=\dim Q_1+\Delta Q_1$. We want to find a bound η on $|\tan\phi|$, where $|\phi|$ is the largest angle (in the sense of [3]) between the spaces span (Q_1) and span $(Q_1+\Delta Q_1)$. We can give an idea of η by observing that S_c can be thought of as the smallest invariant subspace of the transformation A that contains the vectors B. Hence, we can estimate η with a closely related quantity $\tau=|\tan\theta|$, where θ is the largest angle between the computed space span $(Q_1+\Delta Q_1)$ and the nearest invariant subspace of A.

To give a bound for τ, we need to define some additional quantities: we define the notation

$$\operatorname{sep}(M, N)=\|T^{-1}\|_2^{-1}$$

where M, N are some given matrices and T is a matrix operator defined by

$$T(X)=MX-XN,$$

henceforth called the Lyapunov operator. If A'_{11}, A'_{22} denote the controllable, uncontrollable parts of $A'=Q^TAQ$ in the system (3a) obtained when (1) is put into canonical form, then δ is defined as

$$\delta=\operatorname{sep}(A'_{11}, A'_{22})-\varepsilon.$$

If A is perturbed by a matrix of norm less than ε, and assuming ε is small enough to satisfy $4\varepsilon(\|A'_{12}\|_2+\varepsilon)\leqq[\operatorname{sep}(A'_{11}, A'_{22})-\varepsilon]^2$, then we may apply [3, Thm 4.11] to obtain the bound

$$\tau\leqq 2\tfrac{\varepsilon}{\delta}. \tag{10}$$

Note this does not give a complete answer, since the invariant subspace of A nearest to span $(Q_1+\Delta Q_1)$ may not contain the vectors B at all. However, since $S_c=\operatorname{span}(Q_1)$ is an invariant subspace of A, τ must be a lower bound for η. Hence if τ is large, our original problem must be ill-posed in that small ε-perturbations in A, B result in large changes to the resulting controllable space. We must point out that (10) may give an extremely poor estimate for τ, especially if the two parts A_{11}, A_{22} have eigenvalues that coincide exactly or approximately, resulting in infinite or almost infinite values for δ.

7. Limitations. In applying theorem 1, there are several limitations that should be mentioned. The most obvious is that with the required scaling $\|A\|+\|B\|\leqq\frac{1}{4}$, for large systems, it is easy to obtain small values of μ_s even for reasonably well-conditioned systems (here we mean systems with a reasonably well-conditioned eigenvector matrix). This property shows up in the numerical experiments.

In the block case (multiple inputs), we defined μ_s as the product of the smallest singular values, one per subdiagonal block. Since typically there are fewer blocks than there would be in the single input case (for systems of comparable size), the above limitation may not be as acute in the multiple input case.

8. Numerical experiments. We ran two general sets of tests. First we ran one with a sequence of matrices with increasingly worse conditioning, in the sense of the eigenvector matrices. Second we ran one with increasing sizes.

To construct the examples with increasing ill-conditioning, we took a single example in canonical form (already split) with $A=\operatorname{diag}(-5\ -4\ -3\ -2\ -1\ 1\ 2\ 3\ 4\ 5\ 6)$ and B a random 11×4 matrix with 4 rows all zeros. We then applied

a series of random similarity transformations, progressively more and more badly-conditioned.

We constructed the examples of increasing size similarly: by starting with systems in canonical form and applying similarity transformations to them. In this case, $A = \text{diag}(m_1, \cdots, m_n)$, where $n =$ size of A, and m_i is a random set of distinct integers chosen from $-n \leqq m_i \leqq +n$, and $B = (1\ 0\ 1\ 0\ \cdots)^T$ (one column only). The examples in Table 2b were constructed as in Table 2a, except that the similarity transformations were especially constructed to have a 2-norm condition number of 100 in all cases.

The values of $1/\delta$ were obtained by estimating the condition number (in the sense of solving linear equations) of the Lyapunov operator using the computed A'_{11}, A'_{22}. The estimate was obtained using a method adapted from that described in [7] for estimating condition numbers of general linear maps.

The computations for Tables 1, 2a, were carried out in double precision on an IBM 370/168, which has a precision of about 10^{-15}. Since the basis of the Staircase Algorithm is searching for a small sub-diagonal block in the transformed matrix A', the decision as to when a block is small (or rank deficient) is critical to the success of the algorithm. We chose experimentally to use the square root of the machine precision, specifically 10^{-7}, as the zero tolerance. The computations for Table 2b were carried

TABLE 1
Increasingly worse conditioning, size 11×11.

$\|A\|_\infty$	μ_s	$1/\delta$
13.05	2.09×10^{-4}	7.09
200.7	5.43×10^{-5}	11.97
5751	3.63×10^{-4}	9.73×10^{4}
8424	1.76×10^{-4}	4.03×10^{5}

TABLE 2a
Increasing sizes, all single input.

size	$\|A\|_\infty$	μ_s	$1/\delta$	secs
4	51.7	2.72×10^{-2}	2.59	.01
8	158.6	7.23×10^{-12}	5.03	.01
16	3190	8.08×10^{-21}	115.0	.03
24	793.1	2.18×10^{-31}	28.4	.08
32	970.7	1.73×10^{-54}	48.1	.17
40	5709	9.21×10^{-49}	53.8	.32
48	5269	1.82×10^{-58}	50.6	.54

TABLE 2b
Increasing sizes, all single input (second set, run on VAX).

size	$\|A\|_\infty$	μ_s	$1/\delta$
4	791.5	2.30×10^{-4}	62.4
8	373.4	1.48×10^{-8}	611
16	246.7	3.77×10^{-15}	159
24	772.4	5.30×10^{-25}	196
32	1578	8.46×10^{-38}	1008
40	2781	7.09×10^{-41}	2787
48	2274	2.70×10^{-47}	607

out on a VAX 11/780 running UNIX. Since numbers less than 10^{-38} cannot be represented on a VAX, the entries in the table that were that small were recomputed by hand (the computer generated zeros).

Since the example in Table 1 had multiple inputs, and the example in Table 2 had only a single input, the number of sub-diagonal blocks in example 2 was much greater for comparable size; hence the estimate μ_s was much smaller. Clearly, though the estimate is of some use for smaller systems, for larger systems a more precise measure must be computed. We will discuss possibilities in this regard in a future paper.

To obtain an estimate on the bound η of the errors in the computed controllable space S_c, recall formula (10) to note that one must multiply the $1/\delta$ column by ε, in this case, 10^{-7}. We find that this estimate is small, indicating small errors, except in cases where we specifically built the test case to be badly-conditioned.

As is evident from these tables, the rank determination was not sensitive to conditioning per se as much as it was to size. Just the opposite was true for the bounds on the possible subspace perturbations (the $1/\delta$ column).

9. Concluding remarks. This algorithm provides a stable way to compute the controllable space and is a first attempt at giving an estimate of the rank-robustness of this problem. It is always better than using (2), since the singular values of the matrix in (2) are unchanged by orthogonal similarity transformations. We would obtain the same set of singular values whether we use the original system (1) or the transformed system (3) in canonical form. The example mentioned in the introduction illustrates that (2) is an example of a procedure which can introduce ill-conditioning not present in the original problem. On the other hand, the Staircase Algorithm is an example of a method that is robust in the sense that, though it might fail on certain problems, it does not add ill-conditioning not present in the original problem and will provide flags μ_s, τ to signal a possible failure.

Acknowledgment. The author is deeply grateful for the detailed and helpful comments from the reviewers.

REFERENCES

[1] R. E. KALMAN, *Mathematical description of linear systems*, SIAM J. Control Optim. 1 (1963), pp. 152–192.

[2] D. L. BOLEY, *Computing the controllability/observability of a linear time-invariant dynamic system, a numerical approach*, Ph.D. dissertation, Computer Science rep. STAN-CS-81-860, Stanford Univ., Stanford CA, June 1981.

[3] G. W. STEWART, *Error and perturbation bounds for subspaces associated with certain eigenvalue problems*, SIAM Rev., 15 (1973), pp. 727–764.

[4] C. C. PAIGE, *Properties of numerical algorithms related to computing controllability*, IEEE Trans. Automat. Control, AC-26, (1981), pp. 130–138.

[5] E. B. LEE AND L. MARKUS, *Foundations of Optimal Control Theory*, John Wiley, New York, 1967.

[6] J. H. WILKINSON, *The Algebraic Eigenvalue Problem*, Clarendon Press, Oxford, 1965.

[7] A. K. CLINE, C. B. MOLER, G. W. STEWART AND J. H. WILKINSON, *An estimate for the condition number of a matrix*, SIAM J. Numer. Anal., 16 (1979), pp. 368–375.

[8] T. KAILATH, *Linear Systems*, Prentice-Hall, Englewood Cliffs, NJ, 1980.

[9] P. VAN DOOREN, *The generalized eigenstructure problem in linear systems theory*, IEEE Trans. Automat Control., AC-26 (1981), pp. 111–130.

[10] P. VAN DOOREN, E. EMAMI-NAEINI AND L. SILVERMAN, *Stable extraction of the Kronecker structure of pencils*, Proc. 17th IEEE Conference on Decisions and Control, Jan. 1979, pp. 521–524.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 1, January 1985

008

BOUNDS FOR CUBE COLORING*

BARTON R. PLUMSTEAD† AND JOAN B. PLUMSTEAD‡

Abstract. An n-cube is properly colored if each vertex having an even number of ones is colored white and each vertex having an odd number of ones is colored black. This paper considers programs that color the n-cube with a coloring operation that in one step colors all uncolored vertices of a subcube either all black or all white and leaves previously colored vertices as before. An upper bound of $1.06(\sqrt[3]{5})^n$ steps and a lower bound of $\frac{4}{3}(1.5)^n$ steps for coloring the n-cube are proved. There are relationships between this model of computation and both width-two branching programs and depth 3 circuits for parity.

AMS(MOS) subject classification. 68-05

Introduction. An n-cube is the set of all 2^n n-dimensional binary vectors. The cube is said to be *properly colored* if each vector with an even number of ones is colored white and each vector with an odd number of ones is colored black. The set of vectors defined by fixing $0 \leqq k \leqq n$ components of the vectors is called a *subcube*. In our model, each primitive step of an algorithm to color the n-cube consists of specifying a single subcube and coloring it black or white. Thus, an algorithm for coloring an n-cube can be given as a sequence of ordered pairs, specifying which subcube is colored and what color it is given. Initially, no vectors are colored, and once a vector has been given a color, it keeps that color even if it is an element of a subcube colored by a later step. Figure 1 shows a simple algorithm for coloring an n-cube:

for each vector x with an even number of ones **do**
    color x white

color the entire cube black.

FIG. 1. *Straightforward coloring algorithm.*

This algorithm is obviously correct and takes $2^{n-1}+1$ steps.

In the first section, this trivial upper bound is improved to $1.06(\sqrt[3]{5})^n$ steps, and in the second section a lower bound of $\frac{4}{3}(1.5)^n$ steps is proved. For comparison, note that $\sqrt[3]{5}$ is approximately 1.709. We discuss some possibilities for narrowing this gap between the two bounds in the third section. Finally, in the fourth section, we discuss the relationships between this problem and other models of computation, specifically width-two branching programs and bounded depth circuits.

We introduce some notation for describing algorithms which color n-cubes. We will use a ? in a particular component to indicate that the component is not fixed (i.e. it can take on both 0 and 1 as values). If every component of a subcube is fixed, we call that subcube a *vector*. A subcube is *even* if it is a vector and it has an even number of ones, and *odd* if it is a vector and it has an odd number of ones. If C is a subcube of an n-cube and D is a subcube of an m-cube, then CD denotes the subcube of the $(n+m)$-cube defined by letting the first n components correspond to the components in C and the last m correspond to D. Similarly, if $C_1, C_2, \cdots, C_k$ are subcubes of the

* Received by the editors March 15, 1983, and in revised form September 6, 1983.

† Department of Mathematics and Computer Science, San Jose State University, San Jose, California 95192.

‡ Computer Science Division, University of California at Berkeley, Berkeley, California 94720. The research of this author was supported by DARPA under grant N-00039-82-C-0235.

m-cube, $C_1C_2 \cdots C_k$ is a subcube of the km-cube. An algorithm to color the n-cube will be written $(C_1, a_1); (C_2, a_2); \cdots ; (C_s, a_s)$, where C_i is the subcube colored at step i, and a_i is the color it is given. In constructing an algorithm Q, the statement $Q \leftarrow \Lambda$ will mean that initially Q contains no steps. Then $Q \leftarrow Q; (C, a)$ will add one step to Q, and in that step the subcube C will be given the color a.

1. Upper bounds. One can improve upon the straightforward algorithm for coloring the n-cube by using the algorithm in Fig. 2. Suppose $n = km$ where k and m are both integers. Then each subcube C can be written as $C_1C_2 \cdots C_k$, where each C_i is a subcube of the m-cube.

```
for j := 0 to k do
    for each C such that j of the C_i's are {?}^m
            and k − j of the C_i's are even do
    begin
        if j is even then color C white
        else color C black
    end
end
```

FIG. 2. *The k-parts coloring algorithm.*

For example, if this were run on the 4-cube, with $k = 2$ it would perform the following steps:

$j = 0$) Color $\langle 0000 \rangle$, $\langle 0011 \rangle$, $\langle 1100 \rangle$, and $\langle 1111 \rangle$ white.

$j = 1$) Color $\langle 00?? \rangle$, and $\langle 11?? \rangle$ black.
Color $\langle ??00 \rangle$, and $\langle ??11 \rangle$ black.

$j = 2$) Color $\langle ???? \rangle$ white.

The 2-parts coloring algorithm was discovered by Maria Klawe. Notice that the algorithm in Fig. 1 is the k-parts algorithm for $k = 1$.

THEOREM 1. *If $n = km$ where k and m are both integers, the k-parts coloring algorithm colors the n-cube in $(2^{m-1}+1)^k$ steps.*

Proof. It is easy to show by induction that when $j = j'$, this algorithm colors exactly those vectors $y = y_1y_2 \cdots y_k$ such that j' of the y_i's are odd and $k - j'$ are even, where each $Y_i \in \{0, 1\}^m$. If j' is even, there are an even number of odd subvectors, so the vector is even and the algorithm correctly colors it white. If j' is odd, the number of odd subvectors is odd, making the vector odd, and the algorithm correctly colors the vector black. Thus the algorithm is correct. The number of steps used is $\sum_{j=0}^{k} \binom{k}{j}(2^{m-1})^{k-j} = (2^{m-1}+1)^k$ steps. □

When $m = 3$ the k-parts algorithm requires only $(\sqrt[3]{5})^n$ steps, which is less than what the algorithm uses when $m = 2$ or $m = 4$. Generalizing the method used in the k-parts algorithm, we can obtain the following theorem:

THEOREM 2. *If an n-cube can be colored in S_n steps and an m-cube in S_m steps, then an $(n+m)$-cube can be colored in $S_n \cdot S_m$ steps.*

Proof. Suppose algorithm P_n colors the n-cube in S_n steps and algorithm P_m colors the m-cube in S_m steps. Then P_n has the form $(C_1, a_1); (C_2, a_2); \cdots ; (C_{S_n}, a_{S_n})$, where C_i is the subcube colored at step i, and a_i is the color the subcube is given. Similarly, P_m can be written as $(D_1, b_1); (D_2, b_2); \cdots ; (D_{S_m}, b_{S_m})$, where the D_i's are subcubes and the b_i's colors. Consider the following algorithm.

```
Q ← Λ
for i := 1 to S_n do
begin
    for j := 1 to S_m do
    begin
            if a_i = b_j then Q ← Q; (C_iD_j, white)
                         else Q ← Q; (C_iD_j, black)
    end
end
```

For each step in P_n, Q does S_m steps, so this algorithm does $S_n \cdot S_m$ steps. Consider an arbitrary vector $y \in \{0,1\}^{n+m}$. One can write y as $y_1 y_2$ where $y_1 \in \{0,1\}^n$ and $y_2 \in \{0,1\}^m$. Suppose y_1 was first colored at step i in P_n and y_2 at step j in P_m. Then $y \notin C_{i'} D_{j'}$ for $i' < i$, or $i' = i$ and $j' < j$. Hence, in Q, y is first colored when $C_i D_j$ is colored. If $a_i = b_j$ then y_1 and y_2 are either both even or both odd, so y is even and is thus colored correctly. If $a_i \neq b_j$ then one of y_1 and y_2 is even and the other odd, so y is odd and is colored correctly. Thus the algorithm is correct. □

As mentioned earlier, when $n \equiv 0 \bmod 3$, theorem 1 gives an upper bound of $(\sqrt[3]{5})^n$. Since the 2-cube can be colored in three steps and the 4-cube in nine steps, Theorems 1 and 2 give us upper bounds of $9(\sqrt[3]{5})^{n-4}$ when $n \equiv 1 \bmod 3$, and $3(\sqrt[3]{5})^{n-2}$ when $n \equiv 2 \bmod 3$. Thus $1.06(\sqrt[3]{5})^n$ is an upper bound for the cube coloring problem.

2. An exponential lower bound. We next turn our attention to proving an exponential lower bound for coloring the n-cube. Suppose that any algorithm for coloring the n-cube requires at least S_n steps and let n be greater than 1. We will show that $S_n \geqq \frac{3}{2} S_{n-1}$ by looking at an arbitrary algorithm P for coloring the n-cube. Suppose P requires s steps. Then P has the form $(C_1, a_1); (C_2, a_2); \cdots; (C_s, a_s)$, where C_i is the subcube colored at step i, and a_i is the color it is given. Look at the first component of C_i. It is either 0, 1, or ?, so C_i can be written as $0C_i'$, $1C_i'$, or $?C_i'$, where C_i' is a subcube of the $n-1$-cube. Let P_0 and P_1 be the algorithms defined by the subsequences of P which include exactly those steps i for which the first component of C_i is a 0 or a 1, respectively. More formally, P_0 and P_1 can be defined as in Fig. 3.

```
P_x ← Λ
for i := 1 to s do
begin
    if C_i = xC_i' then P_x ← P_x; (C_i, a_i)
end
```

FIG. 3. *Algorithm* P_x.

Similarly, let $P_0^?(P_1^?)$ be the algorithms defined by the subsequences of P which include exactly those steps i for which the first component of C_i is a 0 or a ? (1 or ?). Then we have Fig. 4.

```
P_x^? ← Λ
for i := 1 to s do
begin
    if C_i = xC_i' or C_i = ?C_i' then P_x^? ← P_x^?; (C_i, a_i)
end
```

FIG. 4. *Algorithm* $P_x^?$.

Let $|P|$ denote the length of P (the number of steps P uses). Then $|P| = |P_0^?| + |P_1| = |P_1^?| + |P_0|$. First we note that $P_0^?$ and $P_1^?$ can both be easily modified to color the $(n-1)$-cube. For $P_0^?$ one can simply strip off the first component of every subcube

used, and for $P_1^?$ one can simply strip off the first component and switch all the colors. Thus $|P| \geqq S_{n-1} + \max(|P_0|, |P_1|)$.

Next we will show that $|P_0| + |P_1| \geqq S_{n-1}$. Consider the algorithm in Fig. 5.

```
Q ← Λ
for i := 1 to s do
begin
    if C_i ≠ ?C'_i then
        if C_i = 0C'_i then Q ← Q; (C'_i, a_i)
        else if a_i = white then Q ← Q; (C'_i, black)
            else Q ← Q; (C'_i, white)
end
```

FIG. 5. *Algorithm Q.*

THEOREM 3. *Algorithm Q colors the* $(n-1)$*-cube.*

Proof. Consider a vector y in the $(n-1)$-cube, and look at the first time that $0y$ or $1y$ was colored in algorithm P. Say this occurred in step i. Then $y \notin C'_j$ for any $j < i$. If $C_i = 0C'_i$, then since y has the same parity as $0y$, y is colored correctly. Similarly, if $C_i = 1C'_i$, then since y has the opposite parity of $1y$, y is colored correctly. If $C_i = ?C'_i$, then P colors $0y$ and $1y$ the same color, but we assumed that P was correct, so this cannot happen. Thus Q correctly colors the $(n-1)$-cube. □

Obviously, the number of steps in Q is $|P_0| + |P_1|$. Hence, $\max(|P_0|, |P_1|) \geqq \frac{1}{2} S_{n-1}$ and $|P| \geqq \frac{3}{2} S_{n-1}$. Since $S_1 = 2$ and $S_2 = 3$, we have proved the following lower bound:

THEOREM 4. *Any algorithm which colors the n-cube has length at least* $2(\frac{3}{2})^{n-1}$.

3. Open problems. Since the upper and lower bounds proven here do not meet, at least one of them is not tight. Which one? In the algorithms given for the upper bound, if a subcube C is specified and the first component of C is fixed, then the second and third components are also fixed. Suppose that for some optimal algorithm P for coloring the n-cube, there exist j and k where $1 \leqq j, k \leqq n$ such that whenever a subcube C is specified and the jth component is fixed, the kth component is also fixed. Given this assumption, it is not difficult to prove a better lower bound. One can always renumber the components, so there exists an algorithm P' such that whenever the first component is fixed, the second is also. If one defines P_0 and P_1 from algorithm P' as they were defined in the proof of Theorem 4, then we have $|P'| \geqq S_{n-1} + \max(|P_0|, |P_1|)$. If a subcube C_i specified in P' is in P_0 or P_1, it can be written as $00C'_i$, $01C'_i$, $10C'_i$, or $11C'_i$, where C'_i is a subcube of the $(n-2)$-cube. Consider algorithms Q_0 and Q_1 which are defined in Fig. 6.

```
Q_0 ← Λ
Q_1 ← Λ
for i := 1 to s do
begin
    if C_i ≠ ?C'_i then
        if C_i = 00C'_i then Q_0 ← Q_0; (C'_i, a_i)
        if C_i = 10C'_i then
            if a_i = white then Q_0 ← Q_0; (C'_i, black)
            else Q_0 ← Q_0; (C'_i, white)
    if C_i = 01C'_i then
        if a_i = white then Q_1 ← Q_1; (C'_i, black)
            else Q_1 ← Q_1; (C'_i, white)
        if C_i = 11C'_i then Q_1 ← Q_1; (C'_i, a_i)
end
```

FIG. 6. *Algorithms* Q_0 *and* Q_1.

An argument similar to that in Theorem 3 shows that the algorithms Q_0 and Q_1 each color the $(n-2)$-cube. Obviously $|Q_0|+|Q_1|=|P_0|+|P_1|$. Hence $\max(|P_0|, |P_1|) \geqq S_{n-2}$, so $|P'| \geqq S_{n-1}+S_{n-2}$, and thus $S_n \geqq \varphi^{n+2}/\sqrt{5}$, where $\varphi=(1+\sqrt{5})/2$.

One can use a similar argument to show that if, in an optimal algorithm P for coloring the n-cube, whenever the first component is fixed, the second and third are also fixed, then $S_n \geqq S_{n-1}+2S_{n-3}$, so S_n is $\Omega(1.6956^n)$.

4. Related models of computation. The problem of finding a good (i.e., exponential) lower bound for the number of colors in an n-cube has been proposed in [BDFP] in relation to branching programs and striped cubes. Branching programs are studied to develop techniques for proving lower bounds for storage space. In [BDFP] they look at lower bounds on the length of width-two branching programs (W2-programs) for computing certain Boolean functions. An arbitrary W2-program can be decomposed as a sequence of strict width-two branching programs. It is shown in [BDFP] that if there is a strict width-two branching program for computing f, a Boolean function of n variables, then $f^{-1}(x)$ is the disjoint union of no more than $4\cdot 2^{n/2}-2$ striped cubes, where a striped cube is the subset of a subcube obtained by specifying the parity of some set of components. One can look at algorithms, called P-programs, which are sequences of ordered pairs, specifying which striped cube is colored and which color it is given. If a P-program colors $f^{-1}(0)$ white and $f^{-1}(1)$ black, we say it computes f. For example, the parity function can be computed with a P-program of length 2. Consider the algorithm in Fig. 1. Since the set of vectors with an even number of ones is a striped cube, it can be colored in one step, and the remainder can be colored in one more step. If the shortest P-program for computing some function f requires $C_P(f)$ steps, then $C_P(f)/(4\cdot 2^{n/2}-2)$ is a lower bound on the length of any W2-program for computing f. Thus one could look at P-programs in trying to prove lower bounds for W2-programs. The problem of finding a natural function f and a nonpolynomial lower bound on the length of the W2-program for computing f was open at the time this work was done.

In order to gain insight into this problem, we looked at the parity function in a more restricted model. We call a P-program which only uses subcubes, rather than the more general striped cubes a *restricted P-program.* We prove a lower bound of $\frac{4}{3}(1.5)^n$ steps to color the n-cube, and thus on the length of one of these restricted P-programs for computing the parity function. After these results were obtained, James Shearer [S] showed that the function $f(x_1, x_2, \cdots, x_n)$, which is 1 if and only if $x_1+x_2+\cdots+x_n \equiv 0 \bmod 3$, requires a P-program of length exponential in n. Interestingly, $(1.5)^n$ comes into his result also.

The problem of computing the parity function has also been considered with Boolean circuits of constant depth. In [FSSa], it is shown that constant depth Boolean circuits for parity require more than a polynomial number of gates. In [FSSb], this is improved to $\Omega(n^{c\log_2 n})$ for depth 3 circuits. It seems likely that a significantly larger lower bound is possible. The parity function is important in this situation as other problems such as multiplication and transitive closure are at least as difficult as parity. In addition, sufficiently large lower bounds for constant depth parity circuits would show the existence of an oracle A separating PSPACE^A from the relativized Meyer-Stockmeyer hierarchy $\bigcup_i \sum_i^{P,A}$.

In fact, there are more direct connections between depth 3 circuits and our model of computation. Consider programs which first color no more than $2^n-(2^{n-1}/f(n))$ vectors black, then have a sequence of steps in which subcubes are colored white, and finally have one last black coloring step. If any such program which properly colors

the n-cube contains at least $f(n)$ white coloring steps, then any depth 3 parity circuit must contain more than $f(n)$ gates. One sees this by looking at a 3 level and-or-and circuit. If the circuit outputs a 1 for vectors with even parity and a 0 for vectors with odd parity, then every or-gate must output a 1 on every even vector, and at least one or-gate must output a 0 on any given odd vector. At least one or-gate must output a 0 for at last an average number of odds, namely $2^{n-1}/f(n)$ odds. The correspondence stated above follows when one views the bottom level of and-gates as subcubes corresponding to the white coloring steps.

Our model of computation may be considered as a restricted class of circuits, namely those circuits in which only the rightmost gate on any level may have nonliteral inputs.

Acknowledgments. We would particularly like to thank Faith Fich for bringing this problem to our attention, for discovering holes in earlier proofs, and for helping to develop notation. We would also like to thank David Shmoys and Alice Wong for comments on earlier versions of the paper.

REFERENCES

[BDFP] A. BORODIN, D. DOLEV, F. FICH AND W. PAUL, *Bounds for width-two branching programs*, Proc. 15th ACM Symposium on Theory of Computing 1983, pp. 87–93.

[FSSa] M. FURST, J. B. SAXE AND M. SIPSER, *Parity circuits and the polynomial time hierarchy*, Proc. 22nd Symposium of the Foundations of Computer Science, 1981, pp. 260–270.

[FSSb] M. FURST, J. B. SAXE AND M. SIPSER, *Depth 3 circuits require* $\Omega(n^{c \log n})$ *gates to compute parity: a geometric argument*, in preparation.

[S] J. SHEARER, *Letter to Wolfgang Paul*, February 8, 1983.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 1, January 1985

RANKING THE VERTICES OF A PAIRED COMPARISON DIGRAPH*

MIKIO KANO† AND AKIO SAKAMOTO‡

Abstract. A paired comparison digraph $D=(V, A)$ is a weighted digraph in which the sum of the weights of arcs, if any, joining two distinct vertices is exactly one; otherwise, there exist no arcs joining them. A one-to-one mapping α from V onto $\{1, 2, \cdots, |V|\}$ is called a ranking of D. We define the backward arcs and the backward length of α. An optimal ranking of D is a ranking whose backward length is minimum among those of all rankings of D. Our method of ranking the vertices of D is one that makes use of these optimal rankings. For certain classes of paired comparison digraphs, we show that the optimal rankings can be explicitly computed.

AMS(MOS) subject classifications. 05C20, 05C99

1. Introduction. Consider a weighted digraph in which every arc vw has a weight $\varepsilon(vw)$. We shall be concerned with a *paired comparison digraph* (PCD) which is defined to be a weighted digraph satisfying the following conditions (see Fig. 1):

(i) $0<\varepsilon(vw)\leqq 1$ for every arc vw;

(ii) $\varepsilon(vw)+\varepsilon(wv)=1$ if both vw and wv are arcs;

(iii) $\varepsilon(vw)=1$ if vw is an arc but wv is not.

A digraph D can be considered as a PCD if we set the weight of every arc of D as follows:

(i) $\varepsilon(vw)=0.5$ if both vw and wv are the arcs;

(ii) $\varepsilon(vw)=1$ if vw is an arc but wv is not.

Throughout this paper, we regard a digraph as a PCD in this way. In particular, a *tournament*, in which every two vertices are joined by exactly one arc, is a PCD.

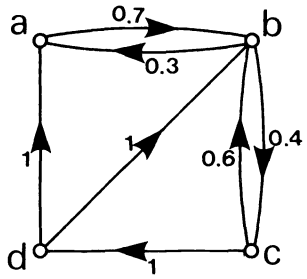


FIG. 1. *A paired comparison digraph.*

A digraph D is a natural way of representing the results of paired experiments; that is, if v is superior to w (v defeats w), then vw is an arc of D but wv is not; if v is equivalent to w (game ends in a draw), then both vw and wv are arcs of D; and if v and w are not compared (game is not played), then there are no arcs joining v and w. Furthermore, we may interpret the weight $\varepsilon(vw)$ of an arc vw in a PCD as the rate with which certain consumers prefer v to w in a paired comparison test (with which v defeats w).

We now explain the method of ranking the vertices which will be discussed in this paper. Let D be a PCD with n vertices. A *ranking* α of D is a one-to-one mapping from the set of vertices of D onto the set of integers $\{1, 2, \cdots, n\}$. For a ranking α,

* Received by the editors January 27, 1983, and in final revised form October 17, 1983.

† Department of Mathematics, Akashi Technological College, Uozumi, Akashi 674, Japan.

‡ Faculty of Engineering, Tokushima University, Minami-josanjima, Tokushima 770, Japan.

the image $\alpha(v)$ of a vertex v is called the *rank* of v defined by α, and an arc vw such that $\alpha(w) < \alpha(v)$ is called a *backward arc* of α. We write $B(\alpha)$ for the set of backward arcs of α, and define the *backward length* of α, denoted by $\|B(\alpha)\|$, as follows:

$$\|B(\alpha)\| = \sum_{vw \in B(\alpha)} \varepsilon(vw)\{\alpha(v) - \alpha(w)\}.$$

Then the backward length of a ranking α can be considered as the value of unreasonableness of α. On the other hand, an arc xy with $\alpha(x) < \alpha(y)$ and $\varepsilon(xy) = 1$ represents only that a player x with higher rank defeats a player y with lower rank.

For example, let α be a ranking of a PCD in Fig. 1 with $\alpha(a) = 1$, $\alpha(b) = 3$, $\alpha(c) = 4$, and $\alpha(d) = 2$, then it follows that $B(\alpha) = \{da, ba, cd, cb\}$ and $\|B(\alpha)\| = 1 \times 1 + 0.3 \times 2 + 1 \times 2 + 0.6 \times 1 = 4.2$ (see Fig. 2).

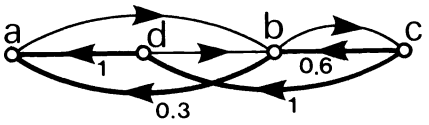


FIG. 2. *A ranking* α.

A ranking α of D is said to be *optimal* if the backward length of α is minimum among those of all rankings of D. For a vertex v of D, the average of the ranks defined by the optimal rankings of D is called the *proper rank* of v. Our method of ranking the vertices of D is one that makes use of these proper ranks which depend on the optimal rankings of D. Therefore our ranking procedure can be applied to every PCD, and we shall show that this method has reasonable properties.

There are several methods of ranking the vertices of a tournament. One approach is to compute the scores, which are the numbers of games won by each player, and compare them. In [8], this ranking method is called the "points system" and characterized by a set of axioms. Another ranking is obtained by making use of the maximum positive eigenvalue and its positive eigenvector, due to Perron and Frobenius, of the adjacency matrix of a tournament (e.g., Wei [9] and Kendall [6], [2, p. 185], Moon and Pullman [7], Berge [1, p. 74]). It follows in the former that the ranks of players whose scores are the same are not distinguished. On the other hand, the latter may discriminate the ranks of those. In many tournaments, these two methods give similar rankings provided the ranks of players having the same score are ignored. We show, however, an example of a tournament in which the vertex, whose rank is determined to be the first by the latter method, does not have a maximum score. It is given in Fig. 3. The maximum positive eigenvalue and its positive eigenvector of the adjacency matrix of this tournament are approximately 3.174 and [.165, .157, .180, .130, .119, .083, .083, .083] respectively. Thus the ranking of the vertices is in the order of c, a, b, d, $e, \cdots$. However, the scores of a and b are both five and that of c is four.

Note that the ranking methods mentioned above are only for a tournament. On the other hand, our aim in this paper is to consider a ranking procedure applicable to every PCD, including tournaments as a special case.

For a ranking α of a PCD D, we can define the forward arcs of α and its forward length similarly. A ranking α of D is defined to be forward optimal if its forward length is *maximum* among those of all rankings of D. The forward optimal rankings may be applied to rank vertices of D. But if we regard D as the results of games, then this ranking method seems not as natural as the backward case discussed in the present paper. However, the ranking method with forward length has different properties and may be useful for another application [4].

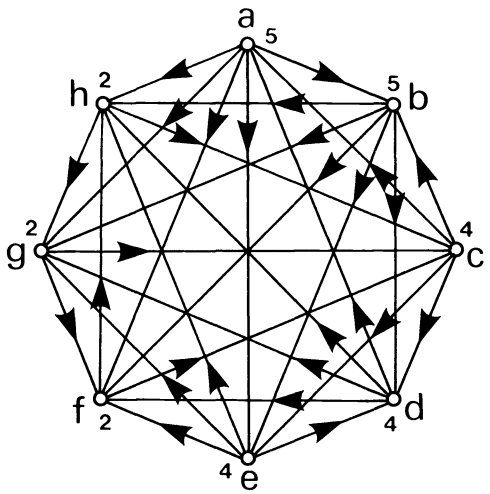


FIG. 3. *A tournament. Numbers denote the scores.*

2. Notation and preliminary results. For finite sets X and Y, we denote the number of elements in X by $|X|$ or $\#\{x \in X\}$, and denote $X \cup Y$ by $X + Y$ if it is a disjoint union. A digraph is said to be *asymmetric* if every two vertices are joined by at most one arc, and is said to be *complete* if every two vertices are joined by at least one arc. A complete asymmetric digraph is called a *tournament.* We say that a digraph is *acyclic* if it contains no oriented cycles.

Let $D = (V, A)$ be a PCD with n vertices. We define five functions; $\bar{\varepsilon}: V \times V \to [0, 1]$, $\mu: V \times V \to \{0, 1\}$, σ^+ and $\sigma^-: V \to [0, n-1]$, and $d^*: V \to \{0, 1, 2, \cdots, n-1\}$ as follows:

$$\bar{\varepsilon}(vw) = \begin{cases} \varepsilon(vw) > 0 & \text{if } vw \text{ is an arc of } D, \\ 0 & \text{otherwise,} \end{cases}$$

$$\mu(vw) = \mu(wv) = \bar{\varepsilon}(vw) + \bar{\varepsilon}(wv),$$

$$\sigma^+(v) = \sum_{x \in V} \bar{\varepsilon}(vx) \quad \text{and} \quad \sigma^-(v) = \sum_{x \in V} \bar{\varepsilon}(xv),$$

$$d^*(v) = n - 1 - (\sigma^+(v) + \sigma^-(v)) = n - 1 - \sum_{x \in V} \mu(vx).$$

It is obvious that if v and w are compared, then $\mu(vw) = 1$; otherwise, $\mu(vw) = 0$. $d^*(v)$ is the number of vertices which are not compared with v. Since we may regard $\sigma^+(v)$ as a generalized score of v, we call $\sigma^+(v)$ the *positive score* (or briefly *score*) of v and $\sigma^-(v)$ the *negative score* of v. Note that if D is an asymmetric digraph, then the positive and negative scores of v are the out-degree and in-degree of v, respectively.

We put the symbol of the digraph, say D, as a subscript of an appropriate function if necessary. For example, we write $\sigma^+_D(v)$.

If α is a ranking of D with $\alpha(v_i) = i$ for $1 \leqq i \leqq n$, then we write $\alpha = [v_1, v_2, \cdots, v_n]$. The vertex whose rank defined by a ranking α is k is denoted by $\alpha^{-1}(k)$. We denote the set of optimal rankings of D by $\mathrm{OR}(D)$. The backward length of an optimal ranking of D is called the *optimal backward length* of D and denoted by $l(D)$. Then we have

$$l(D) = \min_{\alpha} \{\|B(\alpha)\|\} \quad \text{and} \quad \mathrm{OR}(D) = \{\alpha \mid \|B(\alpha)\| = l(D)\}.$$

Note that if H is a subdigraph of D with $V(H) = V(D)$, then $l(H) \leqq l(D)$ since $B_H(\alpha) \subseteq B_D(\alpha)$ for every α.

Remark 2.1. A PCD D is acyclic if and only if $l(D)=0$.

Proof. The remark follows from the fact that an acyclic digraph contains at least one vertex of in-degree zero and that a subdigraph of an acyclic digraph is also acyclic.

Remark 2.2. The optimal backward length of a PCD D is the sum of those of the strongly connected components of D.

Proof. This remark can be proved similarly to Remark 2.1 by regarding each strongly connected component of D as a vertex of an acyclic digraph.

Let α be a ranking of a PCD D and let m and k be integers such that $1 \leqq k < k+m \leqq n$, where n is the number of vertices of D. Then we define a ranking α_m^k by

$$\alpha_m^k(v)=\begin{cases} k+m & \text{if } v=\alpha^{-1}(k), \\ k & \text{if } v=\alpha^{-1}(k+m), \\ \alpha(v) & \text{otherwise.} \end{cases}$$

LEMMA 2.3. *Let α be a ranking of a* PCD *with n vertices. If $\alpha(v)=k$ and $\alpha(w)=k+m$, then*

$$\begin{aligned} \|B(\alpha_m^k)\|-\|B(\alpha)\| &= m\Big(\sigma^+(v)-\sigma^+(w)+\sum_{\alpha(z)>k+m}\{\mu(wz)-\mu(vz)\}\Big) \\ &\quad+\sum_{k<\alpha(y)<k+m}\{\alpha(y)-k\}\{\mu(wy)-\mu(vy)\} \\ &= m\Big(\sigma^-(w)-\sigma^-(v)+\sum_{\alpha(x)<k}\{\mu(vx)-\mu(wx)\}\Big) \\ &\quad+\sum_{k<\alpha(y)<k+m}\{(k+m)-\alpha(y)\}\{\mu(vy)-\mu(wy)\}. \end{aligned}$$

Proof. By the definition, the backward length of α is expressed as

$$\|B(\alpha)\|=\sum_{\alpha(x)<\alpha(y)}\bar{\varepsilon}(yx)\{\alpha(y)-\alpha(x)\}$$

where the summation is over all pairs of vertices x and y satisfying $\alpha(x)<\alpha(y)$. Let $X=\{x\in V \mid \alpha(x)<k\}$, $Y=\{y\in V \mid k<\alpha(y)<k+m\}$, $Z=\{z\in V \mid k+m<\alpha(z)\}$, and $B=\{st\in B(\alpha) \mid \{s,t\}\cap\{v,w\}=\varnothing\}$. Then we have

$$\begin{aligned} \|B(\alpha)\| &= \sum_{x\in X}[\bar{\varepsilon}(vx)\{k-\alpha(x)\}+\bar{\varepsilon}(wx)\{(k+m)-\alpha(x)\}] \\ &\quad+\sum_{y\in Y}[\bar{\varepsilon}(yv)\{\alpha(y)-k\}+\bar{\varepsilon}(wy)\{(k+m)-\alpha(y)\}] \\ &\quad+\sum_{z\in Z}[\bar{\varepsilon}(zv)\{\alpha(z)-k\}+\bar{\varepsilon}(zw)\{\alpha(z)-(k+m)\}] \\ &\quad+m\bar{\varepsilon}(wv)+\sum_{st\in B}\bar{\varepsilon}(st)\{\alpha(s)-\alpha(t)\}. \end{aligned}$$

Moreover, $\|B(\alpha_m^k)\|$ is obtained from the above equation only by interchanging v and w. Hence we have

$$\begin{aligned} \|B(\alpha_m^k)\|-\|B(\alpha)\| &= m\Big(\sum_{x\in X}\{\bar{\varepsilon}(vx)-\bar{\varepsilon}(wx)\}+\sum_{y\in Y}\{\bar{\varepsilon}(vy)-\bar{\varepsilon}(wy)\} \\ &\qquad+\sum_{z\in Z}\{\bar{\varepsilon}(zw)-\bar{\varepsilon}(zv)\}+\bar{\varepsilon}(vw)-\bar{\varepsilon}(wv)\Big) \\ &\quad+\sum_{y\in Y}\{\alpha(y)-k\}\{\mu(wy)-\mu(vy)\}. \end{aligned}$$

On the other hand, it follows that

$$\sigma^+(v)-\sigma^+(w)=\sum_{x\in X}\{\bar{\varepsilon}(vx)-\bar{\varepsilon}(wx)\}+\sum_{y\in Y}\{\bar{\varepsilon}(vy)-\bar{\varepsilon}(wy)\}$$
$$+\sum_{z\in Z}\{\bar{\varepsilon}(vz)-\bar{\varepsilon}(wz)\}+\bar{\varepsilon}(vw)-\bar{\varepsilon}(wv).$$

These equations lead to the first equation of the lemma. The second one is obtained similarly by considering $\sigma^-(w)-\sigma^-(v)$.

3. Complete PCD and semicomplete PCD. We begin with the following lemma which gives us the backward length of any ranking of a complete PCD.

LEMMA 3.1. *Let K be a complete* PCD *with n vertices and let α be any ranking of K. Then*

$$\|B(\alpha)\|=\sum_{v\in V}\sigma^+(v)\alpha(v)-\tfrac{1}{6}n(n^2-1).$$

Proof. We prove the equation by induction on n. The basis, $n=1$, is obvious. Suppose that the equation holds for $n=k$, and let $n=k+1$. Let x be the vertex such that $\alpha(x)=n$, and put $W=V(K)\backslash\{x\}$. By the induction hypothesis on $K-x$, we have

$$\|B_K(\alpha)\|=\sum_{v\in W}\{\sigma^+_K(v)-\bar{\varepsilon}_K(vx)\}\alpha(v)-\tfrac{1}{6}k(k^2-1)+\sum_{v\in W}\bar{\varepsilon}_K(xv)\{n-\alpha(v)\}.$$

Since $\bar{\varepsilon}_K(xv)+\bar{\varepsilon}_K(vx)=1$, the last term can be expanded as follows;

$$\sum_{v\in W}\bar{\varepsilon}_K(xv)\{n-\alpha(v)\}=n\sum_{v\in W}\bar{\varepsilon}_K(xv)+\sum_{v\in W}\{\bar{\varepsilon}_K(vx)-1\}\alpha(v)$$
$$=\sigma^+_K(x)n+\sum_{v\in W}\bar{\varepsilon}_K(vx)\alpha(v)-\tfrac{1}{2}k(k+1).$$

Thus we obtain

$$\|B_K(\alpha)\|=\sum_{v\in W}\sigma^+_K(v)\alpha(v)+\sigma^+_K(x)n-\tfrac{1}{6}k(k+1)(k+2)$$
$$=\sum_{v\in V(K)}\sigma^+_K(v)\alpha(v)-\tfrac{1}{6}n(n^2-1).$$

This completes the proof.

By Lemma 3.1, the optimal rankings, the proper ranks, and the optimal backward length of a complete PCD are easily obtained as the following theorem. Remember that the proper rank of v, denoted by $\pi(v)$, is defined by

$$\pi(v)=\frac{1}{|\mathrm{OR}(D)|}\sum_{\alpha\in\mathrm{OR}(D)}\alpha(v).$$

THEOREM 3.2. *Let K be a complete* PCD *with n vertices. Then*

(1) *A ranking $\alpha=[v_1, v_2, \cdots, v_n]$ of K is optimal if and only if $\sigma^+(v_1)\geqq\sigma^+(v_2)\geqq\cdots\geqq\sigma^+(v_n)$.*

(2) *$\pi(v)=\xi+\frac{1}{2}(\eta+1)$ where $\xi=\#\{x\in V\,|\,\sigma^+(x)>\sigma^+(v)\}$ and $\eta=\#\{y\in V\,|\,\sigma^+(y)=\sigma^+(v)\}$.*

(3) *$l(K)=\sum_{v\in V}\sigma^+(v)\alpha(v)-\frac{1}{6}n(n^2-1)$ where $\alpha\in\mathrm{OR}(K)$.*

Proof. (1) Let $\alpha=[v_1, v_2, \cdots, v_n]$ be any optimal ranking of K. If $\sigma^+(v_i)<\sigma^+(v_j)$ for some $i<j$, then, by Lemma 2.3, we have $\|B(\alpha^i_{j-i})\|-\|B(\alpha)\|=(j-i)[\sigma^+(v_i)-\sigma^+(v_j)]<0$ as $\mu(xy)=1$ for all vertices x and y, which is a contradiction. Thus $\sigma^+(v_1)\geqq\sigma^+(v_2)\geqq\cdots\geqq\sigma^+(v_n)$. On the other hand, let $\beta=[w_1, w_2, \cdots, w_n]$ be a ranking which satisfies the condition $\sigma^+(w_1)\geqq\sigma^+(w_2)\geqq\cdots\geqq\sigma^+(w_n)$. Then, we have $\|B(\alpha)\|=\|B(\beta)\|$ by Lemma 3.1. Hence $\beta\in\mathrm{OR}(K)$.

(2) Put $\{y \in V \mid \sigma^+(y) = \sigma^+(v)\} = \{v = y_1, y_2, \cdots, y_\eta\}$ and $|\mathrm{OR}(K)| = r$. Then it follows from (1) that $\{\alpha(y_1), \alpha(y_2), \cdots, \alpha(y_\eta)\} = \{\xi+1, \xi+2, \cdots, \xi+\eta\}$ for all $\alpha \in \mathrm{OR}(K)$ and $\pi(v) = \pi(y_1) = \pi(y_2) = \cdots = \pi(y_\eta)$. Hence we have

$$\begin{aligned}\eta\pi(v) &= \pi(y_1) + \pi(y_2) + \cdots + \pi(y_\eta) \\ &= \frac{1}{r}\sum_{i=1}^{\eta}\sum_{\alpha\in \mathrm{OR}(K)} \alpha(y_i) = \frac{1}{r}\sum_{\alpha}\sum_{i} \alpha(y_i) \\ &= \frac{1}{r}\sum_{\alpha}\sum_{i} (\xi + i) = \frac{1}{r}\sum_{\alpha}\{\xi\eta + \tfrac{1}{2}\eta(\eta+1)\} = \xi\eta + \tfrac{1}{2}\eta(\eta+1).\end{aligned}$$

Hence $\pi(v) = \xi + \frac{1}{2}(\eta+1)$.

Statement (3) follows at once from Lemma 3.1.

Consider, for example, a tournament T in Fig. 4. By Theorem 3.2, we have the following: $\mathrm{OR}(T) = \{\alpha = [v_1, v_2, \cdots, v_7] \mid v_1 = a, \{v_2, v_3\} = \{b, c\}, \{v_4, v_5, v_6\} = \{d, e, f\}$, and $v_7 = g\}$, $|\mathrm{OR}(T)| = 12$, $\pi(a) = 1$, $\pi(b) = \pi(c) = 2.5$, $\pi(d) = \pi(e) = \pi(f) = 5$, $\pi(g) = 7$, and $l(T) = 7$.

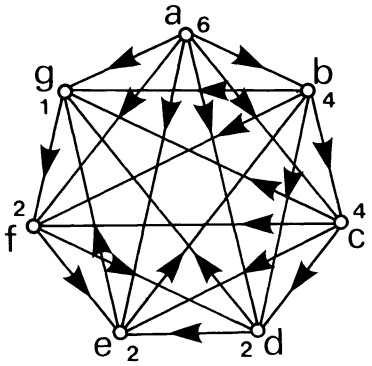


FIG. 4. *A tournament T. Numbers denote the scores.*

If two vertices v and w in a PCD D are not compared (i.e., $\mu(vw) = 0$), then the unordered pair $\{v, w\}$ is called an *uncompared pair* of D. We write $U(D)$ for the set of uncompared pairs of D. If $\{v, w\} \in U(D)$, then D_{vw} denotes the PCD obtained from D by adding a new arc vw of weight one, that is, $D_{vw} = (V(D), A(D) + vw)$. If $\{x, y\}$ is another uncompared pair of D, then $D_{vw,xy}$ is the PCD obtained from D_{vw} by adding a new arc xy of weight one. A complete PCD obtained from D by adding exactly one of arcs vw and wv for each $\{v, w\} \in U(D)$ is called a *completeness* of D. A completeness K of D with $l(K) = l(D)$ is called a *normal completeness* of D, and the set of normal completenesses of D is denoted by $\mathrm{NC}(D)$. Note that we can easily see the existence of a normal completeness of D as follows: Let $\alpha \in \mathrm{OR}(D)$. For every $\{v, w\} \in U(D)$, we add a new arc xy to D such that $\{x, y\} = \{v, w\}$ and $\alpha(x) < \alpha(y)$. Then the resulting complete PCD is a normal completeness of D.

A *semicomplete* PCD is a PCD in which $d^*(v)$ is either zero or one for every vertex v. Then a semicomplete digraph is regarded as the result of an incomplete tournament in which each player has not played at most one game. The optimal rankings and the proper ranks of a semicomplete PCD are obtained by the next theorem. We shall prove it in the next section.

THEOREM 3.3. *Let D be a semicomplete* PCD. *Then*

(1) *The set of optimal rankings of D is the disjoint union of the sets of optimal*

rankings of normal completenesses of D, that is,

$$\mathrm{OR}(D) = \sum_{K \in \mathrm{NC}(D)} \mathrm{OR}(K) \qquad (\textit{disjoint union}).$$

(2) *A completeness K of D is normal if and only if K satisfies the following conditions*:
(i) *if* $\sigma_D^+(v) > \sigma_D^+(w)$ *for* $\{v, w\} \in U(D)$, *then* $vw \in A(K)$, *and*
(ii) *if* $\sigma_D^+(v) = \sigma_D^+(w)$ *for* $\{v, w\} \in U(D)$, *then* $A(K)$ *contains exactly one of arcs* vw *and* wv.

In particular, it follows that $|\mathrm{NC}(D)| = 2^r$ *where* $r = \#\{\{v, w\} \in U(D) \mid \sigma_D^+(v) = \sigma_D^+(w)\}$.

(3) *For each vertex v of D,*

$$\pi_D(v) = \frac{1}{|\mathrm{NC}(D)|} \sum_{K \in \mathrm{NC}(D)} \pi_K(v).$$

For example, let D be a semicomplete PCD as in Fig. 5. Then the normal completenesses of D are K_1 and K_2 shown in Fig. 6. Hence $\mathrm{OR}(D) = \mathrm{OR}(K_1) + \mathrm{OR}(K_2) = \{\alpha = [x_1, x_2, x_3, c, d] \mid \{x_1, x_2, x_3\} = \{a, b, e\}\} + \{\beta = [y_1, y_2, y_3, a, d] \mid \{y_1, y_2, y_3\} = \{b, c, e\}\}$.

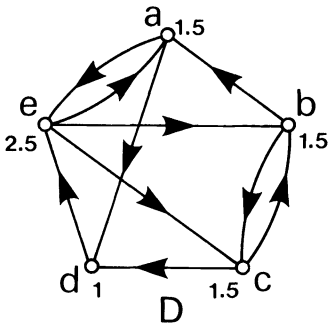


FIG. 5. *A semicomplete* PCD *D.*

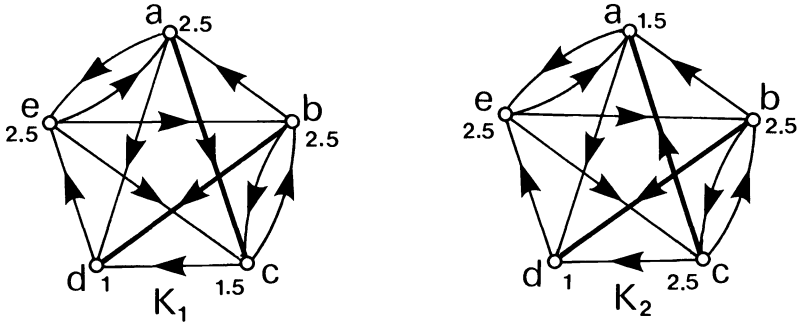


FIG. 6. *The normal completenesses of D.*

4. Optimal rankings. In this section we shall investigate optimal rankings of a PCD and prove Theorem 3.3.

LEMMA 4.1. *Let D be a* PCD *and* $\{v, w\} \in U(D)$. *Then the following statements* (1), (2), *and* (3) *are equivalent*:

(1) $l(D_{vw}) < l(D_{wv})$.
(2) $\alpha(v) < \alpha(w)$ *for all* $\alpha \in \mathrm{OR}(D)$.
(3) $\mathrm{OR}(D) = \mathrm{OR}(D_{vw})$ *and* $l(D) = l(D_{vw})$.

Moreover, the next statements (4), (5), *and* (6) *are also equivalent*:

(4) $l(D_{vw}) = l(D_{wv})$.

(5) *There exist two optimal rankings* α *and* β *of* D *such that* $\alpha(v) < \alpha(w)$ *and* $\beta(w) < \beta(v)$.

(6) $\mathrm{OR}(D) = \mathrm{OR}(D_{vw}) + \mathrm{OR}(D_{wv})$ *and* $l(D) = l(D_{vw}) = l(D_{wv})$.

Proof. (1) implies (2): Suppose $l(D_{vw}) < l(D_{wv})$. Assume that there exists $\alpha \in \mathrm{OR}(D)$ such that $\alpha(w) < \alpha(v)$. Then we have

$$l(D) = \|B_D(\alpha)\| = \|B_{D_{wv}}(\alpha)\| \geqq l(D_{wv}) > l(D_{vw}).$$

Since D is a subdigraph of D_{vw}, it is clear that $l(D) \leqq l(D_{vw})$. This contradicts the above inequality. Thus $\alpha(v) < \alpha(w)$ for all $\alpha \in \mathrm{OR}(D)$.

(2) implies (3): Let $\alpha \in \mathrm{OR}(D)$. Since $\alpha(v) < \alpha(w)$, we have

$$l(D_{vw}) \geqq l(D) = \|B_D(\alpha)\| = \|B_{D_{vw}}(\alpha)\| \geqq l(D_{vw}),$$

which implies that $l(D) = l(D_{vw})$ and $\alpha \in \mathrm{OR}(D_{vw})$, in particular, $\mathrm{OR}(D) \subseteq \mathrm{OR}(D_{vw})$. Conversely, we suppose $\beta \in \mathrm{OR}(D_{vw})$. Then

$$l(D) = l(D_{vw}) = \|B_{D_{vw}}(\beta)\| \geqq \|B_D(\beta)\| \geqq l(D).$$

Therefore $\beta \in \mathrm{OR}(D)$, and thus $\mathrm{OR}(D) \supseteq \mathrm{OR}(D_{vw})$.

We prove later that (3) implies (1).

(4) implies (5): Let $l(D_{vw}) = l(D_{wv})$. Without loss of generality, we may assume that there is $\alpha' \in \mathrm{OR}(D)$ such that $\alpha'(v) < \alpha'(w)$. We shall derive a contradiction assuming that $\alpha(v) < \alpha(w)$ for all $\alpha \in \mathrm{OR}(D)$. Then we may assume $l(D) = l(D_{vw}) = l(D_{wv})$ since (2) implies (3). Let $\gamma \in \mathrm{OR}(D_{wv})$. If $\gamma(v) < \gamma(w)$, then we have

$$l(D) = l(D_{wv}) = \|B_{D_{wv}}(\gamma)\| = \|B_D(\gamma)\| + \gamma(w) - \gamma(v) > \|B_D(\gamma)\| \geqq l(D).$$

This is a contradiction, and thus $\gamma(w) < \gamma(v)$. Hence

$$l(D) = l(D_{wv}) = \|B_{D_{wv}}(\gamma)\| = \|B_D(\gamma)\|,$$

which claims $\gamma \in \mathrm{OR}(D)$. This contradicts the assumption that $\alpha(v) < \alpha(w)$ for all $\alpha \in \mathrm{OR}(D)$.

(5) implies (6): Let $\mathrm{OR}(D)$ be partitioned into two subsets

$$\mathrm{OR}_1 = \{\alpha \in \mathrm{OR}(D) \,|\, \alpha(v) < \alpha(w)\} \quad \text{and} \quad \mathrm{OR}_2 = \{\alpha \in \mathrm{OR}(D) \,|\, \alpha(w) < \alpha(v)\}.$$

We shall prove that if there exists $\alpha \in \mathrm{OR}(D)$ satisfying $\alpha(v) < \alpha(w)$, then $\mathrm{OR}(D_{vw}) = \mathrm{OR}_1$ and $l(D) = l(D_{vw})$. Let $\alpha \in \mathrm{OR}_1$. Then we have

$$l(D_{vw}) \geqq l(D) = \|B_D(\alpha)\| = \|B_{D_{vw}}(\alpha)\| \geqq l(D_{vw}),$$

which implies that $l(D) = l(D_{vw})$ and $\alpha \in \mathrm{OR}(D_{vw})$. In particular $\mathrm{OR}(D_{vw}) \supseteq \mathrm{OR}_1$. On the other hand, let $\beta \in \mathrm{OR}(D_{vw})$. Then

$$l(D) = l(D_{vw}) = \|B_{D_{vw}}(\beta)\| \geqq \|B_D(\beta)\| \geqq l(D),$$

which means that $\beta \in \mathrm{OR}(D)$ and $\beta(v) < \beta(w)$. Therefore we have $\beta \in \mathrm{OR}_1$ and thus $\mathrm{OR}(D_{vw}) \subseteq \mathrm{OR}_1$. This completes the proof of $\mathrm{OR}(D_{vw}) = \mathrm{OR}_1$. Therefore, (6) holds.

(6) implies (4): This is immediate.

(3) implies (1): Let us assume $l(D_{vw}) \geqq l(D_{wv})$. Then we have $l(D_{vw}) = l(D_{wv}) = l(D)$ since $l(D_{wv}) \geqq l(D) = l(D_{vw})$. Hence (4) holds and thus (6) follows, which is contrary to (3). Therefore (3) implies (1).

LEMMA 4.2. *Let D be a* PCD *and $\{v, w\} \in U(D)$. Then we have the following*:
(1) $l(D) = \min\{l(D_{vw}), l(D_{wv})\}$.

$$\text{(2)} \qquad \mathrm{OR}(D) = \begin{cases} \mathrm{OR}(D_{vw}) & \text{if } l(D_{vw}) < l(D_{wv}), \\ \mathrm{OR}(D_{wv}) & \text{if } l(D_{vw}) > l(D_{wv}), \\ \mathrm{OR}(D_{vw}) + \mathrm{OR}(D_{wv}) & \text{if } l(D_{vw}) = l(D_{wv}). \end{cases}$$

(3) *If $l(D) = l(D_{vw})$, then* $\mathrm{OR}(D) \supseteq \mathrm{OR}(D_{vw})$ *and $\alpha(v) < \alpha(w)$ for every $\alpha \in \mathrm{OR}(D_{vw})$.*

Proof. These statements are easy consequences of Lemma 4.1.

Note that there exist D and D_{vw} such that $l(D) < l(D_{vw})$ and $\mathrm{OR}(D) \supseteq \mathrm{OR}(D_{vw})$. Therefore, $l(D) < l(D_{vw})$ implies neither $\mathrm{OR}(D) \not\supseteq \mathrm{OR}(D_{vw})$ nor $\mathrm{OR}(D) \cap \mathrm{OR}(D_{vw}) = \varnothing$. Furthermore, $\mathrm{OR}(D) \supseteq \mathrm{OR}(D_{vw})$ does not imply $l(D) = l(D_{vw})$. For example, let T be the tournament in Fig. 4 and D be the semicomplete PCD which is obtained from T by deleting the arc *ad*. Then it follows from Theorems 3.2 and 3.3 that $l(D) = l(T) = 7 < l(D_{da}) = 10$ and $\mathrm{OR}(D) \supset \mathrm{OR}(D_{da})$.

Remember that $\mathrm{NC}(D)$ denotes the set of normal completenesses K of D, which satisfy $l(K) = l(D)$.

THEOREM 4.3. *Let D be a* PCD. *If* $\mathrm{NC}(D) = \{K_1, K_2, \cdots, K_r\}$, *then* $\mathrm{OR}(D) = \mathrm{OR}(K_1) + \mathrm{OR}(K_2) + \cdots + \mathrm{OR}(K_r)$.

Proof. Let $\{a, b\}$ and $\{c, d\}$ be uncompared pairs of D. Then it follows from Lemma 4.2 that

$$\mathrm{OR}(D) = \sum \mathrm{OR}(D_{vw}) = \sum \mathrm{OR}(D_{vw,xy}) \quad \text{(disjoint union)},$$

where the first summation is over D_{vw} such that $\{v, w\} = \{a, b\}$ and $l(D_{vw}) = l(D)$, and the second summation is over $D_{vw,xy}$ such that $\{v, w\} = \{a, b\}$, $\{x, y\} = \{c, d\}$, and $l(D_{vw,xy}) = l(D)$. By repeating this procedure, we obtain the theorem.

We now give a result which shows a virtue of our method. For example, statement (1) of the following theorem tells us that v is stronger than w if the score of v is greater than that of w even though w wins all unplayed $d^*(w)$ games, that is, if $\sigma^+(v) > \sigma^+(w) + d^*(w)$.

THEOREM 4.4. *Let D be a* PCD *and v and w be two vertices of D. Then*:

(1) *If $\mu(vw) = 1$ and $\sigma^+(v) > \sigma^+(w) + d^*(w)$, then $\alpha(v) < \alpha(w)$ for every $\alpha \in \mathrm{OR}(D)$.*

(2) *If $\mu(vw) = 0$ and $\sigma^+(v) > \sigma^+(w) + d^*(w) - 1$, then $\alpha(v) < \alpha(w)$ for every $\alpha \in \mathrm{OR}(D)$. In particular,* $\mathrm{OR}(D) = \mathrm{OR}(D_{vw})$.

(3) *If $\sigma^+(v) > \sigma^+(w)$ and $\{x \in V \mid \mu(vx) = 1\} = \{y \in V \mid \mu(wy) = 1\}$, then $\alpha(v) < \alpha(w)$ for every $\alpha \in \mathrm{OR}(D)$.*

Proof. We first prove (1). By Theorem 3.2, we may assume D is not complete. By Theorem 4.3, every optimal ranking α of D is one of a certain normal completeness K of D. Since $\sigma^+_K(w) \leqq \sigma^+_D(w) + d^*_D(w) < \sigma^+_D(v) \leqq \sigma^+_K(v)$, it follows from Theorem 3.2 that $\alpha(v) < \alpha(w)$. Hence (1) follows.

We next prove (2). It follows from Lemma 4.2 that every optimal ranking α of D is one of a certain PCD H which is obtained from D by adding exactly one of the arcs xy and yx for each $\{x, y\} \in U(D) \setminus \{\{v, w\}\}$, and satisfies $l(H) = l(D)$. Suppose $\alpha(w) = k < \alpha(v) = k + m$. Since $\mu_H(vx) = \mu_H(wx) = 1$ for every vertex x $(\neq v, w)$, we have by Lemma 2.3 that $\|B_H(\alpha^k_m)\| - \|B_H(\alpha)\| = m\{\sigma^+_H(w) - \sigma^+_H(v)\} < 0$ as $\sigma^+_H(w) \leqq \sigma^+_D(w) + d^*_D(w) < \sigma^+_D(v) \leqq \sigma^+_H(v)$. This is contrary to $\alpha \in \mathrm{OR}(H)$. Consequently, $\alpha(v) < \alpha(w)$.

Statement (3) can be proved similarly by Lemma 2.3.

In order to prove Theorem 3.3 we require the following lemma.

LEMMA 4.5. *Let v and w be vertices of a* PCD D. *Then*:

(1) *If* $\sigma^+(v) > \sigma^+(w)$, $d^*(v) = d^*(w) = 1$, *and* $\mu(vw) = 0$, *then* $\alpha(v) < \alpha(w)$ *for all* $\alpha \in \mathrm{OR}(D)$. *In particular,* $\mathrm{OR}(D) = \mathrm{OR}(D_{vw})$ *and* $l(D) = l(D_{vw})$.

(2) *If* $\sigma^+(v) = \sigma^+(w)$, $d^*(v) = d^*(w) = 1$, $\mu(vw) = 0$, *and* $\alpha(v) = k < \alpha(w) = k + m$ *for* $\alpha \in \mathrm{OR}(D)$, *then* $\alpha_m^k \in \mathrm{OR}(D)$. *In particular,* $\mathrm{OR}(D) = \mathrm{OR}(D_{vw}) + \mathrm{OR}(D_{wv})$ *and* $l(D) = l(D_{vw}) = l(D_{wv})$.

Proof. Statement (1) is a corollary of (3) in Theorem 4.4. By Lemma 2.3 we have $\|B(\alpha)\| - \|B(\alpha_m^k)\| = 0$, and thus (2) follows from Lemma 4.1.

Proof of Theorem 3.3. Statement (1) follows at once from Theorem 4.3. Statement (2) is an easy consequence of Lemma 4.5. We now prove (3). Let K and K' be any two normal completenesses of D. If $\{v, w\} \in U(D)$, then we have $\{\sigma_K^+(v), \sigma_K^+(w)\} = \{\sigma_{K'}^+(v), \sigma_{K'}^+(w)\}$ by (2). Hence $\#\{v \in V(K) \mid \sigma_K^+(v) = t\} = \#\{v \in V(K') \mid \sigma_{K'}^+(v) = t\}$ for every positive real number t, and thus it follows from Theorem 3.2 that $|\mathrm{OR}(K)| = |\mathrm{OR}(K')|$. By Theorem 4.3 we have

$$\begin{aligned}
\pi_D(v) &= \frac{1}{|\mathrm{OR}(D)|} \sum_{\alpha \in \mathrm{OR}(D)} \alpha(v) \\
&= \frac{1}{|\mathrm{NC}(D)| \cdot |\mathrm{OR}(K)|} \sum_{K \in \mathrm{NC}(D)} \sum_{\alpha \in \mathrm{OR}(K)} \alpha(v) \\
&= \frac{1}{|\mathrm{NC}(D)|} \sum_K \left\{ \frac{1}{|\mathrm{OR}(K)|} \sum_\alpha \alpha(v) \right\} = \frac{1}{|\mathrm{NC}(D)|} \sum_K \pi_K(v).
\end{aligned}$$

5. 2-semicomplete asymmetric digraphs. A 2-*semicomplete* PCD is a PCD in which $d^*(v)$ is less than or equal to two for every vertex v. Then a 2-semicomplete asymmetric digraph may represent the result of an incomplete tournament in which no game ended in a draw and each player has not played at most two games. We show some theorems by which we can obtain the optimal rankings of a 2-semicomplete asymmetric digraph. We omit, however, their proofs because they are rather complicated and long. Slightly generalized results of these theorems together with their complete proofs will be found in [5].

For a PCD D, we write $(V(D), U(D))$ for the graph whose vertex set is $V(D)$ and whose edge set is the set $U(D)$ of uncompared pairs of D. It is clear that if D is a 2-semicomplete PCD, then each component of $(V(D), U(D))$ with more than one vertex is either a path or a cycle. A sequence $[v_1, v_2, \cdots, v_r]$ of vertices of a PCD D is called an *uncompared path* of D if $[v_1, v_2, \cdots, v_r]$ is the sequence of vertices of a path from v_1 to v_r in $(V(D), U(D))$. Similarly, an *uncompared cycle* $[v_1, v_2, \cdots, v_r]$ with r vertices of D can be defined.

Let D be a 2-semicomplete asymmetric digraph and $[v, w]$ be an uncompared path of D such that $d^*(v) = 1$ and $d^*(w) = 1$ or 2. Then we have by Theorem 4.4 that if $\sigma^+(v) > \sigma^+(w) + 1$, then $\mathrm{OR}(D) = \mathrm{OR}(D_{vw})$; and if $\sigma^+(v) < \sigma^+(w)$, then $\mathrm{OR}(D) = \mathrm{OR}(D_{wv})$. Hence we may restrict ourselves to the case that $\sigma^+(v) = \sigma^+(w)$ or $\sigma^+(v) = \sigma^+(w) + 1$.

THEOREM 5.1. *Let D be a 2-semicomplete asymmetric digraph and $[v, w = w_1, w_2, \cdots, w_r, u]$ $(r \geqq 1)$ be an uncompared path with $d^*(v) = 1$. Suppose $\sigma^+(w) = \sigma^+(w_2) = \cdots = \sigma^+(w_r) = t$ for some integer t. Then the following statements hold*:

(1) *If* $\sigma^+(v) = t + 1$ *and* $\sigma^+(u) \geqq t + 1$, *then*

$$\mathrm{OR}(D) = \begin{cases} \mathrm{OR}(D_{vw}) + \mathrm{OR}(D_{wv}) & \text{if } r \text{ is even,} \\ \mathrm{OR}(D_{vw}) & \text{otherwise.} \end{cases}$$

(2) *If* $\sigma^+(v)=t+1$ *and* $\sigma^+(u)\leqq t-1$ *or if* $\sigma^+(v)=t+1$, $\sigma^+(u)=t$ *and* $d^*(u)=1$, *then*

$$\mathrm{OR}(D)=\begin{cases}\mathrm{OR}(D_{vw}) & \text{if } r \text{ is even,}\\ \mathrm{OR}(D_{vw})+\mathrm{OR}(D_{wv}) & \text{otherwise.}\end{cases}$$

(3) *If* $\sigma^+(v)=t$ *and* $\sigma^+(u)\geqq t+1$, *then*

$$\mathrm{OR}(D)=\begin{cases}\mathrm{OR}(D_{wv}) & \text{if } r \text{ is even,}\\ \mathrm{OR}(D_{wv})+\mathrm{OR}(D_{vw}) & \text{otherwise.}\end{cases}$$

(4) *If* $\sigma^+(v)=t$ *and* $\sigma^+(u)\leqq t-1$ *or if* $\sigma^+(v)=\sigma^+(u)=t$ *and* $d^*(u)=1$, *then*

$$\mathrm{OR}(D)=\begin{cases}\mathrm{OR}(D_{wv})+\mathrm{OR}(D_{vw}) & \text{if } r \text{ is even,}\\ \mathrm{OR}(D_{wv}) & \text{otherwise.}\end{cases}$$

THEOREM 5.2. *Let* D *be a 2-semicomplete asymmetric digraph and* $[v, w, u]$ *be an uncompared path with* $d^*(v)=d^*(u)=2$ *or an uncompared cycle of* D. *Then*

(1) *if* $\sigma^+(v)+1\leqq\sigma^+(w)\geqq\sigma^+(u)+1$, *then* $\mathrm{OR}(D)=\mathrm{OR}(D_{wv})=\mathrm{OR}(D_{wv,wu})$ *and*

(2) *if* $\sigma^+(v)\geqq\sigma^+(w)+1\leqq\sigma^+(u)$, *then* $\mathrm{OR}(D)=\mathrm{OR}(D_{vw})=\mathrm{OR}(D_{vw,uw})$.

THEOREM 5.3. *Let* D *be a 2-semicomplete asymmetric digraph and* $[v=v_1, w=v_2, v_3, \cdots, v_r]$ $(r\geqq 3)$ *be an uncompared cycle of* D. *If every vertex* v_i *has the same score, then* $\mathrm{OR}(D)=\mathrm{OR}(D_{vw})+\mathrm{OR}(D_{wv})$.

Let D be a 2-semicomplete asymmetric digraph and C be any uncompared cycle of D. Then, we may assume that the length of C is greater than three by Theorems 5.2 and 5.3, and so we can take an uncompared path $[x, v_1, v_2, \cdots, v_r, y]$ $(r\geqq 2)$ such that $\sigma^+(x)<\sigma^+(v_1)=\sigma^+(v_2)=\cdots=\sigma^+(v_r)>\sigma^+(y)$ by the above two theorems. In this case, we can use the following theorem or Theorem 4.4.

THEOREM 5.4. *Let* D *be a 2-semicomplete asymmetric digraph and* $[x, v=v_1, w=v_2, v_3, \cdots, v_r, y]$ $(r\geqq 2)$ *be an uncompared path of* D. *Suppose* $\sigma^+(x)=t-1$, $\sigma^+(v_1)=\sigma^+(v_2)=\cdots=\sigma^+(v_r)=t$ *and* $\sigma^+(y)=t-1$ *for some integer* t. *Then*

$$\mathrm{OR}(D)=\begin{cases}\mathrm{OR}(D_{vw})+\mathrm{OR}(D_{wv}) & \text{if } r \text{ is even,}\\ \mathrm{OR}(D_{vw}) & \text{otherwise.}\end{cases}$$

6. Ranking-equal PCD and NP-hardness. A PCD is said to be *balanced* if $\sigma^+(v)=\sigma^-(v)$ for every vertex v, and is said to be *ranking equal* if the proper ranks of all vertices are the same, that is, $\pi(v)=(|V|+1)/2$ for every vertex v. For a ranking α of a PCD with n vertices, we define the *reversed* ranking of α to be a ranking $\bar{\alpha}$ such that $\bar{\alpha}(v)=n+1-\alpha(v)$ for each vertex v.

THEOREM 6.1. *Let* D *be a* PCD *with at least one arc. Then every ranking of* D *is optimal if and only if* D *is balanced and complete.*

Proof. If D is a balanced complete PCD, then every ranking of D is optimal by Theorem 3.2. Conversely, we suppose that every ranking of D is optimal. Put $n=|V|$. For every two distinct vertices v and w, let us consider a ranking α such that $\alpha(v)=1$ and $\alpha(w)=2$. By Lemma 2.3 we have $\|B(\alpha_1^1)\|-\|B(\alpha)\|=\sigma^-(w)-\sigma^-(v)$. Since α and α_1^1 are both optimal, we have $\sigma^-(v)=\sigma^-(w)$. Similarly, considering a ranking β such that $\beta(v)=n-1$ and $\beta(w)=n$, we obtain $\sigma^+(w)=\sigma^+(v)$. Hence there exist two constants δ and δ' such that $\sigma^+(v)=\delta$ and $\sigma^-(v)=\delta'$ for every vertex v. Since $\sum\sigma^+(v)=\sum\sigma^-(v)$, we have $\delta=\delta'$ and conclude that D is balanced. We next prove D is complete. We may clearly assume $n\geqq 3$. For every three distinct vertices u, v, and w, let us consider a ranking α such that $\alpha(u)=1$, $\alpha(v)=2$, and $\alpha(w)=3$. By Lemma

2.3, $\|B(\alpha_1^2)\| - \|B(\alpha)\| = \sigma^-(w) - \sigma^-(v) + \mu(vu) - \mu(wu)$. Since $\|B(\alpha_1^2)\| = \|B(\alpha)\|$ and $\sigma^-(w) = \sigma^-(v)$, we have $\mu(vu) = \mu(wu)$. Therefore $\mu(xy)$ is constant (i.e., $\mu(xy) = 1$ or 0) for every two vertices x and y. Since D has at least one arc, we conclude that D is complete.

LEMMA 6.2. *Let α be a ranking of a* PCD $D = (V, A)$. *Then*

$$\|B(\alpha)\| - \|B(\bar{\alpha})\| = \sum_{vw \in A} \varepsilon(vw)\{\alpha(v) - \alpha(w)\} = \sum_{v \in V} \alpha(v)\{\sigma^+(v) - \sigma^-(v)\}.$$

Proof. Since $B(\bar{\alpha}) = A \backslash B(\alpha)$ and $\bar{\alpha}(v) - \bar{\alpha}(w) = \alpha(w) - \alpha(v)$, we have

$$\begin{aligned}
\|B(\alpha)\| - \|B(\bar{\alpha})\| &= \sum_{vw \in B(\alpha)} \varepsilon(vw)\{\alpha(v) - \alpha(w)\} - \sum_{vw \in B(\bar{\alpha})} \varepsilon(vw)\{\bar{\alpha}(v) - \bar{\alpha}(w)\} \\
&= \sum_{vw \in A} \varepsilon(vw)\{\alpha(v) - \alpha(w)\} = \sum_{v,w \in V} \bar{\varepsilon}(vw)\{\alpha(v) - \alpha(w)\} \\
&= \sum_{v \in V} \left\{ \sum_{w \in V} \bar{\varepsilon}(vw)\alpha(v) \right\} - \sum_{w \in V} \left\{ \sum_{v \in V} \bar{\varepsilon}(vw)\alpha(w) \right\} \\
&= \sum_{v} \sigma^+(v)\alpha(v) - \sum_{w} \sigma^-(w)\alpha(w) = \sum_{v} \alpha(v)\{\sigma^+(v) - \sigma^-(v)\}.
\end{aligned}$$

THEOREM 6.3. *A* PCD D *is ranking equal if and only if the reversed ranking of every optimal ranking of D is also optimal.*

Proof. We first suppose $D = (V, A)$ is ranking equal. Since $\pi(v) = \pi(w)$ for every two vertices v and w, it follows that

$$\sum_{\alpha \in \mathrm{OR}(D)} \alpha(v) = \sum_{\alpha \in \mathrm{OR}(D)} \alpha(w).$$

Hence we have the following equation by Lemma 6.2

$$\begin{aligned}
0 &= \sum_{vw \in A} \varepsilon(vw) \left[\sum_{\alpha \in \mathrm{OR}(D)} \{\alpha(v) - \alpha(w)\} \right] \\
&= \sum_{\alpha} \left[\sum_{vw} \varepsilon(vw)\{\alpha(v) - \alpha(w)\} \right] = \sum_{\alpha} \{\|B(\alpha)\| - \|B(\bar{\alpha})\|\}.
\end{aligned}$$

Since $\|B(\alpha)\| \leqq \|B(\bar{\alpha})\|$ for every $\alpha \in \mathrm{OR}(D)$, we obtain $\|B(\alpha)\| = \|B(\bar{\alpha})\|$. Hence $\bar{\alpha}$ is also optimal.

We next suppose that $\bar{\alpha}$ is optimal for every $\alpha \in \mathrm{OR}(D)$. Then we have the following equation for any vertex v of D.

$$\pi(v) = \frac{1}{|\mathrm{OR}(D)|} \sum_{\alpha \in \mathrm{OR}(D)} \alpha(v) = \frac{1}{|\mathrm{OR}(D)|} \sum_{\alpha \in \mathrm{OR}(D)} \bar{\alpha}(v).$$

Hence we have

$$2\pi(v) = \frac{1}{|\mathrm{OR}(D)|} \sum_{\alpha} \{\alpha(v) + \bar{\alpha}(v)\} = \frac{1}{|\mathrm{OR}(D)|} \sum_{\alpha} (n+1) = n+1,$$

where n is the number of vertices of D. Therefore $\pi(v) = (n+1)/2$ and thus D is ranking equal.

COROLLARY 6.4. *If D is a balanced* PCD, *then D is ranking equal.*

Proof. By Lemma 6.2, $\bar{\alpha} \in \mathrm{OR}(D)$ for all $\alpha \in \mathrm{OR}(D)$. Hence the corollary follows from the above theorem.

The converse statement of this corollary is not true. A counterexample of the converse statement is obtained easily in the following way: Let D be a balanced PCD

and vw be an arc of D. We often obtain a nonbalanced ranking-equal PCD D' form D by changing the weights of vw (and wv if it exists) as follows; $\varepsilon_{D'}(vw) = \bar{\varepsilon}_D(vw) + \delta$ and $\varepsilon_{D'}(wv) = \bar{\varepsilon}_D(wv) - \delta$ where δ is a sufficiently small positive real number. Furthermore, there exists a counterexample even if we restrict a PCD to a digraph. It is shown in Fig. 7. It is verified to be ranking equal by using a computer. We have, however, no counterexample in the case of an asymmetric digraph.

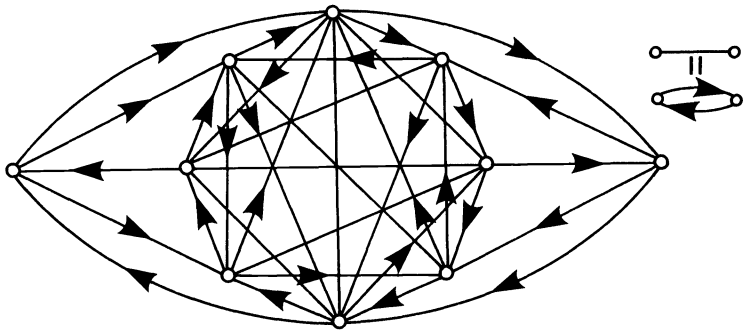


FIG. 7. *An example of a nonbalanced ranking-equal digraph.*

THEOREM 6.5. *A problem of finding the optimal backward length for a* PCD *is* NP-*hard.*

Proof. The following problem, called "simple optimal linear arrangement", is NP-complete [3].

Input: Graph $G = (V(G), E(G))$ and positive integer k.

Property: There is a ranking α such that

$$\sum_{\{v,w\} \in E(G)} |\alpha(v) - \alpha(w)| \leqq k.$$

For a graph G given as the input for "simple optimal linear arrangement," let $D = (V(D), A(D))$ where $V(D) = V(G)$, $A(D) = \{vw, wv \mid \{v, w\} \in E(G)\}$, and the weight of each arc is 0.5. Then we have

$$\sum_{\{v,w\} \in E(G)} |\alpha(v) - \alpha(w)| = 2 \sum_{vw \in B_D(\alpha)} 0.5\{\alpha(v) - \alpha(w)\} = 2\|B_D(\alpha)\|.$$

Therefore, there exists a ranking α which satisfies the property of "simple optimal linear arrangement" with integer k if and only if there exists a ranking α such that $\|B_D(\alpha)\| \leqq k/2$. That is, "simple optimal linear arrangement" is polynomial reducible to the problem of finding the optimal backward length. Hence the theorem is proved.

7. Concluding remarks. We have proposed a new method of ranking the vertices of a paired comparison digraph. This method can be applied not only to a tournament but also to every digraph. Some good properties of the method are given in Remark 2.1, Theorems 3.2 and 4.4, and Corollary 6.4. The basic idea of proof technique is indicated in Lemmas 2.3 and 4.2. However, a problem of finding the optimal backward length of a PCD is NP-hard as seen in Theorem 6.5.

It seems to be dangerous to decide the ranks of participants taken in a round-robin tournament if a lot of games are unplayed because of accidents or the like. We can fortunately obtain the optimal rankings of a PCD D by Lemma 4.2 and Theorems 3.2 and 4.3 if the number of uncompared pairs of D, which corresponds to the number of unplayed games, is small. The optimal rankings of the example given in Fig. 7 are obtained by a computer in this way. We have shown in Theorem 3.3 that if $d^*(v)$,

which corresponds to the number of games which have not been played by v, is less than or equal to one for every vertex v of a PCD D, then it is easy to obtain the optimal rankings of D. Furthermore, if D is an asymmetric digraph in which $d^*(v)$ is less than or equal to two, then we can get the optimal rankings of D by theorems in § 5.

REFERENCES

[1] C. BERGE, *Graphs and Hypergraphs*, North-Holland, Amsterdam, 1973.

[2] J. A. BONDY AND U. S. R. MURTY, *Graph Theory with Applications*, Macmillan, London, 1976.

[3] M. R. GAREY, D. S. JOHNSON AND L. STOCKMEYER, *Some simplified polynomial complete problems*, Proc. 6th Annual ACM Symposium on Theory of Computing, 1974, pp. 47–63.

[4] M. KANO AND A. SAKAMOTO, *Ranking the vertices of a weighted digraph using the length of forward arcs*, Networks, 13 (1983), pp. 143–151.

[5] ———, *Ranking the vertices of a paired comparison digraph with normal completeness theorems*, Bulletin of Faculty of Engineering, Tokushima University, 20 (1983), pp. 119–128.

[6] M. G. KENDALL, *Further contributions to the theory of paired comparison digraphs*, Biometrics, 11 (1955), pp. 43–62.

[7] J. W. MOON AND N. J. PULLMAN, *On generalized tournament matrices*, SIAM Rev., 12 (1970), pp. 384–399.

[8] A. RUBINSTEIN, *Ranking the participants in a tournament*, SIAM J. Appl. Math., 38 (1980), pp. 108–111.

[9] T. H. WEI, *The algebraic foundation of ranking theory*, Ph.D. thesis, Cambridge Univ., Cambridge, 1952.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 1, January 1985

010

BISECTION OF CIRCLE COLORINGS*

CHARLES H. GOLDBERG† AND DOUGLAS B. WEST‡

Abstract. Consider $2n$ beads of k colors arranged on a necklace, using $2a_i$ beads of color i. A *bisection* is a set of disjoint strings ("intervals") of beads whose union captures half the beads of each color. We prove that any arrangement with k colors has a bisection using at most $\lceil k/2 \rceil$ intervals. In addition, if k is odd, an endpoint of one interval can be specified arbitrarily. The result is best possible. For fixed k, there is a polynomial-time algorithm to find such a bisection; it runs in $O(n^{k-2})$ for $k \geq 3$. We consider continuous and linear versions of the problem and use them to obtain applications in geometry, VLSI circuit design, and orthogonal functions.

AMS(MOS) classifications. 05A, 68C, 55

1. Introduction. Consider necklaces formed using two colors of beads. If an even number of each color is used, it is easy to show that a single interval can be chosen that will capture exactly half the beads of each color. Start with any interval capturing half the total number of beads. If it is short in color A and has too much of color B, then the complementary interval is imbalanced the other way. Sliding from an interval to its complement one bead at a time changes the imbalance by at most one bead at a time, so there must be some intermediate stage where the colors are in balance, i.e. where the interval contains half of each color.

A natural generalization of this problem arose in the study of VLSI circuit design. Bhatt and Leiserson [1] asked the corresponding question for 3-color necklaces, i.e. whether it is always possible to chose two intervals that together capture half the beads of each color. In this paper we prove that this is true, and in fact we obtain the best possible result for the general case of k colors. In order to do this, we prove a stronger result about continuously integrable functions on the circle. The continuous result has several geometric applications, and the discrete result has an application to "graph separators."

First we state the discrete result. Suppose $2n$ beads of k colors are placed around a circle, using $2a_i$ beads of color i; such a configuration is called a *necklace* or *discrete coloring*. A *bisection* of a coloring is a set of nonoverlapping intervals on the circle whose union contains exactly half of each color. The *size* of the bisection is the number of intervals used. We prove the following theorem.

THEOREM 1. *Every necklace with k colors of beads has a bisection using no more than* $\lceil k/2 \rceil$ *intervals. No smaller size suffices for all arrangements of k colors. If k is odd, this size suffices even if one of the intervals is required to end between a specified pair of beads.*

Of course, corresponding to this extremal problem, there is an optimization problem. Given a particular arrangement of beads, what is the bisection of smallest size? The complexity of this problem is open; it may be NP-complete. When the number of colors in the arrangement is bounded by k, the proof of Theorem 1 leads to an efficient algorithm to find a bisection of size at most $\lceil k/2 \rceil$. Its running time is bounded by a polynomial in $2n$, the number of beads in the arrangement. After time

* Received by the editors July 25, 1983, and in final form December 19, 1983. This work was presented at the SIAM Second Conference on the Applications of Discrete Mathematics, held at Massachusetts Institute of Technology, Cambridge, Massachusetts, June 27–29, 1983.

† Mathematics Department, Trenton State College, Trenton, New Jersey 08625.

‡ Mathematics Department, University of Illinois, Urbana, Illinois 61801.

$O(n)$ for preprocessing, the algorithm runs in $O(n^{k-2})$ if $k \geqq 3$. In particular, it is linear if $k = 3$.

The direct proof given above for the 2-color case becomes increasingly ugly as the number of color increases, because no longer does the imbalance in the captured colors change by "at most one" when the parameters describing the choice of intervals change by one. We initially obtained the proof for $k \leqq 4$ by this direct method, but here we will present only a more elegant approach that works equally well for arbitrarily many colors. This proof uses a natural extension to a continuous problem. Paint the perimeter of the circle with various colors. The colors no longer need be restricted to disjoint intervals; instead, they may "mix." We require only that each color have a continuously integrable density function (i.e., the cumulative distribution is continuous), and that the densities of the various colors sum to 1 at each point. (Note that some "densities" may actually be negative.) We call such a painting a *circle coloring.* Bisection is defined as before. The previous result holds again in the continuous case.

THEOREM 2. *Every circle coloring with k colors has a bisection using at most* $\lceil k/2 \rceil$ *intervals. When k is odd, one endpoint of one interval may be chosen arbitrarily. No smaller size suffices for all circle colorings with k colors.*

The proof uses methods from topology. An appropriate parameter space is defined to describe the possible placements of $\lceil k/2 \rceil$ intervals, and the cumulative density functions map this space continuously into a $(k-1)$-hyperplane. The coordinates of the hyperplane sum to $\sum a_i$, and the "target point" $(a_1, \cdots, a_n)$ corresponds to capturing half of each color. Any inverse image of this point is a bisection. To show that this inverse image is nonempty, we want to show that the target point is "inside" the image of the boundary and apply continuity. In topological terms, "inside" is related to winding number; we show that if the boundary of the parameter space does not hit the target point, then it maps to a surface that has odd winding number relative to the target point. This odd parity follows by induction on the number of colors; we construct a problem with fewer colors in which the winding number of the boundary has the same parity. For example, a problem with fewer colors could be obtained by amalgamating the last two colors. That this all works depends heavily on a simple fact used in the 2-color argument mentioned at the beginning. The complement of a set of $\lceil k/2 \rceil$ disjoint intervals on the circle is also a set of $\lceil k/2 \rceil$ disjoint intervals on the circle, and it captures a complementary amount of each color. In particular, the complement of a bisection is also a bisection.

There are several applications of both the discrete and the continuous result. The continuous result can be viewed as a version applicable in projective space of the "intermediate value theorem" of single-variable calculus. In the four-color case, the continuous result implies that for any (directed) closed curve in $\mathbf{R}^3$, there are two equal and opposite chords that together cut off half the length of the curve. In $\mathbf{R}^2$, this can be strengthened to obtain a rectangle, or right angle subtended at any point using two other points on the curve, etc. These arguments lead to a proof of an old geometry conjecture for the special case of "Nice curves" in $\mathbf{R}^2$. The conjecture is that any closed curve in $\mathbf{R}^2$ contains four points that determine a square. The conditions and proof are similar to those of Jerrard [4] (see also [5]). The unproved case is that of nondifferentiable curves, for which these methods are less well suited.

The discrete result applies to "separators," a construct used in VLSI design theory [6]. For present purposes, the following definition suffices. An $f(n)$*-separator* of a graph on n vertices recursively splits its nodes into two parts having sizes $\lfloor n/2 \rfloor$ and $\lceil n/2 \rceil$, with at most $f(n)$ edges between them. Roughly speaking, if the nodes of a graph with an $f(n)$-separator are colored arbitrarily with up to k colors, where

$f(n) > cn$, then the discrete bisection result implies that the graph has an $O(kf(n))$-separator that splits each of the colors in half when it separates the vertices. This uses a restatement of Theorem 1 for the problem of bisecting "opened necklaces." Suppose beads of k colors are arranged on a line segment. Then intervals for a bisection can be obtained by making at most k cuts. In this version it is not necessary to distinguish between odd and even k, since the endpoint of the segment, which corresponds to the opening of the necklace, serves as the arbitrary prescribed cut if k is odd. Unfortunately, no proof for this cleaner formulation is known that does not use the circle coloring result.

2. Discrete from continuous. In this section we show that Theorem 2 implies Theorem 1. We begin by noting that some arrangements using k colors require the full $\lceil k/2 \rceil$ intervals. (This example also provides the lower bound for the continuous problem.) Arrange the beads so that the beads of each color appear contiguously. Since no bisection can include all the beads of a color or exclude them all, there must be at least one "cut" (a switch between inclusion and exclusion) among the beads of each color. With at least k switches between beads chosen and beads omitted, the beads chosen must be separated into at least $\lceil k/2 \rceil$ intervals.

Now, suppose that Theorem 2 holds. We can turn a necklace of $2n$ beads into a circle coloring by partitioning the circle into $2n$ equal segments ("units") and coloring them with k pure colors corresponding to the order of beads on the necklace. Theorem 2 guarantees a bisection with at most $\lceil k/2 \rceil$ intervals, but the endpoints of the intervals (the cuts) need not occur at endpoints of the $2n$ units. Using induction on the number of these bad cuts, this can be corrected to obtain a bisection for the discrete problem. If there are no bad cuts, we are finished. Otherwise, suppose there is a cut inside a unit of color i, which is used on $2a_i$ units around the circle. Since the continuous bisection captures altogether an integral amount a_i of color i, there must be another bad cut in a unit of color i. At any cut within color i, the interval on one side contributes to the amount if color i captured, and the interval on the other side does not. Move the two bad cuts, by the same amount, so as to shrink the interval contributing to color i at one of them and expand the interval contributing to color i at the other, until one of the cuts reaches the endpoint of a unit or another bad cut. This produces a bisection with fewer bad cuts.

The same argument can be used to treat necklaces where the number of beads of a color need not be even. Apply the continuous result and transform the resulting bisection as above. Cuts will occur only at endpoints of units except that a color contributing an odd number of beads will have one unit with a cut at its midpoint. That cut can be moved a half unit in either direction to obtain the following result, where we broaden the definition of discrete bisection to mean a choice of beads capturing $\lceil 2a_i/2 \rceil$ or $\lfloor 2a_i/2 \rfloor$ beads of color i.

THEOREM 3. *Every necklace with k colors of beads has a discrete bisection using no more than* $\lceil k/2 \rceil$ *intervals. No smaller size suffices for all arrangements of k colors. If k is odd, one cut can be specified arbitrarily. Any pattern of* $\lceil 2a_i/2 \rceil$*'s and* $\lfloor 2a_i/2 \rfloor$*'s can be specified for the bead colors used an odd number of times.*

3. The parameter space. In this section we specify the parameters used to describe a choice of intervals and obtain properties of the parameter space that will be important in the proof. For ease of discussion, we refer to any choice of intervals that together capture half the total length as a *snare*; if it consists of at most j disjoint intervals, we call it a *j-snare.* In the discrete case, length is measured by number of beads. In the continuous case, we measure length by integrating the densities, with the total amount

of color i being $2a_i$. In either case, the total length is $2\sum a_i$; let $L=\sum a_i$. A snare *captures* the points contained in its intervals.

Given a coloring, specify an arbitrary point P on the circle as a reference point. In the discrete case, P must lie between two units. The intervals are determined by the cuts made between them, so we could specify the intervals by measuring the distances from P to those cuts. However, it will be more convenient to organize the parameters into simplices.

Henceforth let $m \equiv \lceil k/2 \rceil$, and restrict attention to m-snares. Since the complement of any bisection with m intervals is also a bisection with m intervals, the parameter space need only describe at least one m-snare from every complementary pair. In fact, our parameter space B_k contains points describing all m-snares for which P is not an interior point of a (captured) interval. Given this, the parameters describing such a snare are easily computed. Moving in a counterclockwise direction, let y_0 be the distance from P to the first beginning of an interval in the snare. Continuing counterclockwise, let x_i be the length of the ith interval in the snare, and let y_i be the length of the gap between the ith and $(i+1)$th intervals, with y_m the length between the mth and P. If the snare has less than m intervals, this description still works, by setting leftover parameters to 0. Note that $\sum x_i = \sum y_i = L$.

As noted above, a bisection of size m exists if and only if there is one where P is not an interior point of an interval. Thus, the parameter space in which we look for bisections can be described as $B_k = X \times Y$, where X and Y are simplices whose coordinates sum to L, representing the choices for $\bar{x}$ and $\bar{y}$. However, for odd k we claim that one cut can be made arbitrarily. We choose this to be the reference point P. By setting $y_0 = 0$, we obtain all snares in which the first interval starts at P. The snares in which the last interval ends at P have $y_m = 0$ and possibly $y_0 > 0$; we ignore these since they are complements of those with $y_0 = 0$. Thus, when k is odd we drop y_0, so that X and Y are both m-parameter, $(m-1)$-dimensional simplices. If k is even, X is $(m-1)$-dimensional and Y is m-dimensional. In either case, B_k is $(k-1)$-dimensional; it has $k+1$ parameters, restricted by the two sum relations $\sum x_i = \sum y_i = L$.

Let A_k be a $(k-1)$-dimensional hyperplane of k-tuples whose coordinates sum to L. Let $f: B_k \to A_k$ be the function whose value on a particular snare gives the amount of each color captured. In the discrete case, this counts the captured beads; in the continuous case, it integrates the density functions over the intervals of the snare. Given a coloring with total amount $2a_i$ of color i, the point of interest is $\bar{a} = (a_1, \cdots, a_k)$; we want to insure that $\bar{a}$ is in the image of f. We say that two points $\bar{b}, \bar{b}'$ are *antipodal* in A_k if $\bar{b} = 2\bar{a} - \bar{b}'$. Expressing this as $\bar{b} - \bar{a} = \bar{a} - \bar{b}'$, note that for antipodal points any excess for $\bar{b}$ in a color becomes a deficiency of the same size for $\bar{b}'$ in that color. In particular, points of B_k that describe complementary snares map to antipodal points of A_k under f.

For purposes of the proof, consider now the continuous case; we will return to the discrete case to discuss algorithms. Geometrically, a direct way to show that $\bar{a}$ is hit by f would be to study the image of the boundary of B_k, and attempt to use the Borsuk–Ulam theorem regarding antipodal maps. Intuitively, one wants to show that $f(\partial B_k)$ "surrounds" $\bar{a}$, so that $\bar{a}$ must lie "inside" $f(\partial B_k)$, in which case continuity implies $\bar{a} \in f(B_k)$.

We claim the points of $f(\partial B_k)$ come in antipodal pairs, with each pair the image of points in ∂B_k describing complementary snares. If all the points of ∂B_k could be paired up this way, then the Borsuk–Ulam theorem could be applied to get the desired result. Unfortunately, not all of ∂B_k participates, so this theorem cannot be used directly. This problem arises because $(m-1)$-snares have many descriptions in the

parameter space. Another map could be introduced to "collapse" the poorly-behaved part of the parameter space and then apply the Borsuk–Ulam theorem; this approach works but leads to very tedious and technical arguments, so we omit it. The inductive proof presented in the next section circumvents the difficulty, using arguments like those used to prove the Borsuk–Ulam theorem. Before embarking on that, we pause to obtain properties of the parameter space and the map f that will be useful in the proof.

We need to study the boundary of B_k. The boundary points are those where at least one of the parameters in the simplices X or Y is 0. We will break the boundary into four pieces, a "front face" B^+, a "back face" B^-, and two other parts C and D, depending on which of the parameters reaches an extreme value. In general, if $x_i = 0$, one of the intervals in the corresponding snare vanishes and two of the gaps merge. Similarly, if $y_i = 0$ for $1 \leqq i < m$, one of the gaps vanishes and two of the intervals merge. In either case, the snare actually uses at most $m-1$ intervals. Conversely, describing a snare with less than m intervals requires one of these parameters to be 0; all $(m-1)$-snares are described by points in ∂B_k. For even values of k, the parts C and D will contain the points that describe $(m-1)$-snares; for odd k they will describe $(m-1)$-snares with a restricted endpoint, at P. B^+ and B^- allow more freedom; for even k their points describe m-snares with a restricted endpoint, and for odd k they describe all $(m-1)$-snares.

Since C and D describe snares with less freedom, it comes as no surprise that their images have smaller dimension than the rest of the boundary. In fact, we will show that they do not contribute anything to image of the boundary; i.e. $f(\partial B_k) = f(B^+) \cup f(B^-)$. This justifies the terms "front face" and "back face". In relation to the preceding topological motivation, it is important that we can in this sense throw away C and D, because these are the parts of ∂B_k that do not participate in the pairing of points with antipodal images.

To define B^+, B^-, C, D, we consider two cases. First suppose k is even, so that Y has points $\bar{y} = y_0, \cdots, y_m$. Let $B^+ = \{(\bar{x}; \bar{y}): y_0 = 0\}$, $B^- = \{(\bar{x}; \bar{y}): y_m = 0\}$, $C = \partial X \times Y$, and $D = X \times \partial Y - (B^+ \cup B^-)$. Since B^+ and B^- consist of the points of B_k with $y_0 = 0$ or $y_m = 0$, they describe all m-snares in which the first or last interval (which may be empty if $x_1 = 0$ or $x_m = 0$) begins or ends at P. C and D describe $(m-1)$-snares. Note that $\partial B_k = B^+ \cup B^- \cup C \cup D$.

Next, suppose that k is odd, so that $Y = \{(y_1, \cdots, y_m)\}$. Let $B^+ = \{(\bar{x}; \bar{y}): x_1 = 0\}$, $B^- = \{(\bar{x}; \bar{y}): y_m = 0\}$, $C = (\partial X \times Y) - B^+$, and $D = (X \times \partial Y) - B^-$. This time B^+ and B^- describe $(m-1)$-snares that do not or do capture P. C and D describe $(m-1)$-snares where the first or last interval (which may be empty if $x_1 = 0$ or $x_m = 0$) begins or ends at P. Again, $\partial B_k = B^+ \cup B^- \cup C \cup D$.

LEMMA 1. *Let $f: B_k \to A_k$ be a circle coloring, and assume that $\dim f(S) \leqq \dim S$ for any $S \subset B_k$. Then the pieces B^+, B^-, C, D of the boundary of B_k satisfy the following properties.*

(a) *B^+ is naturally isomorphic to B_{k-1}.*

(b) *The points of B^+ and B^- come in pairs with antipodal images.*

(c) *$f(\partial B_k) = f(B^+) \cup f(B^-)$, and in fact $f(C) \cup f(D) \subset f(B^+) \cap f(B^-)$.*

(d) *$\dim (f(C) \cup f(D)) \leqq \dim A_k - 2$.*

Proof. We consider the even and odd cases separately. Keep in mind that when k is odd $\bar{x}$ and $\bar{y}$ have the same number of components, but when k is even $\bar{y}$ has one more component than $\bar{x}$.

Assume k is even. (a) is trivial, since deleting the fixed value $y_0 = 0$ makes B^+ precisely the parameter space B_{k-1} for a problem with one less color. B^- is a translate

of B_{k-1}. For (b), note that $f(\bar{x}; y_0, \cdots, y_{m-1}, 0) = 2\bar{a} - f(y_0, \cdots, y_{m-1}; 0, \bar{x})$, since these points from B^- and B^+ describe complementary snares.

(c) and (d) rest on the fact that an $(m-1)$-snare has many descriptions in B_k; any one of the gaps or intervals could be the one that collapses. In particular, it has a description in both B^+ and B^-. For the image of a point in C with $x_i = 0$,

$$f(x_1, \cdots, x_{i-1}, 0, x_{i+1}, \cdots, x_m; \bar{y})$$
$$= f(0, x_1, \cdots, x_{i-1}, x_{i+1}, \cdots, x_m; 0, y_0, \cdots, y_{i-2}, y_{i-1} + y_i, y_{i+1}, \cdots, y_m) \in f(B^+)$$
$$= f(x_1, \cdots, x_{i-1}, x_{i+1}, \cdots, x_m, 0; y_0, \cdots, y_{i-2}, y_{i-1} + y_i, y_{i+1}, \cdots, y_m, 0) \in f(B^-),$$

because all three parameter sets describe the same snare. Similarly, for the image of a point in D with $y_i = 0$ for $1 \leqq i < m$,

$$f(\bar{x}; y_0, \cdots, y_{i-1}, 0, y_{i+1}, \cdots, y_m)$$
$$= f(0, x_1, \cdots, x_{i-1}, x_i + x_{i+1}, x_{i+2}, \cdots, x_m; 0, y_0, \cdots, y_{i-1}, y_{i+1}, \cdots, y_m) \in f(B^+)$$
$$= f(x_1, \cdots, x_{i-1}, x_i + x_{i+1}, x_{i+2}, \cdots, x_m, 0; y_0, \cdots, y_{i-1}, y_{i+1}, \cdots, y_m, 0) \in f(B^-).$$

Since we have given descriptions from both $f(B^+)$ and $f(B^-)$, this proves (c). Moreover, the description from $f(B^+)$ for each point of $f(C) \cup f(D)$ shows that the dimension of $f(C) \cup f(D)$ is at most $2m-3 = k-3$, because there are $2m-1$ nonfixed parameters and two sum relations $\sum x_i = L = \sum y_i$, and the hypothesis of the lemma states that f does not raise dimension. Since $\dim A_k = k-1$, we have (d).

Now suppose k is odd. (a) follows from the fact that $(0, x_2, \cdots, x_m; \bar{y}) \in B^+$ and $(x_2, \cdots, x_m; \bar{y}) \in B_{k-1}$ describe the same snare. For (b), $(0, x_2, \cdots, x_m; \bar{y}) \in B^+$ and $(\bar{y}; x_2, \cdots, x_m, 0) \in B^-$ describe complementary snares, whose images under f are antipodal. To show the other claims, we proceed as in the even case. If some $x_i = 0$ with $i > 1$,

$$f(x_1, \cdots, x_{i-1}, 0, x_{i+1}, \cdots, x_m; \bar{y})$$
$$= f(0, x_1, \cdots, x_{i-1}, x_{i+1}, \cdots, x_m; 0, y_1, \cdots, y_{i-2}, y_{i-1} + y_i, y_{i+1}, \cdots, y_m) \in f(B^+)$$
$$= f(x_1, \cdots, x_{i-1}, x_{i+1}, \cdots, x_m, 0; y_1, \cdots, y_{i-2}, y_{i-1} + y_i, y_{i+1}, \cdots, y_m, 0) \in f(B^-).$$

Similarly, if some $y_i = 0$ with $i < m$,

$$f(\bar{x}; y_1, \cdots, y_{i-1}, 0, y_{i+1}, \cdots, y_m)$$
$$= f(0, x_1, \cdots, x_{i-1}, x_i + x_{i+1}, x_{i+2}, \cdots, x_m; 0, y_1, \cdots, y_{i-1}, y_{i+1}, \cdots, y_m) \in f(B^+)$$
$$= f(x_1, \cdots, x_{i-1}, x_i + x_{i+1}, x_{i+2}, \cdots, x_m, 0; y_1, \cdots, y_{i-1}, y_{i+1}, \cdots, y_m, 0) \in f(B^-).$$

This establishes (c). To show (d), note from the $f(B^+)$ description that in $f(B^+) \cap f(B^-)$ there are $2m-2$ nonfixed parameters. Together with the same two sum relations, this yields $\dim f(C) \cup f(D) \leqq 2m-4 = k-3 = \dim A_k - 2$. □

4. $\lceil k/2 \rceil$ intervals suffice. In this section, we use induction on k to show that an arbitrary circle coloring with k colors and continuously integrable color densities has a bisection with at most $\lceil k/2 \rceil$ intervals. We consider only bisections described by points in the parameter space B_k. In particular, none of the m (possibly empty) specified intervals has the reference point P in its interior, and P is the starting point of an interval if k is odd. The proof uses concepts and well-known results from topology. Due to the origin of the problem and the application of the results, we expect many of the readers of this paper to be discrete mathematicians. Therefore, we will try to

describe the relevant topological concepts in footnotes for interested readers; missing definitions or results can be found in [2] or [3].

If the boundary of B_k contains a solution, there is nothing to prove, so assume $\bar{a} \notin f(\partial B_k)$. We would like to draw conclusions about the number of points in $f^-(\bar{a})$, in particular that this number is odd and therefore nonzero. Unfortunately, $|f^-(\bar{a})|$ does not behave well enough for arbitrary continuous functions f (for example, it may become infinite), so instead we study a quantity that agrees with $|f^-(\bar{a})|$ whenever f is well-behaved (a local homeomorphism) at each point of $f^-(\bar{a})$. This quantity is the degree of the map $f:(B_k, \partial B_k) \to (A_k, A_k - \bar{a})$, henceforth denoted $\deg(f)$. The degree of a continuous map from a set and boundary into another set with a point removed describes how many times the image covers the target point. In particular, if $f^-(\bar{a})$ is empty, then $\deg(f) = 0$.[1] Since $\deg(f)$ is a construct using homology, $\deg(f)$ does not change under continuous perturbations ("homotopies") of f, a fact that will be used when f is poorly behaved. Later, we will relate $\deg(f)$ to the winding number of $f(\partial B_k)$ around the target point, since that is the quantity to which we can apply the antipodal facts we derived about ∂B_k.

Since we seek only bisections corresponding to points in B_k, we will refer to such points as "solutions," and we will denote the problem of finding such solutions by the corresponding color function f. We will prove Theorem 2 by showing that $\deg(f)$ is odd (i.e., nonzero) when the problem has no solutions in ∂B_k. Theorem 2 follows readily; if a k-color problem has no solutions in B_k, then it has none in ∂B_k and hence $\deg(f)$ is nonzero. This contradicts the fact that $\deg(f) = 0$ if $f^-(\bar{a})$ is empty. We have reduced the theorem to the following topological lemma.

LEMMA 2. *Let $f: B_k \to A_k$ be a coloring bisection problem. If f has no solutions in ∂B_k, then $\deg(f)$ is odd.*

Proof. We prove this by induction on k. For $k = 1$, the parameter space B_1 consists of the single point $(L; L)$. Since the sum of the color densities must be 1 at each point, one color paints the circle uniformly. The snare corresponding to $(L; L)$ is the semicircle starting at P, so indeed it bisects the coloring. Recall that $\deg(f)$ is the degree of the map $f:(B_1, \partial B_1) \to (A_1, A_1 - \bar{a})$; in this case the second element of each pair is empty, and the first is a 1-point set. Any mapping of an n-point set to a 1-point set covers the target point n times and has degree n, so here $\deg(f) = 1$.

For the induction step, we must show the following: given a k-color problem f with no solutions in ∂B_k, there exists a $(k-1)$-color problem f' with no solutions in ∂B_{k-1} and with $\deg(f) \equiv \deg(f') \pmod 2$.

We approach the computation of $\deg(f)$ by looking at the winding number of the boundary around the target point, because it turns out that this quantity equals

[1] The precise definition of degree uses homology groups. Given a set C and a specified subset D, these groups are denoted $H_i(C, D)$ and called "the ith homology group of C modulo D." The elements of $H_i(C, D)$ are classes of formal sums of continuous functions mapping an i-dimensional ball or simplicial complex Δ^i into C, with the requirement that the boundary of Δ^i must map into D. Two of these formal sums belong to the same homology class if they can be continuously deformed into one another (i.e., are "homotopic"). For the domain and range of the color function f, the $(k-1)$st homology groups $H_{k-1}(B_k, \partial B_k)$ and $H_{k-1}(A_k, A_k - \bar{a})$ are infinite cyclic, i.e. isomorphic to $\mathbf{Z}$. The generator g of $H_{k-1}(B_k, \partial B_k)$ is a standard homeomorphism of a $(k-1)$-simplex into B_k, since B_k is $(k-1)$-dimensional. The generator g' of $H_{k-1}(A_k, A_k - \bar{a})$ maps the $(k-1)$-simplex into a ball centered at $\bar{a}$; this covers $\bar{a}$ exactly once. The map f induces a homomorphism f_* between these groups. The *degree* of f is defined to be the value of d such that $f_*(g) = d \cdot g'$, where f_* is the induced homomorphism on the top-dimensional homology group. In other words, f maps B_k into something that covers the target point $\deg(f)$ times. For example, if the entire image lies in the punctured set $A_k - \bar{a}$, so that $f^-(\bar{a}) = \emptyset$, then $f_*(g)$ is in the homology class that is the 0 multiple of the generator, and $\deg(f) = 0$.

deg (f). Let ∂f denote the restriction of f to ∂B_k. Under the assumption that the boundary does not hit the target point, we have $\partial f: B_k \to A_k - \bar{a}$. The *winding number* $W(f|D, \bar{a})$ is defined to be the degree of the map $f|D: D \to A_k - \bar{a}$ (technically, the degree of the map $f|D: (D, \emptyset) \to (A_k - \bar{a}, \emptyset)$); we will compute $W(\partial f, \bar{a})$.[2] An elegant result in topology states that $W(\partial f, \bar{a}) = \deg(f)$, using the fact that $H_i(C, D)$ and $H_{i-1}(D, \emptyset)$ are isomorphic when C is contractible.[3]

The winding number can be computed by following any fixed reference direction l^+ from the target point and counting crossings with the image surface, a fact that is intuitively clear in two dimensions. Of course, the crossings must be counted with appropriate sign, depending on whether the ray enters or leaves the region of interest. Fixing the ray l^+ from $\bar{a}$ and counting the crossings with the proper sign yields the *intersection number* $I(\partial f, l^+)$. Being a homology concept, $I(\partial f, l^+)$ is invariant under continuous deformations of f or small changes in l^+. The relevant topological fact is that $W(\partial f, \bar{a}) = I(\partial f, l^+)$.[4] If f does not raise dimension, then $f(C) \cup f(D) \cup f(\partial B^+) \cup$

---

[2] The winding number is a useful and sensible construct only when $H_{\dim(D)}(D, \emptyset)$ is infinite cyclic (isomorphic to $\mathbf{Z}$), which holds here for $D = \partial B_k$. To compute the winding number, we need the induced homomorphism f_* between the $(k-2)$nd homology groups, since the dimension of ∂B_k is $k-2$. $H_{k-2}(\partial B_k, \emptyset)$ and $H_{k-2}(A_k - \bar{a}, \emptyset)$ are both infinite cyclic. When the second argument to $H_i(C, D)$ is the null set, the formal sums in each homology class must be such that all the image contributions arising from the boundary of the i-dimensional domain cancel. In other words, the generator g of $H_{k-2}(\partial B_k, \emptyset)$ covers ∂B_k exactly once with $(k-2)$-dimensional simplices whose boundaries cancel out. Similarly, the generator g' of $H_{k-2}(A_k - \bar{a}, \emptyset)$ covers a sphere centered at $\bar{a}$ exactly once. $W(\partial f, \bar{a})$ is the value of d such that $f_*(g) = d \cdot g'$.

[3] This result can be explained as follows. Given a set C and subset D, there is a natural sequence of homology groups

$$H_i(C, \emptyset) \to H_i(C, D) \to H_{i-1}(D, \emptyset) \to H_{i-1}(C, \emptyset),$$

in which the arrows represent natural group homomorphisms. The outer homomorphisms are inclusion mappings. The central one is the "boundary mapping." (To define the boundary mapping, consider a map $\sigma: \Delta^i \to C$ belonging to one of the homology classes in $H_i(C, D)$. Its restriction $\partial\sigma$ maps $\partial\Delta^i$ to D. Since $\partial\Delta^i$ is a union (formal sum) of $(i-1)$-simplices, define σ_j by restricting σ to the jth simplex in $\partial\Delta^i$. As a formal sum, it follows that $\partial \sum \sigma_j = 0$ (since $\partial^2 = 0$ by cancellation), so indeed $\sum \sigma_j$ belongs to an element of $H_{i-1}(D, \emptyset)$.) This sequence of homomorphisms is an exact sequence, in the sense of group homomorphisms (the image of the previous map is the kernel of the next). However, if C is contractible, i.e. has no holes, then the groups at the ends of the sequence are the trivial group; the only formal sum of continuous maps that can take the boundary of Δ^i to the empty set is the 0 formal sum. Since the sequence is exact, the two groups in the middle are isomorphic. Applying this to $(B_k, \partial B_k)$ and $(A_k, A_k - \bar{a})$, we find that $H_{i-1}(\partial B_k, \emptyset)$ and $H_{i-1}(A_k - \bar{a}, \emptyset)$ are also infinite cyclic. Considering the induced homomorphisms $f^0_*: H_i(B_k, \partial B_k) \to H_i(A_k, A_k - \bar{a})$ and $f^1_*: H_{i-1}(\partial B_k, \emptyset) \to H_{i-1}(A_k - \bar{a}, \emptyset)$ yields a commutative diagram, with two ways to get from $H_i(B_k, \partial B_k)$ to $H_{i-1}(A_k - \bar{a}, \emptyset)$. The diagram is commutative because, for a mapping $\Delta^i \to B_k$ that is restricted to $\partial\Delta^i$ and composed with f, it doesn't matter whether the restriction or the composition happens first. If g and g'' are the generators, then following f^0_* by the A-isomorphism takes g to $d \cdot g''$, where $d = \deg(f)$. Following the B-isomorphism by f^1_* takes g to $w \cdot g''$, where $w = W(\partial f, \bar{a})$. Since the two routes yield the same result, $\deg(f) = W(\partial f, \bar{a})$.

[4] This fact has a simpler explanation than that of the previous footnote. The intersection number is defined for any oriented i-dimensional surface intersecting an oriented $(d-i)$-dimensional surface in d-space. Given orientations for each, i.e. coordinate systems, the combined coordinate system at an intersection has a well-defined sign, corresponding to "left-handed" or "right-handed." In particular, l^+ is an oriented 1-dimensional surface and $f(\partial B_k)$ is an oriented $(k-2)$-dimensional surface in $(k-1)$-space, which means that $f(\partial B_k)$ has a well-defined "top" and "bottom" (given that f does not raise dimension). As you travel l^+ away from $\bar{a}$, if you cross $f(\partial B_k)$ from top to bottom count -1, and from bottom to top count $+1$. Tangency causes no problem, because you can count $\pm\frac{1}{2}$ when you enter and $\pm\frac{1}{2}$ when you leave, in the appropriate way. Now consider the definition of $W(\partial f, \bar{a})$ using degree and the description of the generator in $H_{k-2}(A_k - \bar{a}, \emptyset)$. Note that $W(\partial f, \bar{a})$ is the number of times $f(\partial B_k)$ covers each point when radially projected onto a reference sphere centered at $\bar{a}$ (counted with appropriate sign in case of folds). This is precisely $I(\partial f, l^+)$ for any l^+.

$f(\partial B^-)$ has dimension at most $k-2$, by Lemma 1(d). Thus, almost every direction l^+ misses $f(C)\cup f(D)\cup f(\partial B^+)\cup f(\partial B^-)$, in which case $(\partial f)^-(l^+)$ consists only of points in $\operatorname{int}(B^+)\cup\operatorname{int}(B^-)$. If $x\in\operatorname{int}(B^-)$ and $f(x)\in l^+$, then for the "antipodal point" $x^*\in B^+$ under the pairing established in Lemma 1(b), $f(x^*)=2\bar{a}-f(x)$ belongs to l^-, the opposite ray from $\bar{a}$. Conversely, if $x\in\operatorname{int}(B^+)$ and $f(x)\in l^-$, then $x^*\in B^-$ and $f(x^*)\in l^+$. Consequently, $I(f|B^-, l^+)=I(f|B^+, l^-)$. Let $l=l^+\cup l^-$ be the full line through $\bar{a}$. Note that orienting l requires us to change the orientation on l^-. We have proved

$$W(\partial f,\bar{a})=I(\partial f, l^+)=I(f|B^+, l^+)+I(f|B^-, l^+)=I(f|B^+, l^+)+I(f|B^+, l^-)$$

$$\equiv I(f|B^+, l^+)-I(f|B^+, l^-)=I(f|B^+, l) \qquad \text{(congruence mod 2)}.$$

All that remains is to obtain the desired $(k-1)$-colored problem. We do this by projecting along an appropriate line l into a space with fewer colors. Let $A_{k-1}\subseteq A_k$ be those k-tuples whose last coordinate is 0. Intuitively, the most satisfying projection to use is the one that amalgamates the last two colors, although any direction will work as long as it is not parallel to A_{k-1}, does not hit $f(C)\cup f(D)$, and does not hit $f(\partial B^+)$. Within a hyperplane, such as A_k, a direction is specified by a k-tuple whose coordinates sum to 0. Linear projection along a direction simply adds a multiple of that k-tuple. Projection along the direction $(0,\cdots,0,-1,1)$ maps $\bar{b}\in A_k$ to $(b_1,\cdots,b_{k-2},b_{k-1}+b_k,0)\in A_{k-1}$. If f does not raise dimensions, there is a direction arbitrarily close to this that avoids $f(C)\cup f(D)\cup f(\partial B^+)\cup f(\partial B^-)$, and to which we can apply the chain of equalities and congruences we have built under that assumption.

Let π be a projection along such a direction. If $\bar{\rho}(x)=(\rho_1(x),\cdots,\rho_k(x))$ are the color densities at a point x on the circle, then $\pi\bar{\rho}(x)$ are also continuously integrable color densities summing to 1, and this may be considered a $(k-1)$-color problem since the last coordinate is always 0. Note that the projection may make some color density negative, but that is allowed in the class of coloring problems we originally defined. To compute the color function f' for the new problem, note that the color function is a sum of m integrals of the color densities. Summation and integration commute with linear projection, so $f'=\pi\cdot f$ is the $(k-1)$-color problem with densities $\pi\cdot\bar{\rho}(x)$: f counts the original colors, and π redistributes them.

The parameter space for f' is simply B^+, which by Lemma 1(a) is isomorphic to B_{k-1}. The target point in f' is $\pi(\bar{a})$. In fact, the entire line l maps under π to $\pi(\bar{a})$. This leads us back to $\deg(f')$, using the fact that the projection of f crosses the projection of l whenever f crosses l, i.e. $I(f|B^+, l)=I(\pi f|B^+, \pi(l))$. Since we chose a direction that did not intersect $f(\partial B^+)\cup f(\partial B^-)$, the projected problem satisfies the hypothesis of the lemma; it has no solutions in the boundary of its parameter space. By induction, $\deg(f')$ is odd). But now $\deg(f)$ is also odd, since

$$\deg(f)=W(\partial f,\bar{a})=I(\partial f, l^+)\equiv I(f|B^+, l)$$

$$=I(\pi f|B^+, \pi(l))=\deg(\pi f|B^+)=\deg(f').$$

The only remaining detail is the assumption we have made throughout this argument that f does not increase dimension. As the Peano space-filling curves show, continuous functions can raise dimension. However, this lemma still holds for such functions, because we can apply it to a suitable simplicial approximation $\hat{f}$, and simplicial functions

never raise dimension.[5] The simplicial function is homotopic to f, which means that it has the same degree, so deg $(\hat{f})$ odd implies deg (f) is also odd.

5. Bisection algorithm. Let us now return to the discrete case to discuss algorithms for finding bisections. When the number of colors in the necklace is fixed at k, the guarantee that a bisection with at most $m = \lceil k/2 \rceil$ intervals exists yields a straightforward algorithm to find the smallest bisection. This simple algorithm uses exhaustive search. An m-snare is determined completely by the location of the k "cuts" between captured and noncaptured beads, i.e. the endpoints of the intervals. (Note that if k is odd we have placed an arbitrary cut in advance, so k more determine the snare.) However, the k cuts cannot be placed with complete freedom, since the total number of beads included and excluded must both equal n. The simplest way to implement this restriction is to search the entire parameter space B_k. The snares represented many times are those with fewer intervals, and there are fewer of these. In other words, the number of points in B_k has the same order as the number of snares.

The number of integer points in a d-dimensional simplex whose $d+1$ variables sum to n is $\binom{n+d}{d}$. Thus X has $\binom{n+m-1}{m-1}$ integer points, and Y has that many or $\binom{n+m}{m}$, depending on the parity of k. In either case, the number of integer points in B_k is the product of polynomials in n of degrees $\lceil k/2 \rceil - 1$ and $\lfloor k/2 \rfloor$, so it is $O(n^{k-1})$.

After $O(n)$ operations for preprocessing, the image of any snare can be computed in constant time (i.e., $O(k)$), using the following idea due to Leiserson [7]. In one pass through the necklace, compute the cumulative distributions for the colors. Then, to compute the color amounts captured by a snare, sum the differences of these distributions between the endpoints of the m intervals in the snare. In the boundary, of course, we need only compute the images of the points in B^+. The search finds a bisection from *every* complementary pair of bisections with at most $\lceil k/2 \rceil$ intervals (and with the specified cut if k is odd). It runs in $O(n^{k-1})$ time.

If $k \geqq 3$, we can save one factor of n by using the topological ideas in the proof that $\lceil k/2 \rceil$ intervals suffice. A "divide-and-conquer" search of the parameter space B_k runs in $O(n^{k-2})$ time. As with binary search, to which this reduces when $k = 2$, we split the parameter space into pieces, determine the piece in which to search for the desired point, and recurse. With preprocessing as above, the computation of the color function for any point in B_k takes $O(k)$ time. Other operations required will also be polynomial in k but independent of n, so the asymptotic running time of the algorithm will be determined by the number of function evaluations required.

"Divide-and-conquer" yields an $O(n^{k-2})$ algorithm to find a solution in B_k, but it does not find all solutions. To ensure finding the smallest solution, we must test all snares with less than m intervals individually. As noted above, testing the points in B^+ suffices, but B^+ is a copy of B_{k-1}, which has $O(n^{k-2})$ integer points, so testing all these points is cheap enough. Having checked ∂B_k, we can assume there is no solution in the boundary.

To define a concept of winding number in this discrete situation, we use a simplicial map, because there is an easy way to compute intersection numbers for such maps and hence determine the winding number. At the integer points in B_k, define f by the

[5] Given a color function $f: B_k \to A_k$, the simplicial approximation we need must satisfy the antipodal properties obtained in Lemma 1 for B^+ and B^-. To do this, merely choose the vertices for the simplicial decomposition of ∂B_k in antipodal pairs according to the pairing in Lemma 1(b) for f. Let $\hat{f}$ agree with f at these vertices, and let the simplicial decomposition respect the pairing. Define $\hat{f}$ on the rest of B_k by linear interpolation from the vertices. This simplicial map has the antipodal property, and the argument succeeds.

discrete bead-counting color function. The natural unit regions in the grid formed by the integer points are not simplices. For $k=3$ they are squares, and in general they are the product of $(\lceil k/2\rceil-1)$- and $\lfloor k/2\rfloor$-dimensional simplices. Partition these further into simplices to view B_k as a simplicial complex of dimension $k-1$, and extend f to the entire space by linear interpolation on these simplices. For example, when $k=3$ the product of two segments partitions naturally into two triangles, and when $k=4$ the product of a triangle with a segment partitions naturally into three tetrahedrons.

The computation of winding number uses the value of f at the integer boundary points, the extension of f by linear interpolation, and the idea that winding number equals intersection number. Let D be a simplicial subcomplex of B_k, and suppose that $f(\partial D)$ does not include the target point $\bar{a}$. Then we want to compute $W(f|\partial D, \bar{a})$ by counting the signed intersections of ∂D with some ray emanating from $\bar{a}$. The boundary ∂D consists of simplices of dimension $k-2$, since B_k itself has dimension $k-1$. Since f is simplicial, the image of a boundary simplex Δ is the simplex of dimension at most $k-2$ whose vertices are images of the vertices of Δ. So, the problem of computing contributions to the intersection number reduces to deciding when a specified ray crosses a simplex determined by specified points, and in which "direction."

Let l^+ be an arbitrary direction vector in A_k, let $b_1, \cdots, b_{k-1}$ be the vertices of a simplex Δ in ∂D, and let $c_i = f(b_i) - \bar{a}$. Then $f(\Delta)$ intersects the ray $\bar{a}+tl^+$ $(t>0)$ if and only if l^+ belongs to the convex cone determined by $\{c_i\}$. Since f is simplicial, they can intersect only once, unless the ray is tangent. Membership in this convex cone is equivalent to the feasibility of $\alpha_0 l^+ = \sum \alpha_i c_i$, with $\alpha_i \geqq 0$ and $\alpha_0 > 0$. This homogeneous system of k equations in k unknowns can be solved by Gaussian elimination in $O(k^3)$ operations. If the system has solutions, it is easy to test whether any satisfy the constraints on the α_i. Let $\bar{\alpha}$ be such a solution, if it exists.

When the homogeneous system has a 1-dimensional solution space, the ray $\bar{a}+tl^+$ "crosses" this part of the boundary. To determine the contribution to the intersection number, determine whether the vectors c_i, taken in order, form a right- or left-handed coordinate system. In other words, find the sign of the determinant composed of these vectors, again using $O(k^3)$ operations. Let ε be this sign. Note that when f collapses dimension, $\varepsilon = 0$. The contribution to the intersection number from this piece of the boundary is

0	if the solution space is multidimensional. The ray is tangent here, and the proper contribution will be computed from neighboring simplices.
ε	if $\bar{\alpha}$ has all $\alpha_i > 0$, so that the intersection occurred in the interior of the simplex.
$\varepsilon/2$	if $\bar{\alpha}$ has exactly one $\alpha_i = 0$. One neighboring simplex shares the intersection point, and the contributions will support each other or cancel.
$\varepsilon/r(j)$	if $\bar{\alpha}$ has j values such that $\alpha_i = 0$, and $r(j)$ simplices share that facet.

Alternatively, the complications of nonunit contributions to the intersection number can be avoided by picking a better direction.

Due to the technicalities of simplicial subdivision for large k, we first describe the algorithm in the 3-color case. Here the discrete parameter space B_3 is the product of two segments of length n, simplicially subdivided. To "divide and conquer" the space, split each of the segments in half; this partitions the parameter space into four smaller squares. Together, the boundaries of these squares contain just under $6n$ integer points. Having applied the algorithm first to B_2, we know that none of these boundary points

describe bisections. To determine which region contains a bisection, we must compute the winding number for f restricted to the boundary of each of the four regions.

Let l^+ be an arbitrary direction in A_3, say $(.5, -.25, -.25)'$. The boundary of each region consists of 1-dimensional unit simplices, i.e. segments with endpoints, b_2. Under the simplicial f, each maps to a segment. The segment $f(b_1)$, $f(b_2)$ intersects the ray $\bar{a}+tl^+$ if l^+ lies in the convex hull of $\{c_1, c_2\}$, in which case the sign of $\det(c_1, c_2)$ determines the contribution to the intersection number. Clearly, these operations can be done in constant time per integer point on the boundary as the segments of the boundary are traversed.

Choosing one of the regions with a nonzero winding number, perform the same search on the $n/2$ by $n/2$ grid. The number of function evaluations performed to find a solution point is at most $\sum 6n/2^i = 12n$. If $O(n)$ storage is available, then the contributions obtained from the boundary segments of the current region can be stored when computed, so that that information need not be recomputed when they recur in boundaries of smaller regions. This would save a factor of $\frac{1}{3}$, because the number of function evaluations (with the same amount of computation for each), would be $4n+\sum 2n/2^i = 8n$. Note that the preliminary step of checking the boundary for smaller bisections can be dropped, because the first full step of the main algorithm includes that.

A 2-dimensional simplex can be cut into four half-size simplices. If $y_1+y_2+y_3 = n$, these pieces are those with a given $y_i \geqq n/2$, and the piece where all $y_i \leqq n/2$. Dividing B_4, the product of a 2-simplex and a 1-simplex, partitions it into eight pieces; dividing B_5, the product of two 2-simplices, partitions it into sixteen pieces. The boundaries of these pieces still have approximately one lower-dimensional simplex for each vertex, and can be traversed using a small number of paths in which only one vertex changes between neighboring simplices. B_4 has approximately $4.5n^2+3\binom{n+2}{2}$ of these boundary vertices; B_5 has approximately $9n\binom{n+2}{2}$. With divide-and-conquer, the total number of function evaluations is $\sum 6(n/2^i)^2+o(n^2) = 8n^2+o(n^2)$ for B_4 and $\sum 4.5(n/2^i)^3+o(n^3) = 36n^3/7+o(n^3)$ for B_5. Storage of the outer boundary as before permits a reduction in the leading constant, but the factor saved decreases as k increases. For $k = 3, 4, 5$, it is $\frac{1}{3}$, $\frac{1}{6}$, $\frac{1}{12}$.

In general, a d-dimensional simplex can be cut into 2^d half-size d-dimensional simplices. The number of vertices in the boundaries of the various pieces is $O(n^{k-2})$, and using divide-and-conquer allows the entire algorithm to run in $O(n^{k-2})$.

6. Application to graph separators. Leighton pointed out an application of the discrete result to separators in graphs. For present purposes, we define an *$f(n)$-separator* of a graph to be a balanced binary tree with the vertices of the graph as leaves and no more than $f(n)$ edges of the graph between vertices in different subtrees of a subtree with n leaves. By "balanced," we mean half of the leaves in any subtree belong to each of its subtrees. When the induced subgraph on the n leaf nodes of some partial tree has its nodes split into its "left" and "right" sets, the induced edge cut has "few" edges, i.e. bounded by $f(n)$.

By applying the circle coloring result, it is possible to obtain a more refined splitting of the nodes at the cost of some extra edges in the cut. In particular, suppose the nodes come in k "types," i.e. are labeled arbitrarily with k colors, and let $|H|$ be the number of vertices in H. Then

THEOREM 4. *Suppose any induced subgraph H of a graph G has an $f(|H|)$-separator and the nodes of G are labeled arbitrarily with k colors. If $f(n) = O(n^j)$ for some $j > 0$, then G has an $O(kn^j)$-separator that separates the vertices of each color as evenly as*

possible at each level, in addition to separating the full vertex set as evenly as possible. If $f(n) = O((\log n)^j)$, *then* G *has an* $O((\log n)^{j+1})$*-separator as described.*

To explain this result more directly, we "unbend" the necklace result to an equivalent result about linear arrangements that eliminates the distinction between odd and even values of k. A linear arrangement with k colors of beads has a bisection with at most k cuts. In reading from one end to the other, we switch from capturing beads to omitting beads each time we cross a cut. If the two ends are identified to become the reference point P, this is precisely the necklace result. When k is odd, the first and last intervals are of opposite type, so that the corresponding intervals on the circle end at P, and there are $\lceil k/2 \rceil$ intervals used and omitted. When k is even, the first and last intervals on the line have the same type, so that in the circle version they merge at P, and again there are $\lceil k/2 \rceil$ intervals of each type. Thus "k cuts on the line segment" is a uniform way to state the result.

Any separator yields a particular ordering of the vertices of a graph, by reading the leaf nodes in order. Since the vertices have specified colors, this yields a linear arrangement of beads. Find a bisection with at most k cuts. Taking the intervals of captured beads yields a vertex partition of the original graph in which the vertices of each color are evenly split. To obtain an upper bound on the number of edges between the two parts, we take k times the maximum number of edges that can join vertices on opposite sides of a single cut. The endpoints of any such edge belong to opposite subtrees at some level. At most $f(n/2^i)$ edges cross the cut at level i. If $f(n) = O(n^j)$ for some $j \geqq 0$, then there are $O(kn^j)$ edges across the vertex partition. If $f(n) = O((\log n)^j)$, then there are $O(k(\log n)^{j+1})$ edges across the partition. To build the rest of the desired separator, consider each of the vertex parts separately, find the $f(n/2)$-separator on the subgraphs induced by those vertices, and recurse.

7. Continuous applications. Tom Trotter pointed out an application of the continuous result to a geometric problem in $\mathbf{R}^3$. Consider a curve $\bar{u}(t) = (u_1(t), u_2(t), u_3(t))$, $0 \leqq t \leqq 2$. An old problem [4], [5] asks whether a curve in $\mathbf{R}^2$ always contains four points that determine a square. In $\mathbf{R}^3$ certainly one cannot hope for so much, but one can hope for a parallelogram. Actually, we can guarantee more. Every continuous curve in $\mathbf{R}^3$ contains four points that determine a parallelogram such that the opposite portions of the curve total half the length of the curve.

THEOREM 5. *If* $\bar{u}(t)$ *with* $0 \leqq t \leqq 2$ *is a continuous curve in* $\mathbf{R}^3$, *then there are four points* $0 \leqq t_1 \leqq t_2 \leqq t_3 \leqq t_4 \leqq 2$ *such that* $\bar{u}(t_2) - \bar{u}(t_1) = \bar{u}(t_3) - \bar{u}(t_4)$ *and* $t_2 - t_1 + t_4 - t_3 = 1$.

Proof. Define a four-color circle coloring with density function $\rho_i(t) = u_i'(t)$ for $i = 1, 2, 3$ and $\rho_4(t) = 1 - u_1'(t) - u_2'(t) - u_3'(t)$. The derivatives may be discontinuous at isolated points, but they are continuously integrable and can be defined at discontinuities so that they sum to 1 at all points. The cumulative distributions yield the displacements of $u_i(t)$ from $u_i(0)$ and t minus the sum of those displacements. Integrating all the way from 0 to 2 yields the total amount of each color, which is 0 for the first three colors and 2 for the fourth, since the net displacement in any direction around the closed curve is 0.

Applying Theorem 2, let (t_1, t_2, t_3, t_4) be the endpoints of intervals in a bisection. Given the above "total amounts" of each color, we have

$$0 = u_i(t_4) - u_i(t_3) + u_i(t_2) - u_i(t_1) \quad \text{for } i = 1, 2, 3,$$

$$1 = (t_4 - \textstyle\sum u_i(t_4)) - (t_3 - \sum u_i(t_3)) + (t_2 - \sum u_i(t_2)) - (t_1 - \sum u_i(t_1)) \quad \text{for } i = 4,$$

which yields $\bar{u}(t_4) - \bar{u}(t_3) = \bar{u}(t_1) - \bar{u}(t_2)$ and $t_4 - t_3 + t_2 - t_1 = 1$. □

Narendra Karmarkar noticed another application, of which the following is a variant.

THEOREM 6. *Let $f_1, \cdots, f_{k-1}$ be $k-1$ continuously integrable functions on an interval $[a, b]$. Then there is a function orthogonal to each of $f_1, \cdots, f_{k-1}$ that takes on only the values ± 1 and changes sign at most k times in the interval $[a, b]$.*

Proof. Introduce a kth function that is 1 minus the sum of the others. Apply the linear version of the continuous bisection result to obtain a bisection using at most k cuts. Define the new function to be $+1$ on the intervals in the bisection and -1 on those in its complement. □

Karmarkar actually noted the corresponding result for periodic functions, using the version of the continuous result applicable to circle colorings.

8. Conclusion. In addition to the NP-completeness of the original problem, related problems remaining open include generalizations to higher dimensions and splits in other proportions. If intervals must be chosen to capture a fraction α of each color, it is no longer always possible to do it with $\lceil k/2 \rceil$ intervals. For example, consider the following arrangement with four colors of beads: let the beads of colors 1, 2, 3 appear contiguously, and put $\frac{1}{3}$ of the beads of color 4 between color 1 and color 2, between color 2 and color 3, and between color 3 and color 1. Restricted to two intervals, there must be a cut within each of colors 1, 2, 3, and the fourth cut appears somewhere. This means that one stretch of color 4 is entirely included, and one stretch is entirely omitted, so the fraction of color 4 captured must lie between $\frac{1}{3}$ and $\frac{2}{3}$. The question is, for what range of values of α will $\lceil k/2 \rceil$ intervals suffice? If $\alpha = 1/l$, a more difficult variation would be to find l disjoint sets of intervals such that each set contains $1/l$ of each color. Here one might want to minimize the total number of intervals or the maximum in any set.

In moving to higher dimensions, the torus and 2-simplex may be considered. For the discrete or continuous torus, a simple analogue of choosing intervals would be choosing an even number of horizontal and vertical lines to create a "checkerboard," capturing the units in the regions having a given parity in the checkerboard. However, it is no longer clear that a bisection must always exist. For colorings of a simplex, the bisection result for linear arrangements is the candidate for generalization. Given a triangulated simplex, again cuts could be made parallel to the boundaries to capture the material in the resulting regions of a given parity, but here there seems to be an example with two colors where such a bisection does not exist. There may be an analogue of the continuous result for nonrectilinear cuts in a simplex, for regions formed on the surface of a sphere by passing planes through the origin, etc.

REFERENCES

[1] S. N. BHATT AND C. E. LEISERSON, *How to assemble tree machines* (extended abstract), Proc. 14th Symposium on the Theory of Computing, San Francisco, 1981, pp. 99–104.
[2] J. DUGUNDJI, *Topology*, Allyn and Bacon, New York, 1966.
[3] V. GUILLEMIN AND A. POLLACK, *Differential Topology*. Prentice-Hall, Englewood Cliffs, NJ, 1974.
[4] R. P. JERRARD, *Inscribed squares in plane curves*, Trans. Amer. Math. Soc., 98 (1961), pp. 234–241.
[5] V. KLEE, *Some unsolved problems in plane geometry*, Math. Magazine, 52 (1979), pp. 131-145.
[6] F. T. LEIGHTON, *A layout strategy for* VLSI *which is provably good*, (extended abstract), Proc. 14th Symposium on the Theory of Computing, San Francisco, 1981, pp. 85-98.
[7] C. E. LEISERSON, private communication.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 1, January 1985

011

ON THE DISCRETE RICCATI MATRIX EQUATION*

MINH THANH TRAN† AND MAHMOUD E. SAWAN‡

Abstract. In this note, the inequality which is satisfied by the determinant of the positive definite solution of the discrete algebraic Riccati matrix equation is presented. The result gives lower bound for the product of the eigenvalues of the matrix solution.

1. Introduction. The discrete algebraic Riccati matrix equation has been used widely in various areas of engineering system theory, particularly in control system theory. The techniques of solving this equation numerically are well-established. Those techniques are mostly iterative algorithms which require making an initial guess of the solution. So if the initial guess is chosen wisely, one can save a lot of unnecessary computations. Therefore, to obtain a precise estimate of the "size" of the solution, we provide here a lower bound for the determinant of the matrix solution of the algebraic Riccati equation.

In the following, the notations X^T, $\lambda_i(X)$, tr (X) and $|X|$ denote the transpose, eigenvalue, trace and determinant of the matrix X, respectively. Also for our derivation later, we will make use of the following results [1, p. 70], [2, p. 225].

i) For any $n\times n$ matrices L and H with $L>0$

$$\text{tr}\,(L^{-1}HLH^T)\geqq\sum_{i=1}^{n}|\lambda_i(H)|^2\geqq\frac{1}{n}[\text{tr}\,(H)]^2. \tag{1}$$

ii) For any real $n\times n$ matrices R and S such that $R=R^T>0$, $S=S^T>0$

$$|R|^{1/n}=\min_{|S|=1}\frac{\text{tr}\,(RS)}{n}. \tag{2}$$

iii) For any $m\times n$ matrix Y, $n\times m$ matrix Z, $n\times n$ matrix W and $m\times m$ matrix X, we have the property

$$[W+ZX^{-1}Y]^{-1}=W^{-1}-W^{-1}Z[X+YW^{-1}Z]^{-1}YW^{-1}. \tag{3}$$

2. Main result. In this section, we derive a lower bound for the determinant of the solution of the discrete algebraic Riccati matrix equation

$$P=A^TPA-A^TPB(I+B^TPB)^{-1}B^TPA+Q, \tag{4}$$

where A, P, $Q\in R^{n\times n}$, $B\in R^{n\times m}$, $Q=Q^T>0$.

Here we assume $|BB^T|\geqq|Q|$ and the matrix A is stable, therefore the solution matrix P is positive definite.

THEOREM. *The determinant of the positive definite matrix solution P of* (4) *satisfies the following inequality*:

$$|P|\geqq\left[\frac{M+(M^2+4n^2|BB^T|^{1/n}|Q|^n)^{1/2}}{2n|BB^T|^{1/n}}\right]^n \tag{5}$$

where $M=\sum_{i=1}^{n}|\lambda_i(A)|^2+\text{tr}\,(BB^TQ)-n.$

* Received by the editors June 30, 1983. This work was presented at the SIAM Second Conference on the Applications of Discrete Mathematics, held at Massachusetts Institute of Technology, Cambridge, Massachusetts, June 27–29, 1983.

† Boeing Military Airplane Company, Wichita, Kansas 67210.

‡ Electrical Engineering Department, Wichita State University, Wichita, Kansas 67208.

Proof. Using (3) with $W = P^{-1}$, $Z = B$, $X^{-1} = I$ and $Y = B^T$, (4) becomes

$$P = A^T[P^{-1} + BB^T]^{-1}A + Q. \tag{6}$$

Multiplying (6) by $(P^{-1} + BB^T)$ from the left yields

$$[P^{-1} + BB^T]P = [P^{-1} + BB^T]A^T[P^{-1} + BB^T]^{-1}A + [P^{-1} + BB^T]Q. \tag{7}$$

Computing the traces of both sides of (7), using (1) with $L^{-1} = [P^{-1} + BB^T]$, $H = A^T$ and rearranging terms, we have

$$\operatorname{tr}(BB^TP - P^{-1}Q) \geqq M. \tag{8}$$

Now using (2) with $R = P$ and noting that $|X| = 1/|X^{-1}|$ and $\operatorname{tr}(X) = \operatorname{tr}(X^T)$, (8) becomes

$$n|BB^T|^{1/n}|P|^{2/n} - M|P|^{1/n} - n|Q|^{1/n} \geqq 0. \tag{9}$$

Solving (9) for $|P|^{1/n}$, we get the inequality (5). Q.E.D.

3. Conclusion of the solution. The inequality (5) makes it possible to estimate a lower bound for the determinant of the discrete algebraic Riccati matrix equation. This bound does not require A to be nonsingular and only requires a few matrix computations. These computations can even be further simplified by comparing the first and last terms of (1) instead of the middle term. However the tightness of the bound may reduce considerably.

REFERENCES

[1] E. F. BECKENBACK AND R. BELLMAN, *Inequalities*, Springer-Verlag, Berlin, 1965.

[2] R. V. PATEL AND M. TODA, *Modeling error analysis of stationary linear discrete-time filters*, NASA, TMX-73, Ames Research Center, Moffett Field, CA, Feb. 1977.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 1, January 1985

012

COVERING, PACKING AND GENERALIZED PERFECTION*

GERARD J. CHANG† AND GEORGE L. NEMHAUSER‡

Abstract. Given a graph $G=(V, E)$, let $T_k=(V(T_k), E(T_k))$ be a tree of diameter $\leqq k$ that is a partial graph of G. Let $\mathcal{T}_k$ be the set of $V(T_k)$ for all such T_k. We consider covering and packing problems defined with respect to $\mathcal{T}_k$, for all integers $k \geqq 1$. $\theta(G: \mathcal{T}_k)$ is the minimum number of elements of $\mathcal{T}_k$ that cover V, and $\alpha(G: \mathcal{T}_k)$ is the maximum size of a $P \subseteq V$ such that no element of $\mathcal{T}_k$ contains more than one element of P. In particular, $\theta(G: \mathcal{T}_1)$ is the edge covering number of G, $\theta(G: \mathcal{T}_2)$ is the domination number of G, and $\alpha(G: \mathcal{T}_1)$ is the stability number of G. We study classes of chordal graphs for which $\theta(H: \mathcal{T}_k)=\alpha(H: \mathcal{T}_k)$ for all induced subgraphs H of G. We also give simple algorithms for solving these problems on some classes of graphs. These results are applicable to location problems.

Key words. combinatorial optimization, graph theory, domination, stability, chordal graphs

AMS(MOS) subject classifications. 05C70, 90C10

1. Introduction. Graph covering and packing problems provide a generic model for discrete facility location. Location problems have many applications in operations research, two general references are Christofides [1975] and Tansel et al. [1981]. As an example, consider a geographical area that is partitioned into regions. Facilities are going to be placed in some of the regions. We construct a graph whose vertices represent the regions and whose edges represent pairs of regions that are adjacent. A region (vertex) and all of the regions that are adjacent to it is called a neighborhood.

In this context, we will consider a minimum covering problem, a maximum packing problem and the relationship between them. Suppose each neighborhood is to be served by a costly, but necessary, facility such as a school or hospital. The covering problem is to choose regions at which to place these service facilities so that each neighborhood contains at least one and the number of them is minimum.

Now suppose that a company wishes to select regions for the location of noxious facilities such as pollution producing factories. The packing problem is to place as many of these facilities as possible, subject to the constraints that each neighborhood contains no more than one of them.

There is an interesting min-max duality relation between the packing and covering location problems. Consider a feasible solution to the covering problem, i.e. a set of regions for service facilities such that each neighborhood of the area contains at least one service facility. Thus the subset of neighborhoods induced by the regions at which these facilities are located contains all of the regions. Now in any feasible solution to the packing problem, none of these neighborhoods can contain more than one noxious facility. But since these neighborhoods contain all of the regions, their number is an upper bound on the number of noxious facilities that can be placed in the area. This inequality is true for any pair of feasible covering and packing solutions. Thus we have proved that the maximum value of the packing problem is equal to or less than the minimum value of the covering problem. We call this relationship weak min-max duality and say that strong min-max duality holds when the two values are equal.

* Received by the editors March 4, 1983. This research was supported by the National Science Foundation under grant ECS-8005350 to Cornell University. This work was presented at the SIAM Second Conference on the Applications of Discrete Mathematics, held at Massachusetts Institute of Technology, Cambridge, Massachusetts, June 27–29, 1983.

† Department of Mathematics, National Central University, Ching-li, Taiwan 320, Republic of China.

‡ School of Operations Research and Industrial Engineering, College of Engineering, Cornell University, Ithaca, New York 14853.

In this paper, we are interested in characterizing problems of this type for which strong duality holds. Strong duality is an important, if not essential, property used in the construction of algorithms. The reason is that when strong duality holds we can decide the optimality of feasible solutions to the covering and packing problems simply by checking whether both solutions have the same value. Thus, even when we are interested in only one of the two problems, we frequently construct algorithms that solve both.

We now give a formal description of graph covering and packing problems. Let $G=(V, E)$ be a simple graph, i.e. finite, undirected, loopless and without multiple edges. Let $\mathcal{S}$ be a family of subsets of V. An $\mathcal{S}$-covering of G is a subset $\mathcal{F}(\mathcal{S})$ of $\mathcal{S}$ with the property that

$$\cup\{F: F\in\mathcal{F}(\mathcal{S})\}=V. \tag{1.1}$$

An *$\mathcal{S}$-packing of* G is a subset $\mathcal{P}(\mathcal{S})$ of V with the property that

$$|\mathcal{P}(\mathcal{S})\cap S|\leqq 1 \quad \text{for all } S\in\mathcal{S}. \tag{1.2}$$

For any pair $(\mathcal{P}(\mathcal{S}), \mathcal{F}(\mathcal{S}))$ of packings and coverings, (1.1) and (1.2) imply

$$|\mathcal{P}(\mathcal{S})|\leqq\sum_{F\in\mathcal{F}(\mathcal{S})}|\mathcal{P}(\mathcal{S})\cap F|\leqq|\mathcal{F}(\mathcal{S})|. \tag{1.3}$$

Let $\alpha(G:\mathcal{S})=\max|\mathcal{P}(\mathcal{S})|$ and $\theta(G:\mathcal{S})=\min|\mathcal{F}(\mathcal{S})|$. $\alpha(G:\mathcal{S})$ and $\theta(G:\mathcal{S})$ are called the *$\mathcal{S}$-packing* and *$\mathcal{S}$-covering* numbers of G, respectively.

Relation (1.3) implies the *weak min-max duality* relation

$$\alpha(G:\mathcal{S})\leqq\theta(G:\mathcal{S}) \tag{1.4}$$

for all G and $\mathcal{S}$. The *strong min-max duality* relation is said to hold for G and $\mathcal{S}$ if (1.4) is an equality.

Throughout the paper, we use the following terminology from graph theory. A *partial graph* $G'=(V', E')$ of $G=(V, E)$ is a graph such that $V'\subseteq V$ and $E'\subseteq E$. An *induced subgraph* of G is a partial graph such that $E'=\{(x, y)\in E: x, y\in V'\}$. The *complement* of $G=(V, E)$ is the graph $\bar{G}=(V, \bar{E})$, where $\bar{E}=\{(x, y): (x, y)\notin E\}$.

The subset $V'\subseteq V$ is *independent* or *stable* if the subgraph it induces contains no edges. The subset $C\subseteq V$ is a *clique* if the complement of the subgraph it induces contains no edges.

A graph G is *connected* if there is a path joining every pair of vertices. The *length* of a path in G joining vertices x and y is the number of edges in the path. The *distance* $d_G(x, y)$ *from x to y* in G is the length of a minimum length path from x to y. The *distance from* $V_1\subseteq V$ to $V_2\subseteq V$ is $d_G(V_1, V_2)=\min_{x\in V_1, y\in V_2} d_G(x, y)$. The *kth power* of a graph $G=(V, E)$ is the graph $G^k=(V, E^k)$ where $E^k=\{(x, y): 1\leqq d_G(x, y)\leqq k\}$.

The *n-neighborhood* $\mathrm{Nbd}(x, n)$ of a vertex x is $\{z: d(z, x)=n\}$. The *neighborhood* $\mathrm{Nbd}(x)$ of x is the 1-neighborhood $\mathrm{Nbd}(x, 1)$. If $d(x, y)=k$ is finite and $0\leqq m\leqq k$, then $\mathrm{Bet}(x, m, y)$ denotes the set of all vertices z *between* x and y such that $d(x, z)=m$ and $d(z, y)=k-m$.

The *diameter* of a connected graph G is $d(G)=\max_{x,y\in V} d_G(x, y)$. The *radius* of G is $r(G)=\min_{x\in V} f(x)$, where $f(x)=\max_{y\in V} d_G(x, y)$; x is a *center* of G if $r(G)=f(x)$. The *bi-radius* of G is $\mathrm{br}(G)=\min_{e\in E} b(e)$, where $b(e)=\max_{z\in V} d_G(e, z)$; e is a *bi-center* if $\mathrm{br}(G)=b(e)$. For any connected graph G, $d(G)/2\leqq r(G)\leqq d(G)$ and $(d(G)-1)/2\leqq\mathrm{br}(G)\leqq r(G)\leqq\mathrm{br}(G)+1$.

A cycle of a graph contains a *chord* if two nonconsecutive vertices of the cycle are an edge. A *hole* is a cycle without a chord. An *anti-hole* is the complement of a hole. A hole with n edges (of length n) is called an *n-hole* and is denoted by H_n. An n-hole is called even or odd according to the parity of n.

A graph is a *forest* if it does not contain any holes. A *tree* is a connected forest. A graph is *bipartite* if it does not contain any odd holes. A graph is *chordal* or *triangulated* if it does not contain any holes of length greater than three.

A graph is called *$\mathscr{S}$-perfect* if strong duality holds for all of its induced subgraphs. Let $\mathscr{C}$ be the family of cliques of a graph. The $\mathscr{C}$-packings are stable sets and $\alpha(G\colon \mathscr{C})$ is called the stability number of G. The $\mathscr{C}$-coverings are clique coverings and $\theta(G\colon \mathscr{C})$ is the clique cover number. In the early 1960's, Berge introduced the notion of *$\mathscr{C}$-perfection*, which is referred to as *perfection*, and the problem of characterizing perfect graphs. Bipartite and chordal graphs are perfect (Hajnal and Surányi [1958], and Berge [1960]; also see Berge [1973] and Golumbic [1980]). Lovász [1972] proved the *perfect graph theorem*, which states that a graph is perfect if and only if its complement is perfect. The *strong perfect graph conjecture* states that a graph is perfect if it does not contain an odd hole or odd anti-hole of length at least five. The converse of this conjecture is obviously true since $\alpha(H_{2n+1}\colon \mathscr{C}) = n < n+1 = \theta(H_{2n+1}\colon \mathscr{C})$ for all integers $n \geqq 2$.

2. $\mathscr{T}_k$-covering and packing problems. Given a graph $G=(V, E)$, let $T_k = (V(T_k), E(T_k))$ be a tree of diameter $\leqq k$ that is a partial graph of G. Let $\mathscr{T}_k$ be the set of $V(T_k)$ for all such T_k. The location problems we discussed in the introduction are special cases of $\mathscr{T}_k$-covering and packing problems. In particular, let $N(V) = \{u \in V\colon (u, v) \in E \text{ or } u = v\}$ be the *star* or *closed neighborhood of* $u \in V$. Then $\mathscr{T}_2 = \{N(v)\colon v \in V\}$, the *set of stars of a graph*, is precisely the family of subsets required for the location problems discussed above. If $\mathscr{F}(\mathscr{T}_2)$ is a $\mathscr{T}_2$-cover, then $S = \{v \in V\colon N(v) \in \mathscr{F}(\mathscr{T}_2)\}$ is called a *dominating set* and $\theta(G\colon \mathscr{T}_2)$ is called the *domination number* of G. For all $x \in V$, there exists a $u \in S$ such that $d_G(u, v) \leqq 1$. A $\mathscr{T}_2$-packing is a vertex subset $P \subseteq V$ such that $d(x, y) > 2$ for all pairs of distinct vertices x and y in P.

By considering $\mathscr{T}_k$-coverings for even values of $k > 2$, we allow a facility to serve all regions within distance $k/2$ of itself. Each element of the cover generates a partial graph with radius $\leqq k/2$ and a facility is located at a center of each partial graph.

For odd values of k, $\mathscr{T}_k$-coverings can be related to facilities located on edges. Such facilities can serve any region within distance $(k-1)/2$ from at least one endpoint of the edge. Now each element of the cover generates a partial graph with bi-radius $\leqq (k-1)/2$ and a facility is located at a bi-center of each partial graph.

In this section, we will establish a simple relationship between clique covering and packing and $\mathscr{T}_k$-covering and packing. We also state a well-known characterization of $\mathscr{T}_1$-perfect graphs and characterize graphs that are $\mathscr{T}_k$-perfect for all integers $k \geqq 1$.

Sections 3, 4 and 5 contain new results. In § 3, we give necessary and sufficient conditions for $\mathscr{T}_k$-perfection for all even integers $k \geqq 2$ under the assumption that the perfect graph conjecture is true and prove that these conditions are sufficient for $\mathscr{T}_2$-perfection. This extends our results given in Chang and Nemhauser [1982].

In § 4, we characterize graphs that are $\mathscr{T}_k$-perfect for all integers $k \geqq 2$. Section 5 gives a simple polynomial-time algorithm for determining a maximum cardinality $\mathscr{T}_k$-packing and a minimum cardinality $\mathscr{T}_k$-covering on these graphs. More generally, this algorithm determines a maximum cardinality stable set and a minimum cardinality clique cover for a class of $\mathscr{C}$-perfect graphs that contains chordal graphs.

Let $\mathscr{C}_k$ be the family of cliques of G^k. Since $x, y \in V(T_k) \in \mathscr{T}_k$ implies $d(x, y) \leqq k$, we have $(x, y) \in E^k$. Thus $\mathscr{T}_k \subseteq \mathscr{C}_k$, which implies

$$\theta(G: \mathscr{C}_k) \leqq \theta(G: \mathscr{T}_k). \tag{2.1}$$

Furthermore, if $\{x, y\} \subseteq C \in \mathscr{C}_k$, then the set of vertices on any path of length $\leqq k$ joining x and y is an element of $\mathscr{T}_k$, which implies

$$\alpha(G: \mathscr{T}_k) = \alpha(G: \mathscr{C}_k). \tag{2.2}$$

From the definitions of $\mathscr{C}_k$ and G^k, we have

$$\alpha(G: \mathscr{C}_k) = \alpha(G^k: \mathscr{C}) \quad \text{and} \quad \theta(G: \mathscr{C}_k) = \theta(G^k: \mathscr{C}). \tag{2.3}$$

From (1.4), (2.1), (2.2) and (2.3), we obtain

$$\alpha(G: \mathscr{T}_k) = \alpha(G^k: \mathscr{C}) \leqq \theta(G^k: \mathscr{C}) \leqq \theta(G: \mathscr{T}_k). \tag{2.4}$$

Therefore, G is $\mathscr{T}_k$-perfect if and only if, for all induced subgraphs H of G,

(F1) $\alpha(H^k: \mathscr{C}) = \theta(H^k: \mathscr{C})$.

(F2) $\theta(H^k: \mathscr{C}) = \theta(H: \mathscr{T}_k)$ or, equivalently, if C is a clique in H^k, then there exists a tree T, which is a partial graph of H, of diameter $\leqq k$ such that $C \subseteq V(T)$.

Note that $\mathscr{C}$-perfection of G^k is not necessary for $\mathscr{T}_k$-perfection of G. For example a 9-hole is $\mathscr{T}_2$-perfect, but its square is not $\mathscr{C}$-perfect since the square contains 5-holes.

For any graph, $\mathscr{T}_1$ is the set of edges of the graph. Thus $\alpha(G: \mathscr{T}_1)$ is the maximum number of vertices such that no two are joined by an edge and $\theta(G; \mathscr{T}_1)$ is the minimum number of edges that cover all vertices. It is well known that $\alpha(G: \mathscr{T}_1) = \theta(G: \mathscr{T}_1)$ if G is bipartite. Furthermore, for an odd hole H_{2k+1}, $\alpha(H_{2k+1}: \mathscr{T}_1) = k < k+1 = \theta(H_{2k+1}: \mathscr{T}_1)$, for all integers $k \geqq 1$. Since every nonbipartite graph contains an odd hole, we have

THEOREM 2.1. *A graph is $\mathscr{T}_1$-perfect if and only if it is bipartite.*

$\mathscr{T}_k$-perfection forbids graphs with holes of length $n > k$ and $n \neq 0 \pmod{k+1}$. Thus, it is not surprising that $\mathscr{T}_k$-perfection for all positive integers k characterizes forests. Specifically we have:

THEOREM 2.2. *The following statements are equivalent for any graph G.*

(1) *G is $\mathscr{T}_k$-perfect for all integers $k \geqq 1$,*

(2) *G is $\mathscr{T}_{a_k}$-perfect for all integers $k \geqq 1$ where $a_1 = 1$, $a_2 = 2$, and $a_{k+1} = \prod_{j=1}^{k} (a_j + 1) - 2$ for $k \geqq 2$.*

(3) *G is a forest.*

Proof. (1)$\Rightarrow$(2) is obvious.

(2)$\Rightarrow$(3). Suppose G is not a forest; then it has a hole H_n, $n \geqq 3$. It is easy to see that $\alpha(H_n: \mathscr{T}_k) = \max\{1, \lfloor n/(k+1) \rfloor\}$ and $\theta(H_n, \mathscr{T}_k) = \lceil n/(k+1) \rceil$. Thus if G is $\mathscr{T}_k$-perfect and H_n is a hole of G such that $n > k$, then $n = 0 \pmod{k+1}$. Choose $k \geqq 2$ such that $a_k < n \leqq a_{k+1}$. Since G is $\mathscr{T}_{a_m}$-perfect, n is a multiple of $a_m + 1$ for $m = 1, \cdots, k$. But the $(a_m + 1)$'s are pairwise relatively prime, so $n \geqq \prod_{j=1}^{k} (a_j + 1) = a_{k+1} + 2$. This contradicts the assumption that $n \leqq a_{k+1}$. So G is a forest.

(3)$\Rightarrow$(1). If G is a forest, then any induced subgraph H of G is also a forest. Powers of forests are chordal (see Golumbic [1980, Thm. 4.8]) and thus perfect, i.e. (F1) holds. Since H is a forest, $\mathscr{C}_k = \mathscr{T}_k$. Hence $\theta(H^k: \mathscr{C}) = \theta(H: \mathscr{C}_k) = \theta(H: \mathscr{T}_k)$, i.e. (F2) holds. □

Remark. (3)$\Rightarrow$(1) was proved by Meir and Moon [1975] in a different way. An algorithm for $\mathscr{T}_k$-covering problem on trees was given by Slater [1976] which generalizes the algorithm for $k = 1$ given by Cockayne et al. [1975].

3. Odd-sun-free chordal graphs and $\mathcal{T}_k$-perfection for even k. Chordal graphs are not necessarily $\mathcal{T}_2$-perfect. The graph of Fig. 3.1 provides an example.

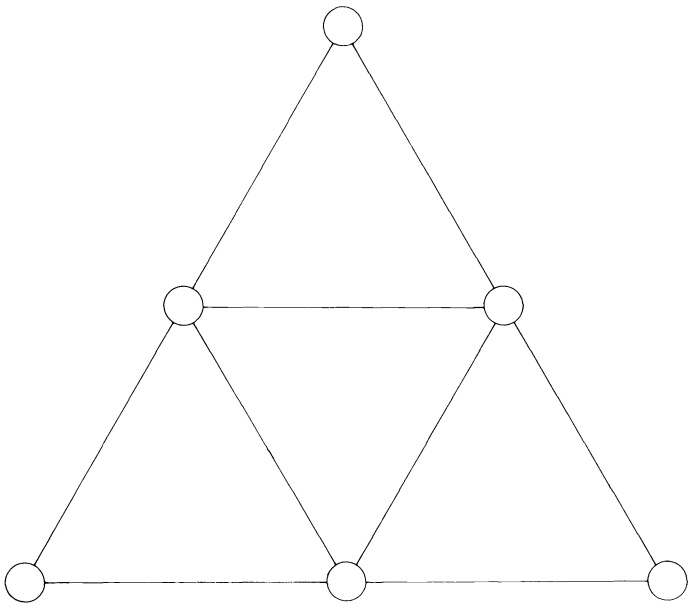


FIG. 3.1. *A chordal graph G with* $\alpha(G: \mathcal{T}_2) = 1 < 2 = \theta(G: \mathcal{T}_2)$.

An *n-sun* is a chordal graph G whose vertex set can be partitioned into $Y = \{y_1, \cdots, y_n\}$ and $Z = \{z_1, \cdots, z_n\}$ such that the following three conditions hold.

(S1) Y is a stable set in G.

(S2) $(z_1, \cdots, z_n, z_1)$ is a cycle in G.

(S3) $(y_i, z_j) \in E$ if and only if $i = j$ or $i = j+1 \pmod{n}$.[1]

In this definition, the z's are called *inner vertices* of the n-sun and the y's *outer vertices.* An *n-anti-sun* is the complement of an n-sun. An n-sun is called *complete* if Z is a clique. An n-sun is called odd or even according to the parity of n. The graph of Fig. 3.1 is a (complete) 3-sun.

A graph that does not contain a sun (3-sun, odd-sun, 3-anti-sun, 3-sun or 3-anti-sun respectively) as an induced subgraph is called *sun-free* (*3-sun-free, odd-sun-free, 3-anti-sun-free, 3-sun-3-anti-sun-free* respectively). For convenience, we use SF-chordal for sun-free chordal, 3SF-chordal for 3-sun-free chordal, OSF-chordal for odd-sun-free chordal, 3ASF-chordal for 3-anti-sun-free chordal, and 3S3ASF-chordal for 3-sun-3-anti-sun-free chordal.

In a previous paper (Chang and Nemhauser [1982]), we proved Theorem 3.1 and gave algorithms for $\mathcal{T}_k$-covering and packing problems on SF-chordal graphs when k is even.

THEOREM 3.1. *If G is* SF*-chordal, then G is* $\mathcal{T}_k$*-perfect for all even integers k.*

The following results from Chang and Nemhauser [1982] were used to prove Theorem 3.1 and are also needed here.

PROPOSITION 3.2. *If C is a cycle of a chordal graph, then for every edge* (u, v) *of C there is a vertex w of C that is adjacent to both u and v.*

[1] It is to be understood in the sequel that for all vertices of an n-cycle or an n-sun addition of indices is done modulo n.

THEOREM 3.3. *If G is a chordal graph and G^k has a hole $H=(x_1, \cdots, x_n, x_1)$ of length $n \geqq 4$, then k is even and $d(x_i, x_{i+1})=k$ for $1 \leqq i \leqq n$. Moreover, if for $1 \leqq i \leqq n$, p_i is any shortest path from x_i to x_{i+1} containing z_i such that $d(x_i, z_i)=d(x_{i+1}, z_i)=k/2$, then $(z_1, \cdots, z_n, z_1)$ is a cycle of length n in G. Also, there are $y_1, \cdots, y_n$ such that $\{y_1, \cdots, y_n, z_1, \cdots, z_n\}$ induces an n-sun with the y's as outer vertices and the z's as inner vertices such that $d(x_i, y_i)=k/2-1$, $i=1, \cdots, n$.*

THEOREM 3.4. *If G is a chordal graph and $d(x, y)=k$ is finite and $0 \leqq m \leqq k$, then* Bet (x, m, y) *is a clique.*

THEOREM 3.5. *In a chordal graph G, if C is a clique and $x \notin C$ is such that $d(x, y)=k$ for all $y \in C$, then $\cap_{y \in C}$* Bet $(y, 1, x)$ *is not empty.*

THEOREM 3.6. *If G is a chordal graph and S is a maximal clique in G^k, then S induces a connected subgraph H of G such that $d_G(x, y)=d_H(x, y)$ for all $x, y \in S$.*

THEOREM 3.7. *If G is a* 3SF*-chordal graph and k is even, then C is a clique in G^k if and only if there is some z such that $d_G(z, y) \leqq k/2$ for all $y \in C$.*

Outline of the proof of Theorem 3.1. Since G is SF-chordal, all induced subgraphs H of G are SF-chordal.

(F1) Since H is SF-chordal, by Theorem 3.3 H^k is chordal for all $k \geqq 1$ and hence $\alpha(H^k: \mathscr{C})=\theta(H^k: \mathscr{C})$.

(F2) This follows from Theorem 3.7 and the fact that a vertex set C of H is covered by a partial graph T_k that is a tree of diameter $\leqq k$ if and only if there is some z such that $d_H(z, y) \leqq k/2$ for all $y \in C$. □

The converse of Theorem 3.1 is false. Complete even suns are $\mathscr{T}_k$-perfect for all even k. In this section, we will use a result on balanced matrices (see Chang [1982]) to prove that OSF-chordal graphs are $\mathscr{T}_2$-perfect. We also prove that if the strong perfect graph conjecture is true, then OSF-chordal graphs are $\mathscr{T}_k$-perfect for all even k.

Suppose A is an $m \times n$ matrix all of whose coefficients are 0 or 1. Consider the following two integer linear programming problems (ILP) and their linear programming relaxations.

$$P(A): \qquad \theta(A)=\min\{x1: xA \geqq 1, x \geqq 0 \text{ and integer}\},$$

$$D(A): \qquad \alpha(A)=\max\{1y: Ay \leqq 1, y \geqq 0 \text{ and integer}\},$$

$$\bar{P}(A): \qquad \bar{\theta}(A)=\min\{x1: xA \geqq 1, x \geqq 0\},$$

$$\bar{D}(A): \qquad \bar{\alpha}(A)=\max\{1y: Ay \leqq 1, y \geqq 0\},$$

where 1 stands for both m- and n-vectors all of whose coefficients equal 1. We assume that A has no zero column to guarantee that these four problems are feasible and have finite optimal solutions. From relaxation and linear programming duality, we obtain the following weak duality for the ILP's:

$$\alpha(A) \leqq \bar{\alpha}(A)=\bar{\theta}(A) \leqq \theta(A). \tag{3.1}$$

The $\mathscr{T}_2$-covering problem is $P(M)$, where M is a symmetric $|V| \times |V|$ matrix with $m_{ij}=1$ if $d(i, j) \leqq 1$ and $m_{ij}=0$ otherwise. The $\mathscr{T}_2$-packing problem is $D(M)$. We call M the *closed neighborhood matrix* of G since each row of M is the incidence vector of a closed neighborhood of a vertex.

A 0–1 matrix A is said to be *balanced* if it does not have a $p \times p$ submatrix all of whose row and column sums equal two, where p is an odd integer.

THEOREM 3.8 (Berge [1972], Fulkerson et al. [1974]). *If A is a balanced matrix, then $\alpha(A)=\theta(A)$. Moreover, all of the extreme points of $\{y: Ay \leqq 1, y \geqq 0\}$ are integral.*

OSF-chordal graphs can be characterized in terms of balanced matrices.

THEOREM 3.9 (Chang [1982]). *G is* OSF-*chordal if and only if its closed neighborhood matrix M is balanced.*

COROLLARY 3.10. *If G is* OSF-*chordal, then G is $\mathcal{T}_2$-perfect.*

Proof. Since G is OSF-chordal, all of its induced subgraphs H are OSF-chordal. If M is the closed neighborhood matrix of H, then $\alpha(H\colon \mathcal{T}_2) = \alpha(M)$ and $\theta(H\colon \mathcal{T}_2) = \theta(M)$. By Theorems 3.8 and 3.9, we have $\alpha(M) = \theta(M)$. Hence $\alpha(H\colon \mathcal{T}_2) = \theta(H\colon \mathcal{T}_2)$ and so G is $\mathcal{T}_2$-perfect. □

COROLLARY 3.11. *If G is* OSF-*chordal, then G^2 is perfect.*

Proof. If G is OSF-chordal, then it is 3SF-chordal. Consider the matrix $M_C(G^2)$ whose rows are incidence vectors of all maximal cliques of G^2. Theorem 3.7 implies that each row of $M_C(G^2)$ is a row of the closed neighborhood matrix M of G, i.e. $M_C(G^2)$ is a submatrix of M. By Theorem 3.9, M is balanced and so $M_C(G^2)$ is balanced. The perfection of G^2 then follows from Theorem 3.8, and the fact that G is perfect if and only if all extreme points of $\{y\colon M_C(G)y \leqq 1, y \geqq 0\}$ are integral (see Golumbic [1980, Thm. 3.14]. □

We now turn to the question of giving necessary conditions for $\mathcal{T}_k$-perfection for all even k.

THEOREM 3.12. *If $G = (V, E)$ is* OSF-*chordal and k is even, then G^k has no odd holes or odd anti-holes of length at least five.*

Proof. Theorem 3.3 implies that G^k has no odd hole of length at least five and hence has no 5-anti-hole, since a 5-anti-hole is also a 5-hole. Suppose G^K has an odd anti-hole $\bar{H}$ of size $n = 2m+1 \geqq 7$. In particular, suppose $H = (x_1, \cdots, x_n, x_1)$ is an n-hole in the complement of G^k.

We say that i and j are *consecutive* if $j = i+1$ or $j = i-1$.

CLAIM 1. *$d_G(x_i, x_j) = k+1$ if i and j are consecutive, else $d_G(x_i, x_j) = k$.*

Proof of Claim 1. Suppose i and j are consecutive and assume $j = i+1$. Then $(x_i, x_{i+3}, x_{i+1}, x_{i+4}, x_i)$ is a 4-hole in G^k, see Fig. 3.2. Hence by Theorem 3.3, there exists y_i and y_j such that $d_G(x_i, y_i) = d(x_j, y_j) = k/2-1$ and $d_G(y_i, y_j) \leqq 3$, which implies $d_G(x_i, x_j) \leqq k+1$. But $d_G(x_i, x_j) > k$ since $(x_i, x_j) \notin E^k$, so that $d_G(x_i, x_j) = k+1$.

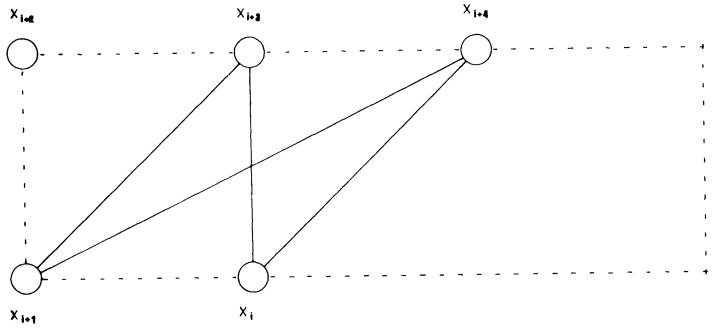


FIG. 3.2. *A 4-hole $(x_i, x_{i+3}, x_{i+1}, x_{i+4}, x_i)$ in G^k.*

If i and j are not consecutive, then either $i-1$ is not consecutive to $j+1$ and hence $(x_i, x_{j+1}, x_{i-1}, x_j, x_i)$ is a 4-hole in G^k, or else $i+1$ is not consecutive to $j-1$ and hence $(x_i, x_{j-1}, x_{i+1}, x_j, x_i)$ is a 4-hole in G^k, see Fig. 3.3. In either case we have a hole of G^k that contains (x_i, x_j) so that Theorem 3.3 implies $d_G(x_i, x_j) = k$. so Claim 1 holds.

For $i = 1, \cdots, n$, consider the set $C_i = \{x_{i+2s}\colon s = 0, 1, \cdots, m-1\}$.

CLAIM 2. *For each i there exists a vertex z_i such that if $x_j \in C_i$, then $d_G(x_j, z_i) = k/2$, else $d_G(x_j, z_i) > k/2$.*

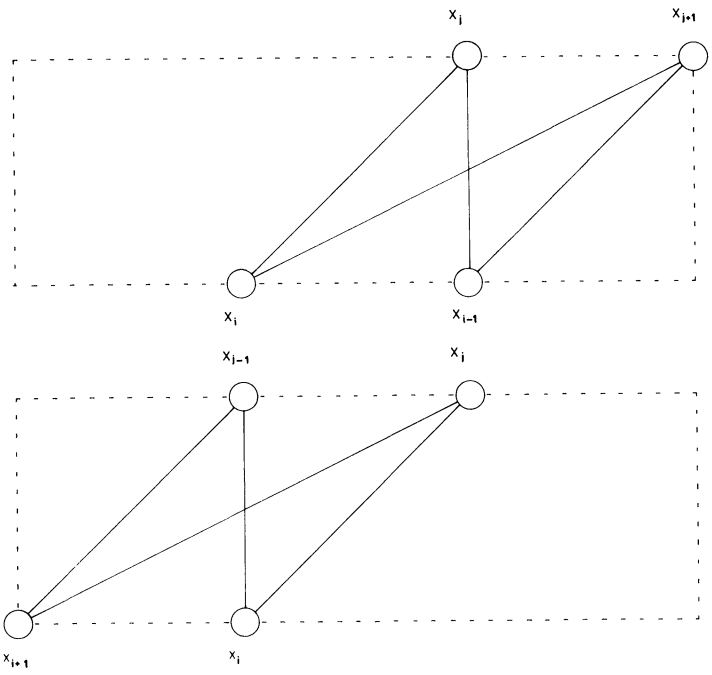


FIG. 3.3. *A 4-hole* $(x_i, x_{j+1}, x_{i-1}, x_j, x_i)$ *or* $(x_i, x_{j-1}, x_{i+1}, x_j, x_i)$ *in* G^k.

Proof of Claim 2. By Claim 1, each C_i is a clique in G^k. Since G is 3SF-chordal, Theorem 3.7 implies that for each C_i there is a z_i such that $d_G(x_j, z_i) \leqq k/2$ for every $x_j \in C_i$. Then since Claim 1 implies that $d_G(x_p, x_q) = k$ for $x_p, x_q \in C_i$, we obtain $d_G(x_j, z_i) = k/2$ for all $x_j \in C_i$. If $x_j \notin C_i$, then there is some $0 \leqq s \leqq m-1$ such that j and $i+2s$ are consecutive. So $d_G(x_j, z_i) > k/2$ follows from the fact that $d_G(x_j, x_{i+2s}) > k$ and $d_G(z_i, x_{i+2s}) = k/2$. So Claim 2 holds.

CLAIM 3. *For* $i = 1, \cdots, n$, *let* z_i *be defined as in Claim* 2. *Then* $(z_i, z_j) \in E$ *for all* $i \neq j$.

Proof of Claim 3. Case 1. Suppose i and j are consecutive, say $j = i+1$. Consider the 4-hole $H = (x_i, x_{i+3}, x_{i+1}, x_{i+4}, x_i)$ of G^k, see Fig. 3.2. $d_G(x_i, z_i) = d_G(x_{i+4}, z_i) = k/2$ by Claim 2 and the fact that $x_i, x_{i+4} \in C_i$; $d_G(x_i, x_{i+4}) = k$ by Claim 1. Thus z_i is the middle vertex of some shortest path p_1 from x_i to x_{i+4}. Similarly, z_j is the middle vertex of some shortest path p_2 from x_{i+1} to x_{i+3}. By Theorem 3.3, G has a 4-sun in which z_i and z_j are two inner vertices. Since G is 3-sun-free, the 4-sun is a complete 4-sun and hence $(z_i, z_j) \in E$.

Case 2. Suppose i and j are not consecutive. Since n is odd, $j = i+2s$ or $i = j+2s$ for some $1 \leqq s \leqq m-1$. Suppose, without loss of generality, that $j = i+2s$. In this case, $x_j \in C_i \cap C_j$, $x_{j-2} \in C_i \backslash C_j$, and $x_{j-3} \in C_j \backslash C_i$. Consider the 4-hole $H = (x_j, x_{j-2}, x_{j+1}, x_{j-3}, x_j)$ in G^k, see Fig. 3.4. Again, z_i is the middle vertex of some shortest path p_1 from x_j to x_{j-2} and z_j is the middle vertex of some shortest path p_2 from x_j to x_{j-3}. So Theorem 3.3 implies $(z_i, z_j) \in E$.

From Claim 2, Claim 3, and Theorem 3.5, there is a y_j such that $d(x_j, y_j) = k/2-1$ and $(y_j, z_{j-2s}) \in E$ for $s = 0, 1, \cdots, m-1$. However, for any other z_i, Claim 2 implies $d_G(x_j, z_i) > k/2$ so that $(y_j, z_i) \notin E$. Thus we have Claim 4.

CLAIM 4. $(y_j, z_i) \in E$ *if and only if* $j - i = 2s$ *for some* $0 \leqq s \leqq m-1$.

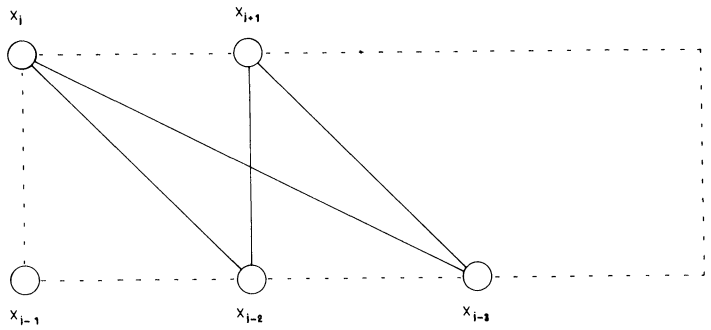


FIG. 3.4. *A 4-hole* $(x_j, x_{j-2}, x_{j+1}, x_{j-3}, x_j)$ *in* G^k.

Now consider $W = \{y_1, y_{n-2}, y_n, y_2, y_{n-1}, z_1, z_3, z_n, z_2, z_{n-1}\}$, which induces a chordal graph that satisfies the following three properties (see Fig. 3.5):

(1) $\{z_1, z_3, z_n, z_2, z_{n-1}\}$ is a clique by Claim 3.

(2) $(y_1, z_1, y_{n-2}, z_3, y_n, z_n, y_2, z_2, y_{n-1}, z_{n-1}, y_1\}$ is a cycle and, by Claim 4, each y in the cycle is adjacent only to exactly two z's.

(3) $\{y_1, y_{n-2}, y_n, y_2, y_{n-1}\}$ is a stable set. This follows from (1), (2), and the fact that G is chordal. For example, suppose $(y_1, y_{n-2}) \in E$; then (1) and (2) imply that $(y_1, y_{n-2}, z_3, z_{n-1}, y_1)$ is a 4-hole in G, which is a contradiction.

From (1)–(3), W induces a complete 5-sun, which is a contradiction since G is odd-sun-free. Hence the theorem is true. □

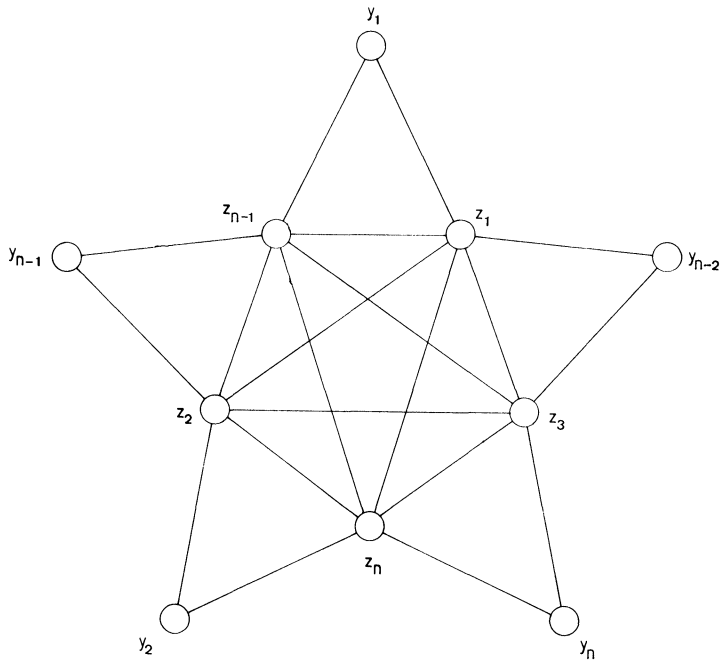


FIG. 3.5

THEOREM 3.13. *If G is* $\mathcal{T}_k$*-perfect for all even k, then G is* OSF*-chordal. Moreover, if the strong perfect graph conjecture is true, then the converse is also true.*

Proof. ($\Rightarrow$). G does not have an n-hole H_n with $n \geqq 4$ as an induced subgraph since $\alpha(H_n: \mathcal{T}_k) = 1 < 2 = \theta(H_n: \mathcal{T}_k)$, where $k = n-2$ if n is even and $k = n-3$ if n is

odd. Hence G is chordal. G does not have an odd sun as an induced subgraph since $\alpha(\text{odd-}n\text{-sun: } \mathcal{T}_2) = (n-1)/2 < (n+1)/2 = \theta(\text{odd-}n\text{-sun: } \mathcal{T}_2)$.

($\Leftarrow$) Induced sugraphs of OSF-chordal graphs are OSF-chordal.

(F1) By Theorem 3.12, if the strong perfect graph conjecture is true, then even powers of OSF-chordal graphs are perfect.

(F2) See the statement in the outline of the proof of Theorem 3.1. □

We close this section with the conjecture that Theorem 3.13 is true independent of the strong perfect graph conjecture.

4. 3-sun-3-anti-sun-free chordal graphs. In this section, we characterize graphs that are $\mathcal{T}_k$-perfect for all $k \geqq 2$.

PROPOSITION 4.1. *Suppose $a_1, a_2, a_3, b_1, b_2, b_3$ are six vertices in a chordal graph $G = (V, E)$ and $\{a_1, a_2, a_3\}$ forms a triangle.*

(A) *If $(a_i, b_j) \in E \Leftrightarrow i = j$, then $\{b_1, b_2, b_3\}$ is a stable set and hence the six vertices induce a 3-anti-sun as in Fig.* 4.1(a).

(B) *If $(a_i, b_j) \in E \Leftrightarrow i \neq j$, then $\{b_1, b_2, b_3\}$ is a stable set and hence the six vertices induce a 3-sun as in Fig.* 4.1(b).

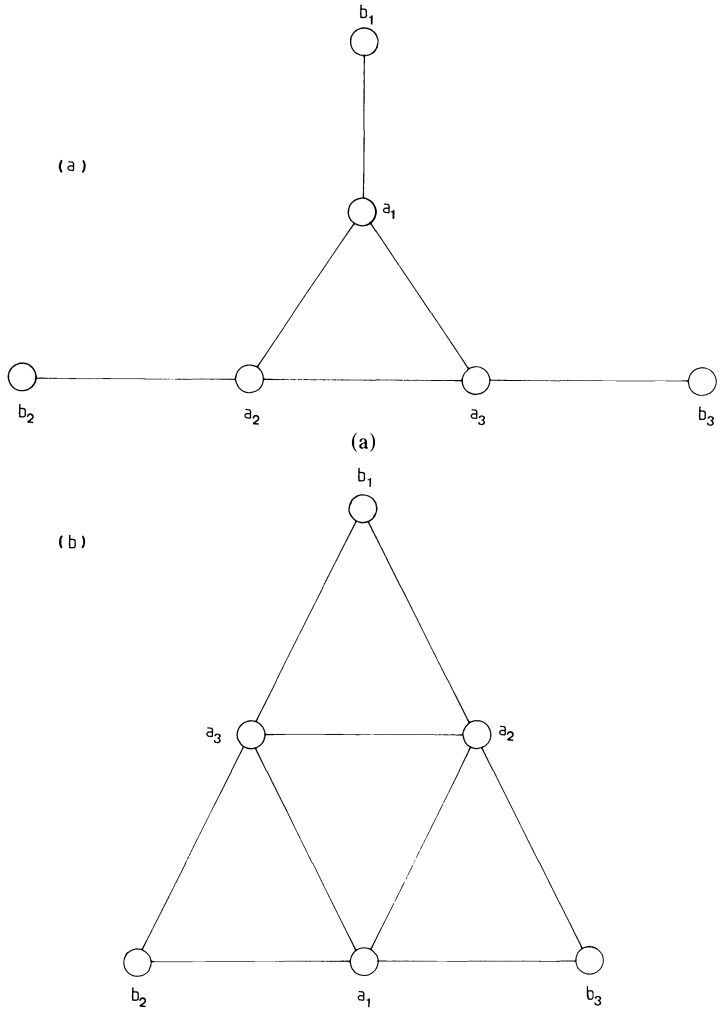


FIG. 4.1

We now characterize 3-anti-sun-free chordal graphs by a minimum bi-radius property. As noted in the introduction, $\text{br}(H) \geqq \lfloor d(H)/2 \rfloor$ for any connected graph H. We say that G has the *minimum bi-radius* property if

(F3) $\text{br}(H) = \lfloor d(H)/2 \rfloor$ for all connected induced subgraphs H of G such that $d(H) \geqq 2$.

Note that if $d(H) = 1$, then H is a complete graph. So $\text{br}(H) = \lfloor d(H)/2 \rfloor = 0$ if H is an edge and $\text{br}(H) = 1 > \lfloor d(H)/2 \rfloor$ otherwise.

THEOREM 4.2. *For any chordal graph G, the following statements are equivalent*;

(1) (F3) *is true.*

(2) (F3) *is true for all H such that $d(H)$ is odd.*

(3) *G is 3-anti-sun-free.*

Proof. (1)$\Rightarrow$(2) is obvious.

(2)$\Rightarrow$(3) holds since $d(\text{3-anti-sun}) = 3$ and $\text{br}(\text{3-anti-sun}) = 2$.

(3)$\Rightarrow$(1). Suppose this is not the case; then G has a connected induced subgraph H with $d(H) \geqq 2$ such that $\lfloor d(H)/2 \rfloor + 1 \leqq \text{br}(H)$ or, equivalently,

$$d(H) \leqq 2\,\text{br}(H) - 1. \tag{4.1}$$

The theorem will be proved by obtaining a contradiction to (4.1).

Note that $2\,\text{br}(H) - 1 \geqq d(H) \geqq 2$ implies that $\text{br}(H) \geqq \lceil 1.5 \rceil = 2$.

For any bi-center (u, v), define $S(u, v) = \{z: d(\{u, v\}, z) = \text{br}(H)\}$. Note that $S(u, v) \neq \varnothing$. Now choose a bi-center (x, y) such that $|S(x, y)|$ is as small as possible.

Let $V(x) = \{z: d(z, x) < d(z, y)\}$ and $V(y) = \{z: d(z, y) < d(z, x)\}$.

CLAIM 1. *$V(x) \neq \varnothing$ and $V(y) \neq \varnothing$.*

Proof of Claim 1. If $V(x) = \varnothing$ then $d(z, y) \leqq d(z, x)$ and hence $d(z, \{x, y\}) = d(z, y)$ for all $z \in V$. Choose $z' \in \text{Nbd}(y, \text{br}(H))$ and $x' \in \text{Bet}(y, 1, z')$. Note that $d(z, \{x, y\}) = d(z, y) \geqq d(z, \{x', y\})$ for all $z \in V$, so (x', y) is a bi-center such that $S(x', y) \subseteq S(x, y)$. But $z' \in S(x, y) \backslash S(x', y)$ and so $|S(x', y)| < |S(x, y)|$, which is a contradiction. So $V(x) \neq \varnothing$. Similarly, $V(y) \neq \varnothing$.

CLAIM 2. *$d(z, x) = d(z, y) < \text{br}(H)$ for all $z \in V \backslash (V(x) \cup V(y))$.*

Proof of Claim 2. Suppose there is z such that $d(z, x) = d(z, y) = \text{br}(H)$. Then, by Theorem 3.5, there is $u \in \text{Bet}(x, 1, z) \cap \text{Bet}(y, 1, z)$. Since $\text{br}(H) \geqq 2$, we can choose $u' \in \text{Bet}(u, 1, z)$. Then u' is not adjacent to x or y, see Fig. 4.2. Choose an x' adjacent to x but not adjacent to u or y. Suppose no such x' exists; then $d(x, v) \geqq d(\{u, y\}, v)$ for all $v \neq x$ and $d(x, z) > d(\{u, y\}, z)$. Hence (u, y) is a bi-center such that $S(u, y) \subseteq S(x, y)$. But $z' \in S(x, y) \backslash S(u, y)$ and so $|S(u, y)| < |S(x, y)|$, which is a contradiction. Similarly, there is a y' adjacent to y but not to u or x. Then, by Proposition 4.1(A), $\{x, y, u, x'y'u'\}$ induces a 3-anti-sun, which is a contradiction. This proves Claim 2.

Now let $d_1 = \max_{z \in V(x)} d(z, x)$ and $d_2 = \max_{z \in V(y)} d(z, y)$. Without loss of generality, assume $d_1 \leqq d_2$.

CLAIM 3. $\text{br}(H) - 1 \leqq d_1 \leqq d_2 = \text{br}(H)$.

Proof of Claim 3. By Claim 2, we know that $\text{br}(H) = \max\{d_1, d_2\} = d_2$. So we only have to prove that $\text{br}(H) - 1 \leqq d_1$. Suppose $d_1 \leqq \text{br}(H) - 2$. Since $(x, y) \in E$, $d(z, y) \leqq \text{br}(H) - 1$ for any $z \in V(x)$. Choose $z' \in \text{Nbd}(y, \text{br}(H))$ and $x' \in \text{Bet}(y, 1, z')$. Then (x', y) is a bi-center such that $S(x', y) \subseteq S(x, y)$ and $z' \in S(x, y) \backslash S(x', y)$. Thus $|S(x', y)| < |S(x, y)|$, which is a contradiction. This proves Claim 3.

CLAIM 4. *$d(H) = 2\,\text{br}(H) - 1$ and $d_1 = \text{br}(H) - 1$. Moreover, for any $z_1 \in V^*(x) = V(x) \cap \text{Nbd}(x, d_1)$ and $z_2 \in V^*(y) = V(y) \cap \text{Nbd}(y, d_2)$, $d(z_1, z_2) = 2\,\text{br}(H) - 1$ and there is a vertex v adjacent to x and y such that $v \in \text{Bet}(z_1, d_1, z_2)$.*

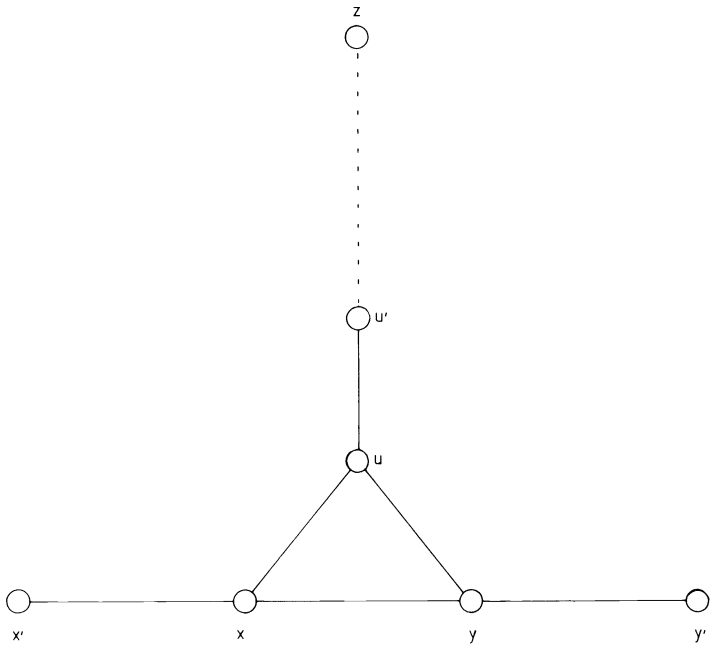


FIG. 4.2

Proof of Claim 4. Choose shortest paths $p_{z_1x}, p_{z_2y}, p_{z_1z_2}$ from z_1 to x, z_2 to y, and z_1 to z_2 respectively as in Fig. 4.3. Note that it is impossible that $x = w_1$ or $y = w_2$, otherwise $d(H) \geqq d(z_1, z_2) = d_1 + 1 + d_2 \geqq 2 \text{ br } (H)$, which contradicts (4.1). Consider the cycle $(w_1, \cdots, w_2, \cdots, y, x, \cdots, w_1)$. By Proposition 3.2, there is a vertex v adjacent to both x and y. Note that v is not between x and w_1 (similarly, not between y and w_2) otherwise $d(z_1, y) \leqq d_1 = d(z_1, x)$, which contradicts the assumption that $z_1 \in V(x)$. Also

$$d(z_1, v) \geqq d(z_1, x), \tag{4.2}$$

since $d(z_1, v) < d(z_1, x)$ implies that $d(z_1, y) \leqq d(z_1, x)$, which contradicts the assumption that $z_1 \in V(x)$. Similarly, we have

$$d(z_2, v) \geqq d(z_2, y). \tag{4.3}$$

From (4.1), (4.2), (4.3) and Claim 3, we have

$$\begin{aligned} 2 \text{ br } (H) - 1 \geqq d(H) &\geqq d(z_1, z_2) = d(z_1, v) + d(z_2, v) \\ &\geqq d(z_1, x) + d(z_2, y) = d_1 + d_2 \geqq 2 \text{ br } (H) - 1. \end{aligned}$$

Thus $d(z_1, z_2) = d(H) = 2 \text{ br } (H) - 1$ and $d(z_1, x) = d(z_1, v) = d_1 = \text{br } (H) - 1$ and $d(z_2, y) = d_2$. So Claim 4 holds.

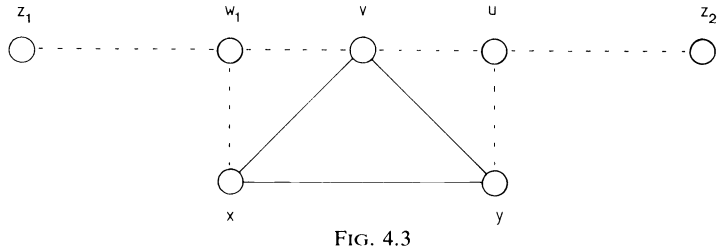


FIG. 4.3

Now choose vertices $z_1 \in V^*(x)$, $z_2 \in V^*(y)$, and a corresponding $v \in$ Bet (z_1, d_1, z_2) that is adjacent to both x and y as in Claim 4. Without loss of generality, we can assume that $S(v) = \{w \in V^*(x)\colon d(v, w) = \text{br}(H)\}$ has as few vertices as possible. Note that $d(v, z) \leqq d(x, z) + 1 = d_1 + 1 = \text{br}(H)$ for all $z \in V^*(x)$.

CLAIM 5. $S(v) \neq \varnothing$.

Proof of Claim 5. Claim 4 implies $d(z_1, x) = d(z_1, v) = d_1$. Since x is adjacent to v, by Theorem 3.5, there is a vertex $v' \in$ Bet $(x, 1, z_1) \cap$ Bet $(v, 1, z_1)$. Similarly, there is a $u \in$ Bet $(y, 1, z_2) \cap$ Bet $(v, 1, z_2)$ and a $u' \in$ Bet $(u, 1, z_2)$ as in Fig. 4.4. Note that u' is not adjacent to y or v; v' is not adjacent to u; and v' is not adjacent to y otherwise $d(z_1, y) \leqq d_1 = d(z_1, x)$, which contradicts the fact that $z_1 \in V^*(x)$. Suppose there is a vertex y' adjacent to y but not u or v. Then Proposition 4.1(A) implies that $\{u, v, y, u', v', y'\}$ induces a 3-anti-sun, which is a contradiction. So every z adjacent to y is either adjacent to u or v. This implies that $d(z, y) \geqq d(z, \{u, v\})$ for every $z \in V$. This fact and $d(v, z) \leqq \text{br}(H)$ for all $z \in V^*(x)$ imply that (u, v) is a bi-center. Suppose $S(v) = \varnothing$, then $S(u, v) \subseteq S(x, y)$. But $z_2 \in S(x, y) \backslash S(u, v)$, so $|S(u, v)| < |S(x, y)|$. This contradicts the choice of (x, y), so Claim 5 holds.

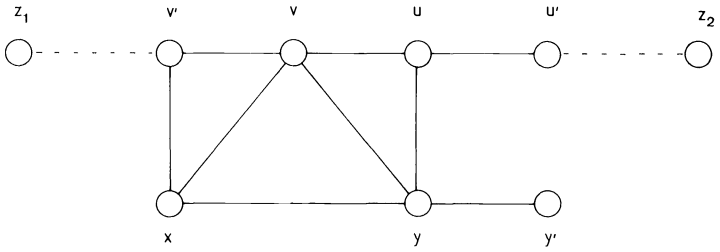


FIG. 4.4

Now fix z_1, z_2 and v as in the proof of Claim 5. By Claim 5, there is a $z \in S(v)$. By the definition of $S(v)$, $z \in V^*(X)$. By Claim 4, there is $w \in$ Bet (z, d_1, z_2) that is adjacent to both x and y. Since $d(z_2, v) = \text{br}(H)$, $d(v, x) = 1$ and $d(z_2, x) > \text{br}(H)$, we have $v \in$ Bet $(x, 1, z_2)$. Similarly, $w \in$ Bet $(x, 1, z_2)$. By Theorem 3.4, v is adjacent to w. So $C = \{w, v, y\}$ is a clique such that $d(z_2, w) = d(z_2, v) = d(z_2, y) = \text{br}(H)$. By Theorem 3.5, there is a vertex $u \in \bigcap_{a \in C}$ Bet $(a, 1, z_2)$ and then there is $u' \in$ Bet $(u, 1, z_2)$ as in Fig. 4.5. By Theorem 3.5, we can choose $w' \in$ Bet $(x, 1, z) \cap$ Bet $(w, 1, z)$.

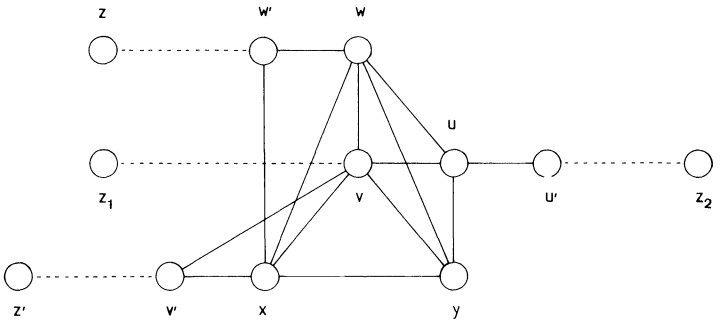


FIG. 4.5

Note that u' is not adjacent to v or w; w' is not adjacent to u; and w' is not adjacent to v otherwise $d(z, v) = \text{br}(H) - 1$ and then $z \notin S(v)$.

For any $z' \in S(w)$, i.e. $z' \in V^*(x)$ and $d(z', w) = \text{br}(H)$, suppose $d(z', v) = \text{br}(H) - 1$. Then there is $v' \in \text{Bet}(v, 1, z') \cap \text{Bet}(x, 1, z')$ as in Fig. 4.5. For the same reasons that w' is not adjacent to u or v, v' is not adjacent to u or w. Then, by Proposition 4.1(A), $\{u, v, w, u', v', w'\}$ induces a 3-anti-sun, which is a contradiction. So $d(z', v) = \text{br}(H)$ and $z' \in S(v)$, which proves that $S(w) \subseteq S(v)$. But $z \in S(v) \backslash S(w)$, so $|S(w)| < |S(v)|$. This contradicts the choice of v, and the proof is complete. □

COROLLARY 4.3. *If G is* 3ASF-*chordal and k is an odd integer* $\geqq 3$, *then C is a clique in G^k if and only if there exists an edge (x, y) such that $d_G(\{x, y\}, z) \leqq (k-1)/2$ for all $z \in C$. Hence $\theta(G: \mathcal{T}_k) = \theta(G^k: \mathcal{C})$.*

Proof. Suppose C is a clique in G^k. Without loss of generality we can assume it is maximal. By Theorem 3.6, C induces a connected subgraph H such that $d_H(u, v) = d_G(u, v)$ for all $u, v \in C$. Hence $d(H) \leqq k$. Since G is 3ASF-chordal, by Theorem 4.2, $\text{br}(H) = \lfloor d(H)/2 \rfloor \leqq (k-1)/2$, i.e. there is a bi-center (x, y) of H such that $d_G(\{x, y\}, z) = d_H(\{x, y\}, z) \leqq (k-1)/2$ for all $z \in C$.

The converse is obvious.

$\theta(G: \mathcal{T}_k) = \theta(G^k: \mathcal{C})$ then follows from the fact that C is covered by a partial graph that is a tree of diameter $\leqq k$ if and only if there is $(x, y) \in E$ such that $d(\{x, y\}, z) \leqq (k-1)/2$ for all $z \in C$. □

LEMMA 4.4. *Every n-sun with $n \geqq 5$ has a 3-anti-sun as a subgraph.*

Proof. Suppose the y's are outer vertices and the z's are inner vertices of the n-sun. By Proposition 3.2, the cycle $(z_1, z_2, \cdots, z_n, z_1)$, has a vertex z_j adjacent to both z_2 and z_3. If $j \notin \{1, 4\}$, then Proposition 4.1(A) implies that $\{z_2, z_3, z_j, y_2, y_4, y_j\}$ induces a 3-anti-sun. Now suppose, without loss of generality, that $j = 4$, see Fig. 4.6. Then in the cycle $(z_1, z_2, z_4, z_5, \cdots, z_n, z_1)$, there is a vertex z_j adjacent to both z_2 and z_4. Choose $i = j+1$ if $j = 5$ and $i = j$ otherwise, then $\{z_2, z_4, z_j, y_3, y_4, y_i\}$ induces a 3-anti-sun. □

COROLLARY 4.5. *If G is* 3S3ASF-*chordal, then it is* OSF-*chordal and hence $\mathcal{T}_2$-perfect.*

THEOREM 4.6. *If G is* 3ASF-*chordal, then G^k is chordal for all integers $k \geqq 3$.*

Proof. Suppose G^k has a hole $H = (x_1, \cdots, x_n, x_1)$ of length $n \geqq 4$. By Theorem 3.3, k is even and $\{y_1, \cdots, y_n, z_1, \cdots, z_n\}$ induces an n-sun. Choose $y \in \text{Bet}(y_1, 1, x_1)$. Theorem 3.3 implies that y is not adjacent to z_1 or z_n. Hence $\{y_1, z_1, z_n, y, y_2, y_n\}$ induces a 3-anti-sun. This is a contradiction, so G^k is chordal. □

Note that Theorem 4.6 is false if $k = 2$, e.g. a complete 4-sun satisfies the hypothesis of the theorem but its square is not chordal.

THEOREM 4.7. *The following statements are equivalent for all graphs G:*

(1) *G is $\mathcal{T}_k$-perfect for all integers $k \geqq 2$.*

(2) *G is $\mathcal{T}_{c_k}$-perfect for all integers $k \geqq 1$, where $c_1 = 2$, $c_2 = 3$ and $c_{k+1} = \prod_{j=1}^{k} (c_j + 1) - 2$ for $k \geqq 2$.*

(3) *G is* 3S3ASF-*chordal.*

Proof. (1)⇒(2) is clear.

(2)⇒(3). G is 3-sun-free since $\alpha(3\text{-sun}: \mathcal{T}_2) = 1 < 2 = \theta(3\text{-sun}: \mathcal{T}_2)$ and 3-anti-sun-free since $\alpha(3\text{-anti-sun}: \mathcal{T}_3) = 1 < 2 = \theta(3\text{-anti-sun}: \mathcal{T}_3)$. The proof of chordality is similar to the proof of (2)⇒(3) of Theorem 2.2.

(3)⇒(1). Because of Corollary 4.5, we only have to prove $\mathcal{T}_k$-perfection for $k \geqq 3$. Since G is 3S3ASF-chordal, its induced subgraphs H are 3S3ASF-chordal.

(F1) Theorem 4.6 implies that H^k is perfect.

(F2) $\theta(H^k: \mathcal{C}) = \theta(H: \mathcal{T}_k)$ follows from the statements in the outline of the proof of Theorem 3.1 when k is even, and from Corollary 4.3 when k is odd. □

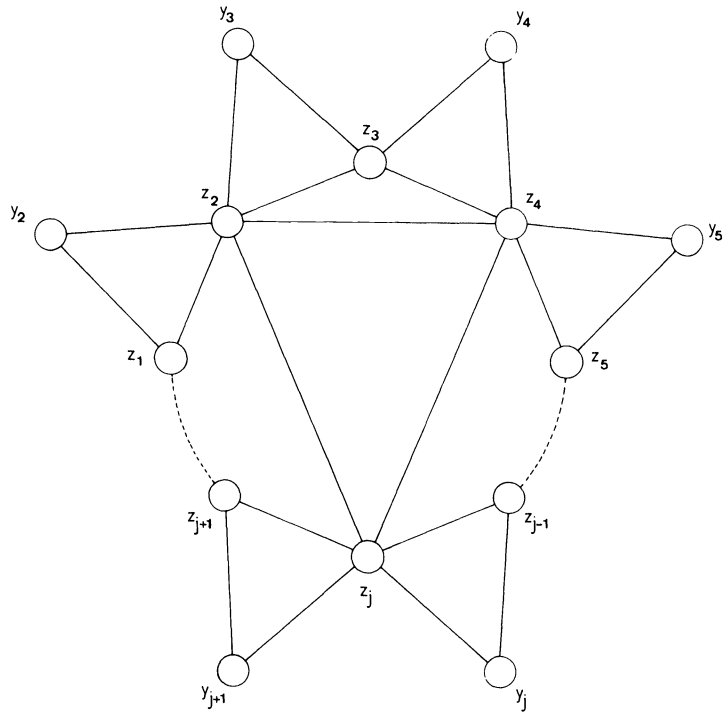


FIG. 4.6

5. Nearly chordal graphs. If G is OSF-chordal, Corollary 3.11 states that G^2 is perfect. Hence the stable set problem ($\mathscr{C}$-packing) and clique covering problem ($\mathscr{C}$-covering) on G^2, as well as the dominating set problem ($\mathscr{T}_2$-covering) on G can be solved in polynomial time by an ellipsoid algorithm (Grötschel, Lovász and Schrijver [1981]). However, no efficient combinatorial algorithm is known for these problems on OSF-chordal graphs. Higher powers of OSF-chordal graphs are not even known to be perfect, but as discussed in § 3, we strongly believe that they are.

The situation is much simpler for 3S3ASF-chordal graphs. If G is 3S3ASF-chordal, then G^k is perfect for all $k \geqq 1$ and indeed chordal for all $k \neq 2$. Hence the stable set and clique covering problem on G^k, $k \neq 2$, and the $\mathscr{T}_k$-packing and $\mathscr{T}_k$-covering problems, $k > 2$, on G can be solved by well-known combinatorial algorithms (see Golumbic [1980]). In this section, we consider $k = 2$.

A graph $G = (V, E)$ is called *nearly chordal* if the following three conditions hold:

(NC1) The maximum length of a hole in G is 4.

(NC2) If $(x_1, x_2, x_3, x_4, x_1)$ is a 4-hole and $v \neq x_i$ for $i = 1, 2, 3, 4$, then v is either not adjacent to any x_i or adjacent to at least three x_i's.

(NC3) If $(x_1, x_2, x_3, x_4, x_1)$ is a 4-hole, then $C_i = \{v \in V: v \text{ is adjacent to both } x_i \text{ and } x_{i+1}\} \cup \{x_i, x_{i+1}\}$ is a clique in G for $i = 1, 2, 3, 4$.

We will give an $O(|V| \cdot |E|)$ algorithm for solving the stable set and clique covering problems on nearly chordal graphs. We will also prove that if G is 3S3ASF-chordal, then G^2 is nearly chordal so the algorithm also solves the dominating set problem on G. It can be shown that the class of nearly chordal graphs is a subset of the perfectly orderable graphs introduced by Chvátal [1981].

PROPOSITION 5.1. *Chordal graphs are nearly chordal. Every induced subgraph of a nearly chordal graph is nearly chordal.*

PROPOSITION 5.2. *In a nearly chordal graph G, every cycle $C=(x_1, x_2, \cdots, x_n, x_1)$ of length $n \geq 5$ has at least $n-3$ chords.*

Proof. We will prove the proposition by induction on n. If $n=5$, then C cannot have only one chord, otherwise G has a 4-hole and a vertex adjacent to exactly two vertices of the 4-hole, which contradicts (NC2). Suppose the proposition holds for all $n' < n \geq 6$. By (NC1), C has a chord.

Case 1. C has a chord of the form (x_i, x_{i+2}). Consider the cycle $C_1 = (x_i, x_{i+2}, x_{i+3}, \cdots, x_{i-1}, x_i)$, of length $n-1 \geq 5$. By the induction hypothesis, C_1 has at least $n-4$ chords all of which are chords of C. So C has at least $n-3$ chords.

Case 2. C has no chord of the type considered in Case 1, but has a chord of the form (x_i, x_{i+3}). Then $C_1 = (x_i, x_{i+1}, x_{i+2}, x_{i+3}, x_i)$ is a 4-hole and $C_2 = (x_i, x_{i+3}, x_{i+4}, \cdots, x_{i-1}, x_i)$ is a cycle of length $n-2 \geq 4$. By the induction hypothesis, C_2 has a least $(n-2)-4 = n-6$ chords (this includes the case of $n-2=4$). Since x_{i-1} is adjacent to a vertex of C_1, (NC2) implies that either $(x_{i-1}, x_{i+1}) \in E$ or $(x_{i-1}, x_{i+2}) \in E$. Similarly, $(x_{i+4}, x_{i+1}) \in E$ or $(x_{i+4}, x_{i+2}) \in E$. So C has at least $1+(n-6)+2 = n-3$ chords.

Case 3. C has a chord that decomposes it into two cycles of length $n_1 \geq 5$ and $n_2 \geq 5$ such that $n_1 + n_2 = n+2$. By the induction hypothesis, these two cycles have at least $n_1 - 3$ and $n_2 - 3$ chords respectively. So C has at least $1+(n_1-3)+(n_2-3) = n-3$ chords. □

PROPOSITION 5.3. *Nearly chordal graphs are perfect.*

Proof. The result follows from Proposition 5.2 and the fact that if every odd cycle of length ≥ 5 of G has at least two chords, then G is perfect (Meyniel [1976]). The proposition also follows from Algorithm 5.2 given below. □

Burlet [1982] has recently given an algorithm for solving the clique covering and stable set problems on graphs with the property that every odd cycle of length ≥ 5 has at least two chords. Although this algorithm can, of course, solve these two problems on nearly chordal graphs, it is much more complicated than the algorithm to be given below, which is applicable only to nearly chordal graphs.

A trivial polynomial-time algorithm can be obtained by enumerating 4-holes and then applying any polynomial algorithm for chordal graphs. However, a more efficient algorithm is obtained by using a scheme based on Lexicographic Breadth First (LBF) search (Rose, Tarjan, and Leuker [1976]).

ALGORITHM 5.1. LBF search.
assign the label $\varnothing$ to each vertex;
for $i \leftarrow n$ **to** 1 **step** -1 **do**
    pick an unmarked vertex v with largest (lexicographic) label;
    assign the index i to v and mark v;
    for each unmarked vertex $w \in \mathrm{Nbd}(v)$ **do** add i to label (w);
    end.

Note that we do not actually need to calculate the labels, rather we keep the unmarked vertices in lexicographic order by using a queue. For details and the following property see Golumbic [1980].

PROPOSITION 5.4 (Golumbic [1980]). *Suppose $[v_1, v_2, \cdots, v_n]$ is an* LBF *ordering of the vertices of a graph. If $a < b < c$ and v_c is adjacent to v_a but not v_b, then there exists $d > c$ such that v_d is adjacent to v_b but not v_a.*

THEOREM 5.5. *Suppose $[v_1, \cdots, v_n]$ is an* LBF *ordering of the vertices of a nearly chordal graph $G=(V, E)$. If there are $i<j<k$ such that $(v_i, v_j) \in E$, $(v_i, v_k) \in E$ and*

$(v_j, v_k) \notin E$, then there are p and q such that $i<j<p<q$ and $(v_i, v_p, v_q, v_j, v_i)$ is a 4-hole in G.

Proof. We consider a sequence of integers $n_{-1} > n_0 < n_1 < \cdots$ defined as follows. Let $n_{-1} = j$, $n_0 = i$, $n_1 = j$, and then choose $n_2 = p$ as large as possible such that $(v_{n_0}, v_{n_2}) \in E$ but $(v_{n_{-1}}, v_{n_2}) \notin E$. Note that p exists and $k \leqq p$. Now by Proposition 5.4, we can choose $n_3 = q > p$ as large as possible such that $(v_{n_1}, v_{n_3}) \in E$ but $(v_{n_0}, v_{n_3}) \notin E$. If $(v_{n_2}, v_{n_3}) \in E$, then the proof is complete. Otherwise suppose $(v_{n_2}, v_{n_3}) \notin E$, we will get a contradiction by considering the following inductive procedure. Assume we are given the increasing sequence of integers $n_0 < n_1 < \cdots < n_m$ with the following properties:

(1) If $r > 0$, then $(v_{n_0}, v_{n_r}) \in E \Leftrightarrow r \leqq 2$.

(2) If $r, s > 0$, then $(v_{n_r}, v_{n_s}) \in E \Leftrightarrow |r-s| = 2$.

(3) For $r > 2$, n_r is as large as possible such that $(v_{n_{r-2}}, v_{n_r}) \in E$ but $(v_{n_{r-3}}, v_{n_r}) \notin E$.

The construction for $m = 3$ has been given above.

The vertices $v_{n_{m-2}}, v_{n_{m-1}}, v_{n_m}$ satisfy the hypothesis of Proposition 5.4 as v_a, v_b, and v_c respectively. Hence, choose $n_{m+1} > n_m$ as large as possible so that $v_{n_{m+1}}$ is adjacent to $v_{n_{m-1}}$ but not $v_{n_{m-2}}$. By the indunction hypothesis, we know that (1), (2), (3) are true for the case of $r, s \leqq m$. We complete the induction by showing that $v_{n_{m+1}}$ is not adjacent to v_{n_t} for $t = m, m-2, m-3, \cdots, 0$.

Suppose $v_{n_{m+1}}$ is adjacent to v_{n_m}. Consider the cycle $C = (v_{n_0}, v_{n_1}, v_{n_3}, \cdots, v_{n_m}, v_{n_{m+1}}, v_{n_{m-1}}, v_{n_{m-3}}, \cdots, v_{n_2}, v_{n_0})$ when m is odd, and $C = (v_{n_0}, v_{n_1}, v_{n_3}, \cdots, v_{n_{m-1}}, v_{n_{m+1}}, v_{n_m}, v_{n_{m-2}}, \cdots, v_{n_2}, v_{n_0})$ when m is even. C contains $m+2$ vertices and by the induction hypothesis and (1) and (2), every chord of C contains $v_{n_{m+1}}$. By Proposition 5.2, C has at least $m-1$ chords, which implies that $(v_{n_{m+1}}, v_{n_r}) \in E$ for $r = 0, 1, \cdots, m$. This is a contradiction since $v_{n_{m+1}}$ is not adjacent to $v_{n_{m-2}}$.

We will prove that $v_{n_{m+1}}$ is not adjacent to v_{n_t} for $t = m-2, m-3, \cdots, 0$ by backward induction on t. By choice, $v_{n_{m+1}}$ is not adjacent to $v_{n_{m-2}}$. Suppose $v_{n_{m+1}}$ is not adjacent to v_{n_t} but is adjacent to $v_{n_{t-1}}$. By Proposition 5.4, there is $n_y > n_{m+1}$ such that v_{n_y} is adjacent to v_{n_t} but not $v_{n_{t-1}}$. So n_y is larger than n_{t+2} and v_{n_y} is adjacent to v_{n_t} but not $v_{n_{t-1}}$, which is a contradiction to the choice of $v_{n_{t+2}}$.

Clearly, the inductive procedure continues indefinitely, but the graph is finite, which is a contradiction. So Theorem 5.5 holds. □

Remark. As in the proof, we can choose p and q in the following way: choose the maximum p such that $i<j<p$ and v_p is adjacent to v_i but not v_j; then choose the maximum $q>p$ such that v_q is adjacent to v_j but not v_i. We will use this procedure in Algorithm 5.2.

ALGORITHM 5.2.

input: A nearly chordal graph G with a LBF ordering $[v_1, \cdots, v_n]$ of its vertices.

output: A minimum clique covering C and a maximal vertex packing S.

method:

Step 0. $i \leftarrow 0$; $C \leftarrow \varnothing$; $S \leftarrow \varnothing$;
all vertices are unmarked.

Step 1. $i \leftarrow i+1$;
if $i \leqq n$ **then go to** Step 2;
else print C and S and STOP.

Step 2. **if** v_i is marked **then go to** Step 1;
else go to Step 3.

Step 3. $A \leftarrow \{v_k : k > i, (v_i, v_k) \in E$, and v_k unmarked$\} \cup \{v_i\}$;
if A is a clique **then go to** Step 4;
else go to Step 5.

Step 4. $C \leftarrow C \cup \{A\}$;
$S \leftarrow S \cup \{v_i\}$;
mark all vertices in A;
go to Step 1.

Step 5. Choose $v_j \in A$ and v_p, v_p as in the above remark such that $(v_i, v_j, v_q, v_p, v_i)$ is a 4-hole;
$A_1 \leftarrow \{z: z \text{ is adjacent to } v_i \text{ and } v_j\} \cup \{v_i, v_j\}$;
$A_2 \leftarrow \{z: z \text{ is adjacent to } v_p \text{ and } v_q\} \cup \{v_p, v_q\}$;
$C \leftarrow C \cup \{A_1, A_2\}$;
$S \leftarrow S \cup \{v_i, v_q\}$;
mark all vertices in A_1 and A_2;
go to Step 1.

THEOREM 5.6. *Algorithm* 5.2 *terminates in* $O(|V| \cdot |E|)$ *time, and gives a minimum cardinality clique covering* C *and a maximum cardinality vertex packing* S *such that* $|C| = |S|$.

Proof. It is easy to see that $|C| = |S|$ at each iteration and hence in the final output and also that the final set of cliques covers all of the vertices. A_1 and A_2 are cliques by condition (NC3). We only have to prove that S is a stable set at each iteration and hence in the final output. When v_i is put into S and A is a clique, then all neighbors of v_i are marked. When v_i and v_q are put into S and A is not a clique, then $(v_i, v_j, v_q, v_p, v_i)$ is a 4-hole, hence by (NC2) all $v \in \text{Nbd}(v_i) \cup \text{Nbd}(v_q)$ are adjacent to at least three vertices of the 4-hole and thus are in $A_1 \cup A_2$. Hence at the beginning of each iteration, a vertex is marked if and only if it is in S or adjacent to some vertex in S. So any unmarked vertex, in particular v_i, is not in S and not adjacent to any vertex in S. Thus when A is a clique, $S \cup \{v_i\}$ remains a stable set. When A is not a clique, v_i and v_j are unmarked. Suppose v_q is marked, then it is adjacent to some vertex x in S. By condition (NC2), x is then adjacent to at least two more vertices in the 4-hole. Hence either v_i or v_j is marked, which is a contradiction. So v_q is unmarked and hence $S \cup \{v_i, v_q\}$ remains stable.

Since the algorithm only examines the neighborhood of each vertex once, its running time is $O(|V| \cdot |E|)$. □

We conclude this paper by proving that if G is 3S3ASF-chordal, then G^2 is nearly chordal.

LEMMA 5.7.[2] *Suppose* $G = (V, E)$ *is* 3S3ASF-*chordal. Then the maximum length of a hole in* $G^2 = (V, E^2)$ *is four. Suppose* G^2 *has a* 4*-hole* $(x_1, x_2, x_3, x_4, x_1)$. *Then the following statements are true.*

(1) *There exist* z_1, z_2, z_3, z_4 *such that* $\{x_1, x_2, x_3, x_4, z_1, z_2, z_3, z_4\}$ *induces a complete* 4*-sun in* G *with the* x*'s as outer vertices and the* z*'s as inner vertices.*

(2) *Suppose* $v \neq x_i$ *for* $i = 1, 2, 3, 4$. *Then* v *is either not adjacent to any* x_i *in* G^2 *or else is adjacent to at least three* x*'s in* G^2.

(3) *Suppose* v *is adjacent to* x_i, x_{i+1}, x_{i+2} *but not* x_{i+3} *in* G^2. *Then* $v \neq z_i$ *for* $i = 1, 2, 3, 4$, *and* v *is adjacent to* z_i *and* z_{i+1} *but not* z_{i+2} *or* z_{i+3} *in* G.

(4) *If* $(v, x_i) \in E^2$ *for* $i = 1, 2, 2, 4$ *then* $d_G(v, z_i) \leqq 1$ *for* $i = 1, 2, 3, 4$.

Proof. Since G is 3-anti-sun-free, by Theorem 3.3 and Lemma 4.4, the maximum length of a hole in G^2 is four.

Suppose G^2 has a 4-hole $(x_1, x_2, x_3, x_4, x_1)$. By Theorem 3.3, there exist four vertices z_1, z_2, z_3, z_4 such that $\{x_1, x_2, x_3, x_4, z_1, z_2, z_3, z_4\}$ induces a 4-sun of G with

[2] Indices for the x_i's and z_i's are taken mod 4 in the statement and proof of the lemma.

the x's as outer vertices and the z's as inner vertices. (Note that $x_i = y_i$ for the case of $k = 2$ in Theorem 3.3.) Since G is 3-sun-free, the 4-sun is complete. This proves (1).

We will prove (2), (3), and (4) by the following steps.

Step 1. Suppose $v = z_j$ for some j. Then v is adjacent to all of the x's in G^2 and $d_G(v, z_i) \leqq 1$ for all i. So (2)–(4) are true. From now on, we assume that v is not a vertex of the complete 4-sun.

Step 2. If $(v, x_j) \in E^2$ for some j, then $(v, z_k) \in E$ for some k.

Proof of Step 2. Either $d_G(v, x_j) = 1$ or $d_G(v, x_j) = 2$.

Suppose $d_G(v, x_j) = 1$ and consider the six vertices $x_j, z_j, z_{j+3}, v, x_{j+1}, x_{j+3}$ (view them as a's and b's as in Fig. 5.1). Since $(a_i, b_1) \in E$ for all i and $(a_1, b_2) \notin E$, $(a_1, b_3) \notin E$, $(a_2, b_3) \notin E$, $(a_3, b_2) \notin E$, from Proposition 4.1(A), either $(a_2, b_1) \in E$ or $(a_3, b_1) \in E$ since G has no 3-anti-sun. So v is adjacent to either z_j or z_{j+3} in G. This proves Step 2 for the case of $d_G(v, x_j) = 1$.

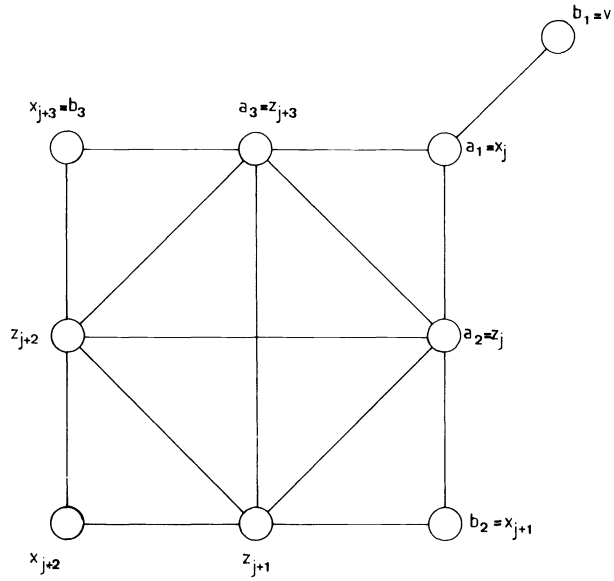


FIG. 5.1

If $d_G(v, x_j) = 2$, let (v, w, x_j) be a path in G. Assume v is not adjacent to any z_k. Then, as in Fig. 5.2, w is not a vertex of the complete 4-sun since w is adjacent to x_j which is adjacent only to z_j and z_{j+3} in the complete 4-sun.

For the same reason as in the previous case, w is adjacent to either z_j or z_{j+3}. Because of symmetry we can assume, without loss of generality, that $(w, z_j) \in E$ as in Fig. 5.3.

Apply Proposition 4.1(A) to the six named vertices in Fig. 5.3. Then w is adjacent to z_{j+1} or z_{j+3} in G. If w is adjacent to z_{j+1} but not z_{j+3}, then $(w, z_{j+1}, z_{j+3}, x_j, w)$ is a 4-hole in G, which is impossible since G is chordal. So $(w, z_{j+3}) \in E$ and we have Fig. 5.4.

Now apply Proposition 4.1(A) to the six named vertices in Fig. 5.4. Then either $(w, x_{j+1}) \in E$ or $(w, x_{j+3}) \in E$. Because of symmetry, we can assume $(w, x_{j+1}) \in E$. In the cycle $(w, x_{j+1}, z_{j+1}, z_{j+3}, w)$, $(x_{j+1}, z_{j+3}) \notin E$ implies $(w, z_{j+1}) \in E$. So we have Fig. 5.5.

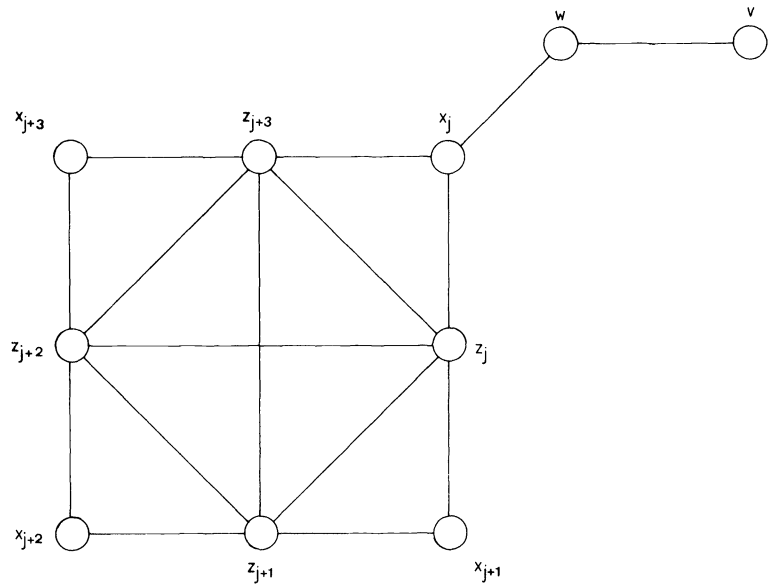


FIG. 5.2

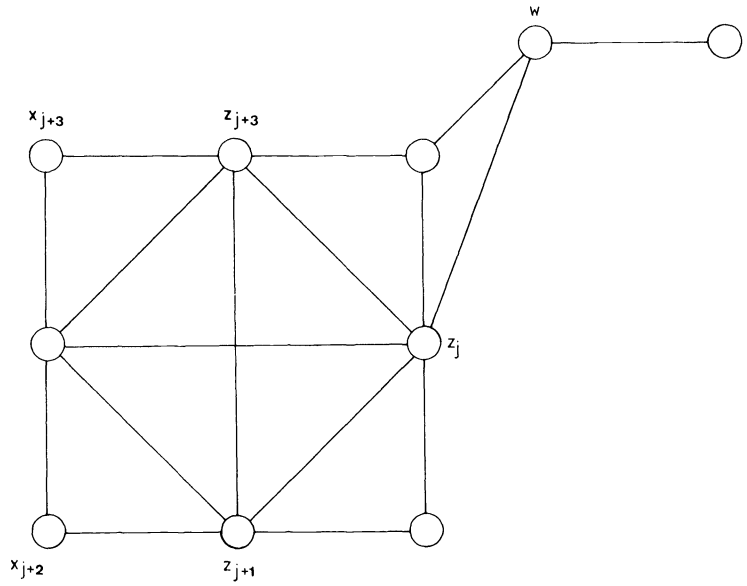


FIG. 5.3

Apply Proposition 4.1(B) to the six named vertices in Fig. 5.5. Note that $(a_1, b_1) \notin E$, $(a_1, b_2) \in E$, $(a_1, b_3) \in E$, $(a_2, b_1) \in E$, $(a_2, b_3) \in E$, $(a_3, b_1) \in E$, $(a_3, b_2) \in E$ and $(a_3, b_3) \notin E$. Since G is 3SF-chordal, Proposition 4.1(B) implies $(a_2, b_2) = (w, z_{j+2}) \in E$. So we have Fig. 5.6.

Apply Proposition 4.1(A) to the six named vertices in Fig. 5.6. Then either $(w, x_{j+2}) \in E$ or $(w, x_{j+3}) \in E$. But we know that $(w, x_j) \in E$ and $(w, x_{j+1}) \in E$. So either

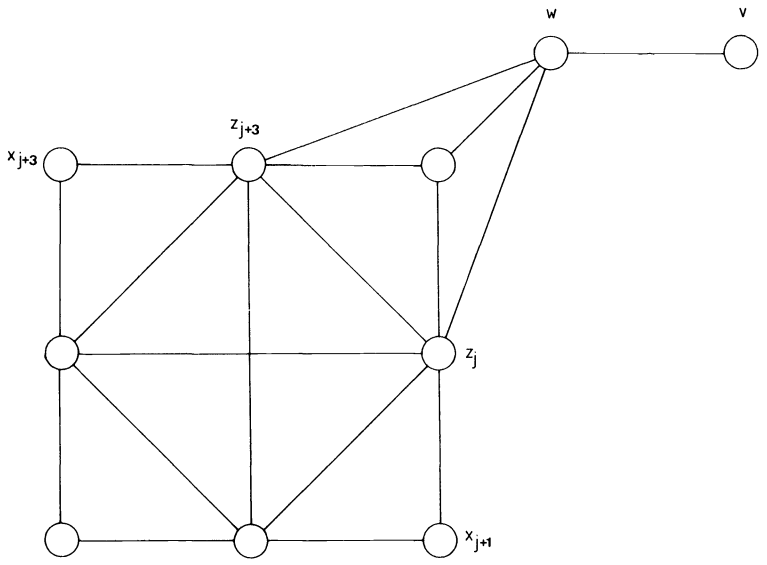


FIG. 5.4

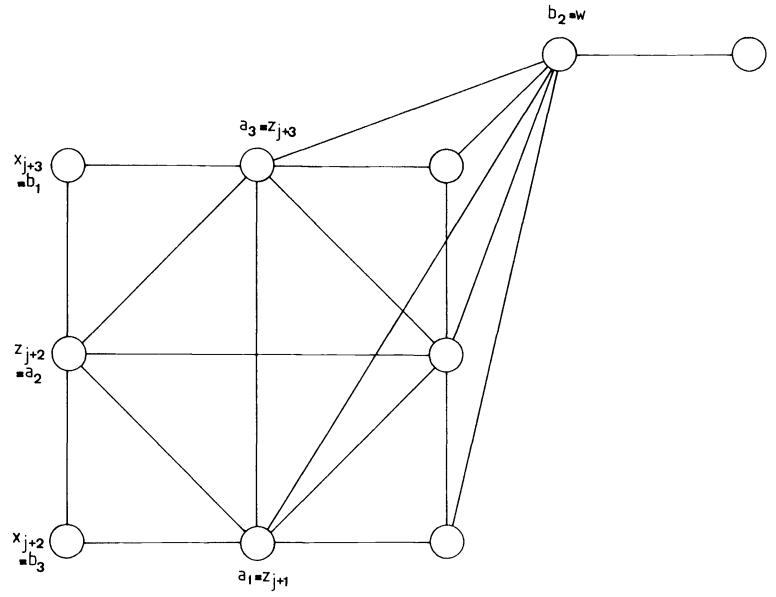


FIG. 5.5

$d_G(x_j, x_{j+2}) = 2$ or $d_G(x_{j+1}, x_{j+3}) = 2$, which is not possible since $(x_1, x_2, x_3, x_4, x_1)$ is a 4-hole in G^2. This proves Step 2 for the case of $d_G(v, x_j) = 2$.

Step 3. If $(v, x_j) \in E^2$ for some j, by Step 2, v is adjacent to some z_k; see Fig. 5.7. Now apply Proposition 4.1(A) to $\{z_k, z_{k+1}, z_{k+3}, v, x_{k+2}, x_{k+3}\}$ and obtain $(v, z_{k+1}) \in E$ or $(v, z_{k+3}) \in E$. Without loss of generality, assume $(v, z_{k+1}) \in E$. So v is adjacent to x_k, x_{k+1}, x_{k+2} in G^2. This proves (2) and (3).

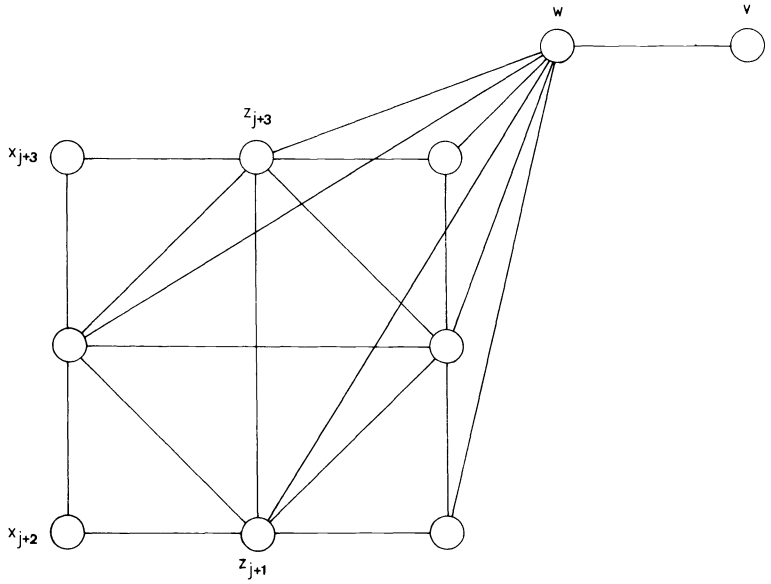


FIG. 5.6

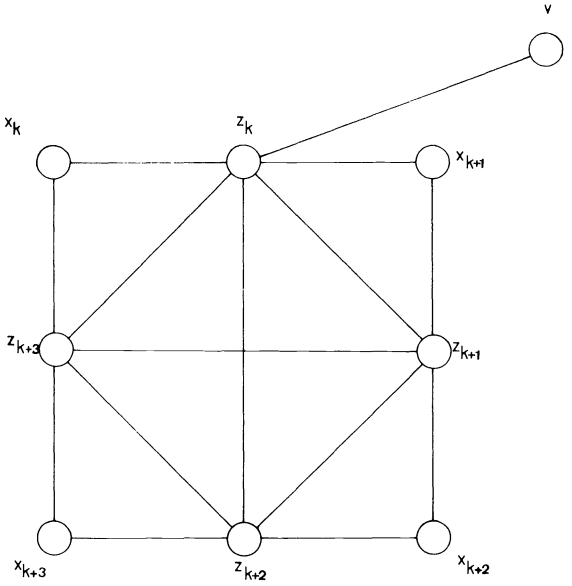


FIG. 5.7

Step 4. For (4) we need to prove that $(v, z_{k+2}) \in E$ and $(v, z_{k+3}) \in E$ when $(v, x_i) \in E^2$ for $i = 1, 2, 3, 4$.

Proof of Step 4. First we prove that either $(v, z_{k+2}) \in E$ or $(v, z_{k+3}) \in E$. Suppose this is not the case.

Case 1. $d_G(v, x_{k+3}) = 1$. Consider the cycle $(v, z_{k+1}, z_{k+2}, x_{k+3}, v)$. $(z_{k+1}, x_{k+3}) \notin E$ implies $(v, z_{k+2}) \in E$, which is a contradiction.

Case 2. $d_G(v, x_{k+3}) = 2$. Let (v, w, x_{k+3}) be a path in G as in Fig. 5.8. Note that w is not in the complete 4-sun since x_{k+3} is adjacent only to z_{k+2} and z_{k+3} in the sun. Consider the cycle $(v, z_{k+1}, z_{k+2}, x_{k+3}, w, v)$. Since $(v, z_{k+2}) \notin E$, $(v, x_{k+3}) \notin E$ and $(z_{k+1}, x_{k+3}) \notin E$, we have $(w, z_{k+1}) \in E$ and $(w, z_{k+2}) \in E$ as in Fig. 5.8.

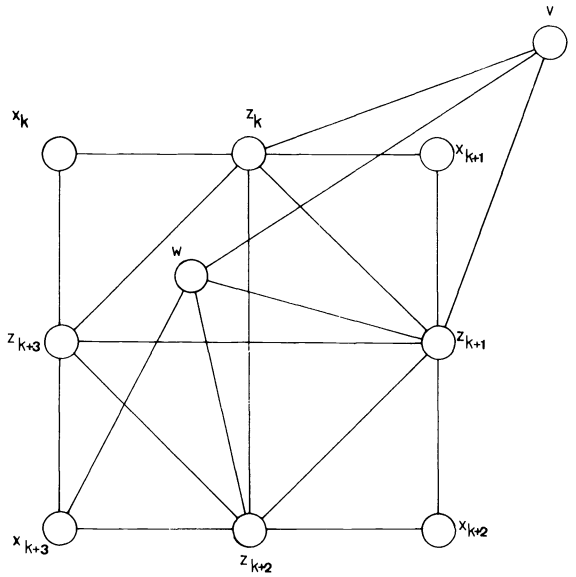


FIG. 5.8

Apply Proposition 4.1(B) to the six vertices $\{w, z_{k+1}, z_{k+2}, x_{k+2}, x_{k+3}, v\}$ in Fig. 5.8. Then $(w, x_{k+2}) \in E$. Similarly, $(w, x_k) \in E$. So $d_G(x_k, x_{k+2}) = 2$, which contradicts the assumption that $(x_1, \cdots, x_4, x_1)$ is a hole in G^2. Therefore $(v, z_{k+2}) \in E$ or $(v, z_{k+3}) \in E$. By symmetry we can assume $(v, z_{k+2}) \in E$ as in Fig. 5.9.

Now apply Proposition 4.1(B) to the six vertices $\{z_k, z_{k+2}, z_{k+3}, x_{k+3}, x_k, v\}$ as in Fig. 5.9, which yields $(v, z_{k+3}) \in E$. This proves Step 4 and hence (4) □

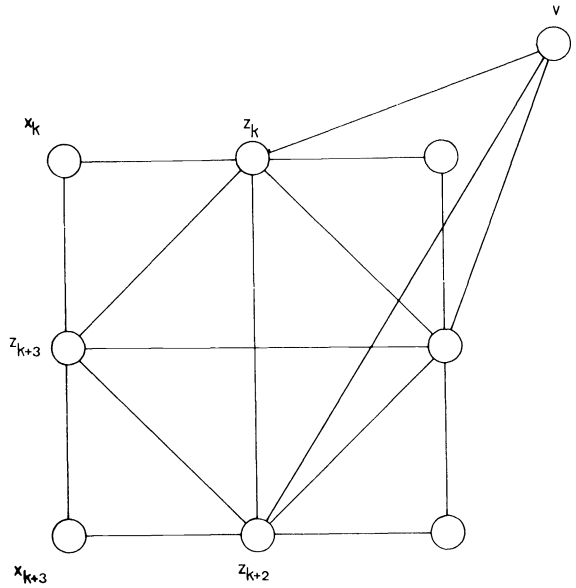


FIG. 5.9

THEOREM 5.8. *If G is* 3S3ASF-*chordal, then G^2 is nearly chordal.*

Proof. (NC1) follows from Lemma 5.7. (NC2) follows from (2) of Lemma 5.7. By (3) and (4) of Lemma 5.7, if v is adjacent to x_i and x_{i+1} in G^2, then $d_G(v, z_i) \leqq 1$. Hence $d_G(v, v') \leqq 2$ for all v, v' adjacent to x_i and x_{i+1} in G^2. This proves (NC3) for G^2. □

REFERENCES

C. BERGE [1960], *Les problèmes de colorations en théorie des graphs*, Publ. Inst. Statist. Univ. Paris, 9, pp. 123–160.

——— [1972], *Balanced matrices*, Math. Programming, 2, pp. 19–31.

——— [1973], *Graphs and Hypergraphs*, American Elsevier, New York.

M. BURLET [1982], Personal communication.

M. BURLET AND J. FONLUPT [1982], *Polynomial algorithm to recognize a Meyniel graph*, Research Report 303, Lab. Inform. Math. Appl. de Grenoble, France.

G. J. CHANG [1982], *k-domination and graph covering problems*, Ph.D. thesis, School of Operations Research and Industrial Engineering, Cornell Univ., Ithaca, NY.

G. J. CHANG AND G. L. NEMHAUSER [1982], *The k-domination and k-stability problems on graphs*, Tech. Report 540, School of Operations Research and Industrial Engineering, Cornell Univ., Ithaca, NY; this Journal, 5 (1984), pp. 332–345.

N. CHRISTOFIDES [1975], *Graph Theory—An Algorithmic Approach*, Academic Press, New York.

V. CHVÁTAL [1981], *Perfectly ordered graphs*, Tech. Report SOCS-81.28, McGill Univ., Montreal.

E. J. COCKAYNE S. GOODMAN AND S. T. HEDETNIEMI [1975], *A linear algorithm for the domination number of a tree*, Inform. Proc. Letters, 4, pp. 41–44.

D. R. FULKERSON, A. J. HOFFMAN AND R. OPPENHEIM [1974], *On balanced matrices*, Math. Programming Study 1, pp. 120–132.

M. C. GOLUMBIC [1980], *Algorithmic Graph Theory and Perfect Graphs*, Academic Press, New York.

M. GROTSCHEL, L. LOVÁSZ AND A. SCHRIJVER [1981], *Polynomial algorithms for perfect graphs*, Report No. 81176-0R, Institut für Ökonometrie und Operations Research, Universität Bonn, Bonn, W. Germany.

A. HAJNAL AND J. SURÁNYI [1958], *Über die Auflösung von Graphen in vollständige Teilgraphen*, Ann. Univ. Sci. Budapest Eötvös. Sect. Math., 1, pp. 113–121.

L. LOVÁSZ [1972], *Normal hypergraphs and the perfect graph conjecture*, Discrete Math., 2, pp. 253–267.

A. MEIR AND J. W. MOON [1975], *Relation between packing and covering of a tree*, Pacific J. Math., 61, pp. 225–233.

H. MEYNIEL [1976], *On the perfect graph conjecture*, Discrete Math., 16, pp. 339–342.

D. J. ROSE, R. E. TARJAN AND G. S. LUEKER [1976], *Algorithmic aspects of vertex elimination on graphs*, SIAM J. Comput., 5, pp. 266–283.

P. J. SLATER [1976], *R-domination in graphs*, J. Assoc. Comp. Mach., 23, pp. 446–450.

B. C. TANSEL, R. L. FRANCIS AND T. J. LOWE [1983], *Location on networks: a survey*, Parts I, II Management Sci., 29, pp. 482–511.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 1, January 1985

013

INTRINSIC LIMITATIONS OF THE MAJORITY RULE, AN ALGORITHMIC APPROACH*

JAMES M. ABELLO†

Abstract. The consistent sets of permutations are those over which unrestricted choice necessarily produces transitive relations under simple majority vote.

The main goal of this paper is to shed some light on the structure of maximal consistent sets. By using graph theoretical techniques we have been able to design an algorithm which generates maximal connected consistent sets. The obtained results support the following author's conjecture: *If $|M_n|$ denotes the cardinality of a maximum consistent set then $2^{n-1} < |M_n| < 2^n$ for $n \geq 4$ where n is the number of alternatives on which voting is taking place.* This implies that if majority rule is the aggregation procedure the number of different opinions an individual is allowed to have is less than 2^n, which indicates that not much more individual diversity is allowed by giving the voter unrestricted choice than by circumscribing him to choose from a more restrictive looking Blackian domain.

Key words. algorithm, Arrow's theorem, Blackian domain, connected graphs, consistent sets, simple majority rule

Introduction. It is a very well-known fact that simple majority voting produces a social relation which is not necessarily transitive (Arrow [2]). Domain restrictions under which majority vote avoids the intransitivity flaw can be found in Bowman [3] and Inada [5]. Some general discussions in this area are presented in Fishburn [4] and Bowman [3] and a probabilistic treatment in Kelly [7] and Klahr [8].

Since the unfortunate aspect of domain restrictions is that a sufficiently rich realm of choice may not remain, it is natural to ask how much freedom of choice is consistent with transitivity, and we take the size of the domain as a rough measure of the degree of choice. When the number of alternatives is n, quite structurally different, transitive domains (consistent sets) of cardinality 2^{n-1} have been constructed (Abello [1], Johnson [6], Abello and Johnson [10]) and it has been proved that the maximum number of votes in a profile following the "single peaked" condition (a Blackian domain) is 2^{n-1} (Raynaud [9]).

We have found general maximal transitive domains of cardinality $(\frac{3}{2})2^{n-1}-4$ for $n>4$, which to our knowledge is the best known lower bound [1], [10].

In this paper we present general results which lay down the foundations of an algorithm to produce maximal consistent sets. Some of the transitive domains constructed in previous works appear as special outputs of the proposed procedure.

The contents of this work add to the already accumulated evidence supporting the following conjecture:

The cardinality of a maximum consistent set is less than 2^n for $n \geq 4$.

The truth of the conjecture will indicate that if majority rule is the aggregation procedure not much more individual diversity is allowed by giving the voter unrestricted choice than by circumscribing him to choose from a more restrictive Blackian domain.

1. Problem formulation and graph representation. See [10] for further discussion. Let $\langle \Sigma, \leqq \rangle$ be a totally ordered set of symbols of cardinality $|\Sigma| = n \in Z^+$, and S_Σ the set of permutations on Σ.

* Received by the editors June 30, 1983. Portions of this work were presented at the SIAM Second Conference on the Applications of Discrete Mathematics, held at Massachusetts Institute of Technology, Cambridge, Massachusetts, June 27–29, 1983.

† Mathematics Department, University of California at San Diego, La Jolla, California 92093.

DEFINITION 1.1. A set $B \subset S_\Sigma$ is called *cyclic* if there are three symbols $x_{i_1}, x_{i_2}, x_{i_3} \in \Sigma$ and three permutations in B which, when restricted to the three symbols, are

$$\begin{pmatrix} x_{i_1} & x_{i_2} & x_{i_3} \\ x_{i_3} & x_{i_1} & x_{i_2} \\ x_{i_2} & x_{i_3} & x_{i_1} \end{pmatrix}.$$

DEFINITION 1.2.

i) Let $u, v, w \in S_\Sigma$; if the set $\{u, v, w\}$ is not cyclic it is called a *consistent* three-set.

ii) A subset C of S_Σ for which every three-subset is consistent is called a *consistent subset* of S_Σ.

Example 1.1. Let $\Sigma = \{1, 2, 3, 4\}$. The set $C = (1234, 4123, 4321, 4312\}$ is consistent because every three-subset is consistent. Notice that this requires checking for the consistency of the $\binom{|C|}{3}$[1] three-subsets of C. Moreover, for each three-subset it is necessary to check each of the $\binom{|\Sigma|}{3}$ triples of symbols of Σ for the noncyclicity condition. It should be clear at this point that this task is computationally expensive even for moderately large values of $n = |\Sigma|$.

DEFINITION 1.3.

i) If $p \in S_\Sigma$, by $T(p)$ we will denote the set of ordered triples of symbols of Σ determined by p and by $\mathcal{P}(p)$ we will denote the set of ordered pairs determined by p. $\tau(p)$ will denote the set of adjacent ordered pair of symbols appearing in p. Each ordered pair can be interpreted in a natural and unique way as a transposition and under this interpretation we will refer to $\tau(p)$ as p's admissible set of transpositions. Notice that $\tau(p) \subsetneq \mathcal{P}(p)$ and $|\tau(p)| = n-1$. If $t \in \tau(p)$ then $t(p)$ is the permutation obtained by applying the transposition t to p. If $t = (x, y)$ then $t^{-1} = (y, x)$ and $t \in \tau(p)$ iff $t^{-1} \in \tau(t(p))$.

ii) If $C \subseteq S_\Sigma$, then $T(C) \equiv \bigcup_{p \in C} T(p)$, $\mathcal{P}(C) \equiv \bigcup_{p \in C} \mathcal{P}(p)$ and $\tau(C) \equiv \bigcup_{p \in C} \tau(p)$.

The following (summarized from [10]) are some elementary properties of consistent sets:

FACT 1.1.

i) *Any subset of a consistent set is consistent.*

ii) *Any superset of a cyclic set is cyclic.*

iii) *The intersection of consistent sets is a consistent set but their union is not always consistent.*

iv) $|T(S_\Sigma)| = P(|\Sigma|, 3)$ (*the number of different* 3*-permutations out of a set of* $|\Sigma|$*-elements*).

v) *If* C *is a consistent subset of* S_Σ *then* $|T(C)| \leqq 4\binom{|\Sigma|}{3}$.

vi) C *is a consistent set iff* $C^* = C \cup \{w | T(w) \subset T(C)\}$ *is consistent.*

Graph representation. Consider a graph $G_n = (V, E)$ where $V = S_\Sigma$, $n = |\Sigma|$ and two vertices u, v are joined by an edge iff there exists an adjacent transposition l such that $u = l(v)$.

When two vertices u, v are adjacent the arc is directed from u to v if $u = u_1 \cdots u_i u_{i+1} \cdots u_n$ and $v = u_1 \cdots u_{i+1} u_i \cdots u_n$ with $u_i < u_{i+1}$. It is clear that the degree of $u = n-1$, $\forall u \in G_n$ (see Fig. 1).

DEFINITION 1.4. If $u = u_1 \cdots u_n$ let $\varepsilon(u)$ be the set of pairs (u_i, u_j) which do not introduce an inversion, i.e.,

$$\varepsilon(u) = \{(u_i, u_j) | i < j, u_i < u_j\}.$$

[1] $\binom{n}{k}$ denotes the binomial coefficient.

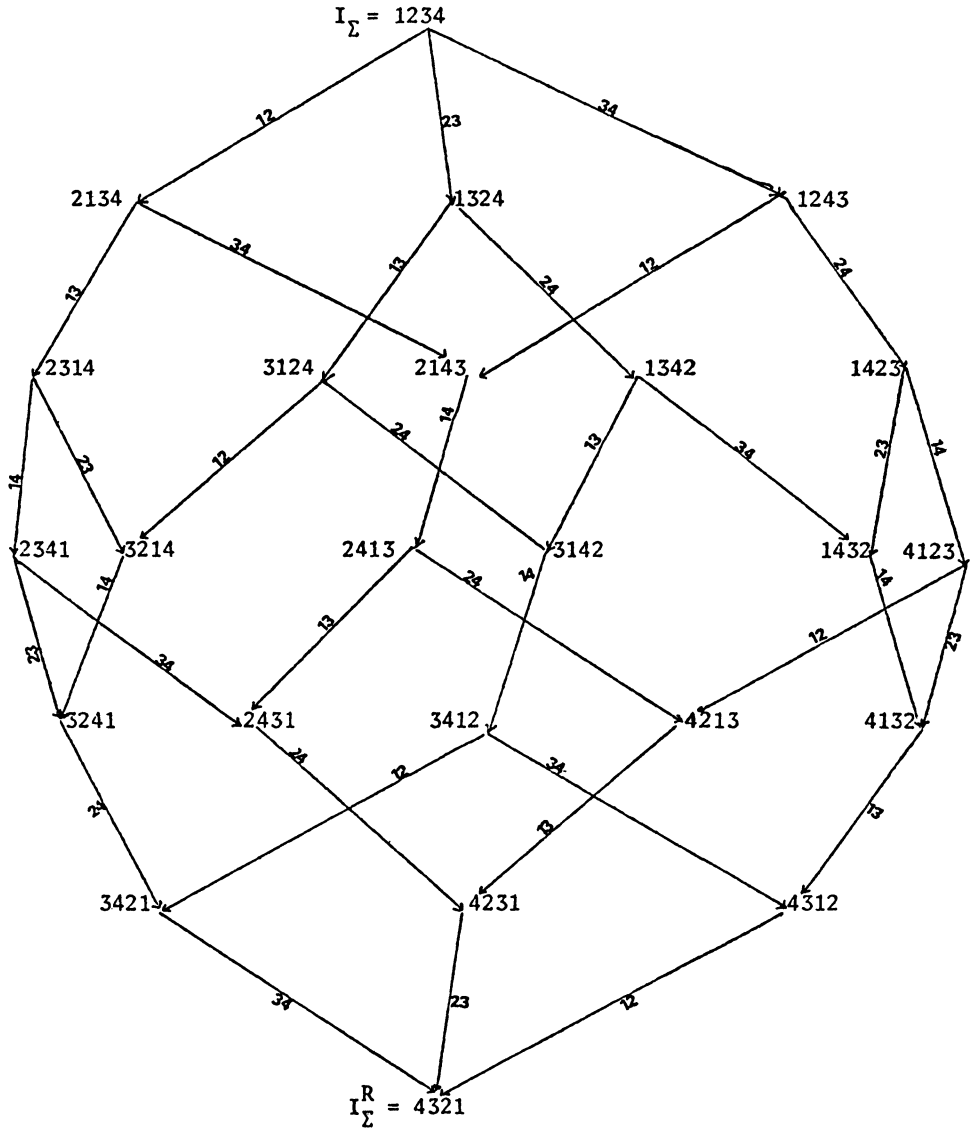


FIG. 1. *Representation of the graph of the permutations of* $\Sigma = \{1, 2, 3, 4\}$. *The relevant transpositions are indicated on each edge.*

DEFINITION 1.5. For $u, v \in S_\Sigma$, $u \geqq v$ iff $\varepsilon(u) \supseteq \varepsilon(v)$.

FACT 1.2. $\geqq$ *is an order relation on* S_Σ *and* $\langle S_\Sigma, \geqq \rangle$ *is a poset with maximum element* I_Σ (*the identity in* S_Σ) *and minimum element* I_Σ^R (*the reverse of* I_Σ).

The following lemma gives the first relation between the poset $\langle S_\Sigma, \geqq \rangle$ and the class of consistent subsets of S_Σ.

LEMMA 1.1. *If* L *is a chain in* $\langle S_\Sigma, \geqq \rangle$ *then* L *is a consistent subset of* S_Σ.

Proof. See [10].

Example 1.2. The set $\{1234, 1243, 1423, 4123, 4132, 4312, 4321\} \subset S_{\{1,2,3,4\}}$ is consistent because it is a chain in $\langle S_{\{1,2,3,4\}}, \geqq \rangle$ (see Fig. 1).

2. Classes of consistent sets.

DEFINITION 2.1.

i) A subset K_0 of a consistent set C is called a kernel for C iff $T(C) = T(K_0)$ and $|K_0| = \min_{|K|}\{K \subset C\}$.

ii) $\mathscr{C}_K \equiv \{A \subset S_\Sigma\colon A \text{ is consistent and } K \text{ is a kernel for } A\}$.

Note. There is a unique maximum cardinality set X_K in $\mathscr{C}_K$.

FACT 2.1. *Let K_0 and K_1 be kernels of $\mathscr{C}_{K_0}$ and $\mathscr{C}_{K_1}$ respectively. If $T(K_0) \subset T(K_1)$ then $X_{K_0} \subset X_{K_1}$ where X_{K_0} and X_{K_1} denote the maximum sets in their classes.*

Proof. (by contradiction). Assume $\exists p\colon p \in X_{K_0} \backslash X_{K_1}$. $T(p) \subset T(K_0) = T(X_{K_0}) \subset T(X_{K_1})$ because K_0 and K_1 are kernels and $T(K_0) \subset T(K_1)$. Then $X_{K_1} \cup \{p\}$ is a consistent set in $\mathscr{C}_{K_1}$ of cardinality greater than $|X_{K_1}|$, a contradiction. Q.E.D.

COROLLARY 2.1. *Any maximum consistent set must contain a kernel K such that $T(K)$ is of cardinality $4\binom{n}{3}$.*

Proof. Immediate from Fact 2.1. Q.E.D.

Any maximum chain L in $\langle S_\Sigma, \geqq\rangle$ is a consistent set such that $|T(L)| = 4\binom{n}{3}$ which suggests that the maximum cardinality sets X_L in the classes $\mathscr{C}_L$ are among the candidates for a maximum consistent set. However it is not always true that if L and L' are maximal chains then $|X_L| = |X_{L'}|$ (see Fig. 2). Moreover, there are maximum consistent sets which do not contain a maximal chain as shown by the following example.

Example 2.1. $n = 3, \Sigma = \{1, 2, 3\}, K = \{123, 213, 321, 312\}, X_K = K$ (see Fig. 3). K does not contain a maximal chain; however K is a maximum consistent set.

We have seen that chains in $\langle S_\Sigma, \geqq\rangle$ appear to be a very important structure for a large class of consistent sets. Now we will define other kernels called "skeletons" which are structurally equivalent to maximal chains from the consistency point of view.

DEFINITION 2.2. A *skeleton* $S(p)$ is an $\binom{n}{2}+1$ ordered set of permutations $(P_0, P_1, \cdots, P_{\binom{n}{2}-1})$ such that;

i) $P_0 = p$, $P_{i+1} = t_{i+1}(P_i)$ where $t_{i+1} \in \tau(P_i)$, $i = 0, 1, \cdots \binom{n}{2} - 1$,

ii) for $i \neq j$, $t_i \neq t_j$ and $t_i \neq t_j^{-1}$.

FACT 2.2.

i) *Let $S(p) = (P_0, P_1, \cdots, P_{\binom{n}{2}})$ be a skeleton and $k \in \{0, 1, \cdots \binom{n}{2} - 1\}$. If $t_{k+1} = (x, y)$ then $(\{P_0, P_1 \cdots, P_k\})$ does not contain the ordered pair (y, x).*

ii) *Let I be the identity in S_n. Then we have the following equivalence. $S(I)$ is a skeleton iff $S(I)$ is a maximal chain in S_n.*

iii) *If $S(p)$ is a skeleton then the set $S^*(p) = S(p) \cup \{w | T(w) \subset T(\{P_0, P_1, \cdots, P_{\binom{n}{2}-1}\})\}$ is a consistent set.*

Proof. i) and ii) follow readily from the definition. For iii) consider $p = p_1 p_2 \cdots p_n \neq I$ and the mapping $p_i \to i$. This mapping gives us a one-to-one correspondence between the skeleton $S(p)$ and some skeleton $S(I)$ such that $S(p)$ is consistent iff $S(I)$ is. But we know that $S(I)$ is consistent because it is a chain (Fact 2.2 ii and Lemma 1.1), therefore $S(p)$ and $S^*(p)$ are consistent. Q.E.D.

For the remainder of this section $(P_0, P_1, \cdots, P_{\binom{n}{2}})$ will denote a skeleton, and for each $i = 0, 1, 2, \cdots, \binom{n}{2} - 1$, t_{i+1} will denote the corresponding transposition (see Definition 2.2 above).

LEMMA 2.1. *Let $w \in S_n$ and $k \in \{0, 1, 2, \cdots, \binom{n}{2} - 1\}$.*

$$T(w) \subset T(\{P_0, P_1, \cdots, P_k\}) \cup T(P_{k+1})$$

$$\Rightarrow T(w)/T(\{P_0, P_1, \cdots, P_k\}) = \varnothing \text{ or } T(P_{k+1})/T(\{P_0, P_1, \cdots, P_k\}).$$

Proof (by contradiction). Assume that $T(w)/T(\{P_0, P_1, \cdots, P_k\}) \neq \varnothing$ and

$$T(w)/T(\{P_0, P_1, \cdots P_k\}) \subsetneq T(P_{k+1})/T(\{P_0, P_1, \cdots, P_k\}). \tag{1}$$

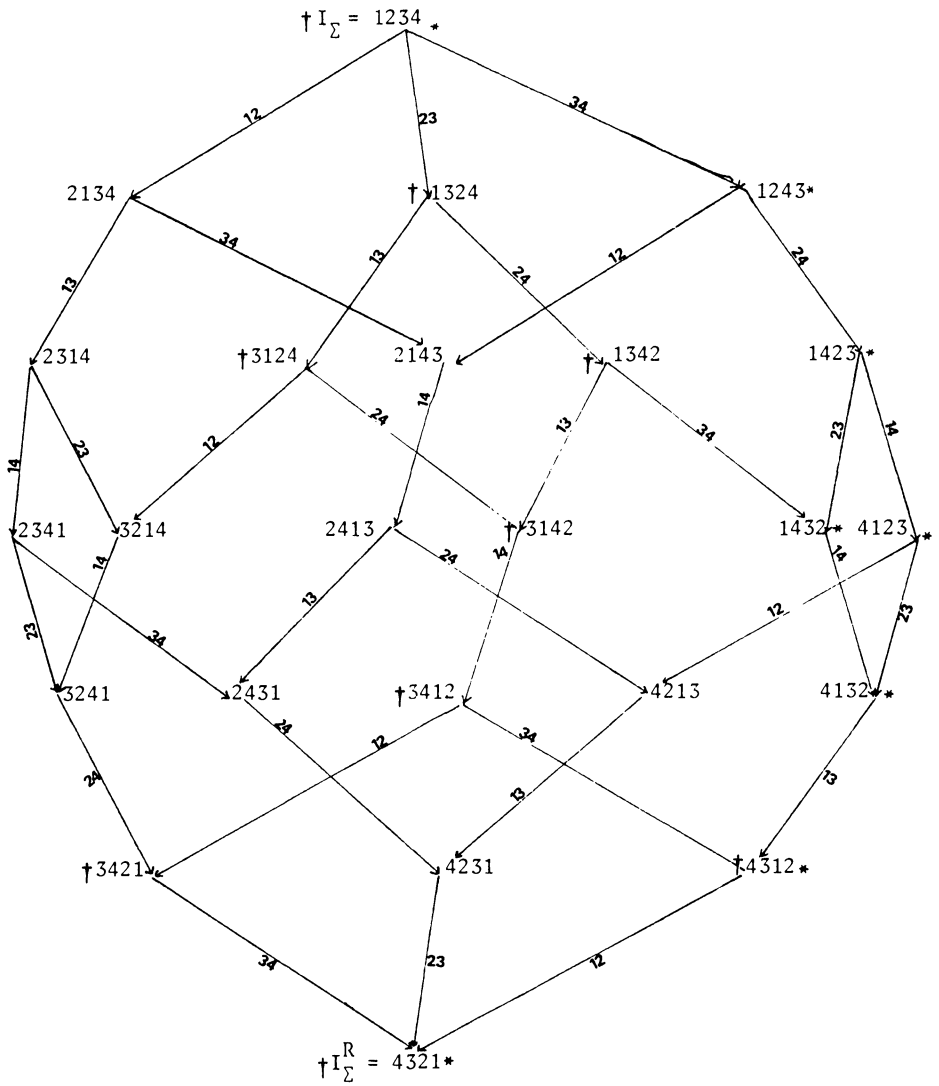


FIG. 2. *Two maximal consistent sets of different cardinality each containing a maximal chain. The elements of one set are marked by † and the elements of the other set appear with a * immediately at their right.*

If $t_{k+1}=(x, y)$ then each ordered triple in $T(w)/T(\{P_0, P_1, \cdots, P_k\})$ must involve (y, x) because $T(P_{k+1})/T(\{P_0, P_1, \cdots, P_k\})=\{(-, y, x), (y, x, -)\}$. Now, w is not of the form $\cdots y \cdots x \cdots$ because it would imply that $T(\{P_0, \cdots, P_k\})$ contains triples of the form $(y, -, x)$, which contradicts the fact that $\mathcal{P}(\{P_0, \cdots, P_k\})$ does not contain the ordered pair (y, x) (see Fact 2.2i). Therefore w must be of the form

(2) $$\cdots yx \cdots$$

Let

(3)

$$(y, x, z) \in T(P_{k+1})/T(\{P_0, P_1, \cdots, P_k\}) \quad \text{and} \quad (y, x, z) \notin T(w)/T(\{P_0, P_1, \cdots, P_k\})$$

(see (1) above).

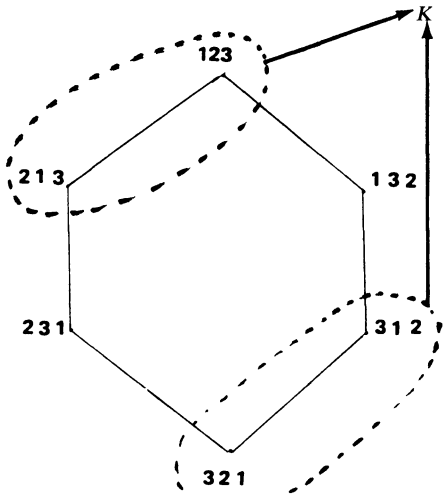


FIG. 3

By (2) and (3) $(z, y, x) \in T(w)/T(\{P_0, P_1, \cdots, P_k\})$ and so $(z, y, x) \in T(P_{k+1})/T(\{P_0, P_1, \cdots, P_k\})$, which is a contradiction to the fact that $(y, x, z) \in T(P_{k+1})/T(\{P_0, P_1, \cdots, P_k\})$. Q.E.D.

COROLLARY 2.2. *If* $T(w) \subset T(\{P_0, P_1, \cdots, P_k\}) \cup T(P_{k+1})$ *then*

$$|T(w)/T(\{P_0, P_1, \cdots, P_k\})| = 0 \text{ or } n-2.$$

Proof. Follows from the preceding lemma and the fact that

$$|T(P_{k+1})/T(\{P_0, P_1, \cdots, P_k\})| = n-2.$$

All the preceding machinery is justified by the following results which show that those maximal consistent sets which have skeletons as kernels have a structure determined completely by the order of the elements of the skeleton and are bounded to be connected subgraphs of $\langle S_n, \geqq \rangle$. The following theorem is sort of a projective result which together with Corollary 2.4 below form the basis of an algorithm to generate maximal connected consistent sets.

THEOREM 2.1. *If* $w \in S_n$ *is such that* $T(w) \subset T(\{P_0, P_1, \cdots, P_i\}) \cup T(P_{i+1})$ *for some* $i \in \{0, 1, \cdots, \binom{n}{2}-1\}$ *then* $w \in C_{i+1} = \{v \in S_n : v = t_{i+1}(u)$ *for* $u \in C_i\} \cup C_i$ *(here* $C_0 = \{P_0\}$*).*

Proof. Without loss of generality we can assume that $P_0 = I$ (the identity in S_n). The proof is by induction on i.

Basis. $i = 0$. Let $w \in S_n$: $T(w) \subset T(P_0) \cup T(P_1)$. In this case the result follows from the fact that the maximum number of permutations in S_n determined by $\binom{n}{3} + (n-2)$ triples is 2.

Induction hypothesis. Assume the result is true for $0 < i = k < \binom{n}{z}$, namely $T(w) \subset T(\{P_0, P_1, \cdots, P_{k-1}) \cup T(P_k) \Rightarrow w \in C_k$. We must prove that $T(w) \subset T(\{P_0, P_1, \cdots, P_k\}) \cup T(P_{k+1}) \Rightarrow w \in C^*_{k+1}$ (see the definition of C^* in Fact 1.1). If $T(w) \subset T(\{P_0, P_1, \cdots, P_k\})$ then $w \in C^*_k \subset C^*_{k+1}$ by the induction hypothesis; so let us assume that $T(w) \not\subset T(\{P_0, P_1, \cdots, P_k\})$. In this case $T(w)/T(\{P_0, P_1, \cdots, P_k\}) = T(P_{k+1})/T(\{P_0, \cdots, P_k\})$ by the preceding lemma and its corollary.

Now if $t_{k+1} = (x, y)$ then x and y must occupy the same adjacent positions in P_{k+1} and w; also any symbol preceding y in P_{k+1} must appear preceding y in w and any symbol preceded by x in P_{k+1} must be preceded by x in w (remember that $t_{k+1}(P_k) = P_{k+1}$ and $T(w)/T(\{P_0, P_1, \cdots, P_k\}) = T(P_{k+1})/T(\{P_0, \cdots, P_k\})$). On the other hand

any other triple not involving both symbols x and y in either P_{k+1} or w must be an element of $T(\{P_0, P_1, \cdots, P_k\})$ because $T(w) \subset T(\{P_0, P_1, \cdots, P_k\}) \cup T(P_{k+1})$. Therefore, the permutation $t_{k+1}^{-1}(w)$ is such that $T(t_{k+1}^{-1}(w)) \subset T(\{P_0, \cdots, P_k\})$ which implies that $t_{k+1}^{-1}(w) \in C_k$ by the induction hypothesis, so w is obtained as a projection by t_{k+1} of an element in C_k, namely $t_{k+1}^{-1}(w)$. Q.E.D.

DEFINITION 2.3. Let T' be a consistent subset of $T(S_n)$. A set C is called *maximally consistent* (m.c.) with respect to T' iff

i) $T' = T(C)$

and

ii) $\{w \in S_n | T(w) \subseteq T'\} \subseteq C$.

DEFINITION 2.4. For $i \in \{0, 1, \cdots, \binom{n}{2} - 1\}$ and a skeleton $(P_0, P_1, \cdots, P_{\binom{n}{2}})$ consider the following recursive definition of C_i^*:

$$C_0^* = \{P_0\}, C_{i+1}^* = \{w \in S_n : T(w) \subseteq T(C_i^*) \cup T(P_{i+1})\}.$$

COROLLARY 2.3. *For* $j \in \{0, 1, \cdots, \binom{n}{2}\}$ C_j^* *is m.c. with respect to* $T(\{P_0, P_1, \cdots, P_j\})$.

Proof. It is clear that

$$T(C_{j+1}^*) = T(C_j^*) \cup T(P_{j+1}) = T(\{P_0, P_1, \cdots, P_j\}) \cup T(P_{j+1})$$
$$= T(\{P_0, P_1, \cdots, P_j, P_{j+1}\}).$$

Thus C_{j+1}^* is consistent because $\{P_0, P_1, \cdots, P_j, P_{j+1}\}$ is a skeleton which is consistent by Fact 2.2 ii above. Q.E.D.

Incidentally notice that

$$C_{\binom{n}{2}}^* = \{P_0, P_1, \cdots, P_{\binom{n}{2}}\} \cup \{w | T(w) \subset T(\{P_0, P_1, \cdots, P_{\binom{n}{2}}\})\}$$

and that if $T(w) \subset T(C_j^*)$ then $w \in C_j^*$.

DEFINITION 2.5. A permutation $\mu \in C_i^*$ is called *projectable* by t_{i+1} iff $t_{i+1} \in \tau(\mu)$, and is called *consistently projectable* when $T(t_{i+1}(\mu)) \subset T(C_i^*) \cup T(P_{i+1})$.

COROLLARY 2.4. $C_{i+1}^* = C_i^* \cup \{v | \mu = t_{i+1}^{-1}(v) \in C_i^*$ *and* μ *is consistently projectable by* $t_{i+1}\}$.

Proof. Follows from Definition 2.5 and Theorem 2.1. Q.E.D.

The preceding results indicate that majority rule produces transitive results if the collection of voter opinions as a whole can be partitioned (at least in the connected case) into no more than $(n^2+n)/2$ groups which can be ordered according with the level of disagreement they have with respect to a fixed permutation P.

The contents of Corollary 2.4 can be depicted graphically as in Fig. 4.

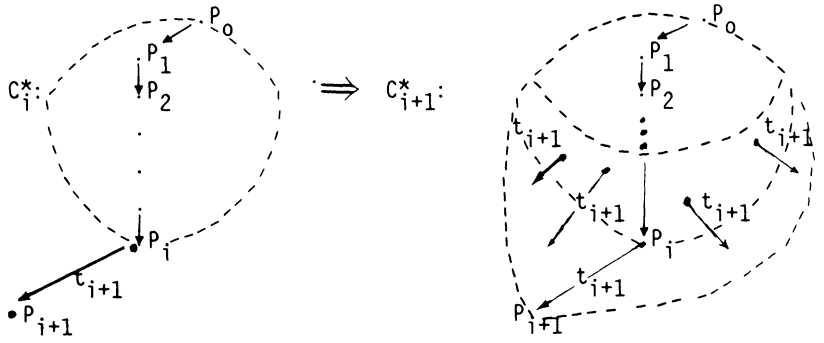


FIG. 4. *The projection theorem.*

THEOREM 2.2. *Let* $v, w \in C_i^*$ *such that* $t_{i+1} \in \tau(v) \cap \tau(w)$. *Assume that* $C_i^* \cup \{t_{i+1}(v), t_{i+1}(w)\}$ *is consistent and that* PATH (v, w) *is a path from* v *to* w: PATH $(v, w) \subseteq \{t_1, t_2, \cdots, t_i\}$.

Under these conditions t_{i+1} *must be disjoint from every transposition* t *in* PATH (v, w).

Proof. By induction on the length of PATH (v, w). Let $t_{i+1} = (x, y)$.

Basis. If $|\text{PATH}(v, w)| = 1$ then we have $v = \cdots xy \cdots$, $w = \cdots xy \cdots$ as in Fig. 5. Now if e is not disjoint from t_{i+1} then

(i) $e = (z, x)$, $z \neq x$, $z \neq y$ or $e = (y, z)$ which implies that t_{i+1} can not be admissible for both v and w.

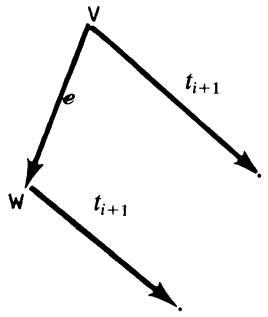


FIG. 5

Induction hypothesis. Assume the result is true for $|\text{PATH } v, w)| = k$ with $1 \leqq k < h \leqq \binom{n}{2}$. We must prove it for $|\text{PATH}(v, w)| = k+1$ where PATH $(v, w) = e_1, e_2, \cdots, e_{k+1}$.

(ii) If e_1 is not disjoint from t_{i+1} then t_{i+1} can not be admissible for $e_1(v)$ (see (i) above).

If t_{i+1} is admissible for $e_1(v)$ then t_{i+1} is disjoint from every e_i $i = 2, \cdots, k+1$ (by the induction hypothesis); now e_1 is disjoint from t_{i+1} by (ii), therefore t_{i+1} would be

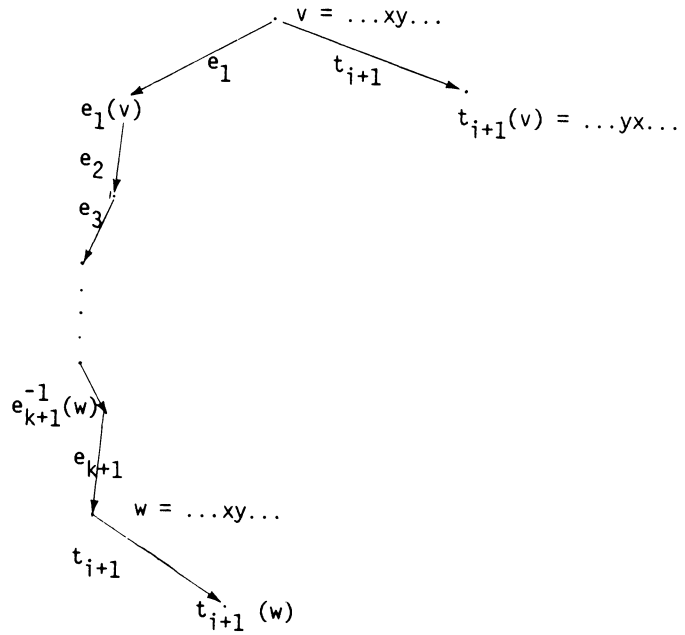


FIG. 6

in this case disjoint from every e_i in PATH (v, w) and we will be done. So, let us assume that t_{i+1} is admissible for both v and w; however e_1 is not disjoint from t_{i+1}. In this case we have that t_{i+1} is admissible for neither $e_1(v)$ nor $e_{k+1}^{-1}(w)$, which implies that t_{i+1} and e_{k+1} are not disjoint.

Now consider the permutations $e_1(v), w, t_{i+1}(w), t_{i+1}(v)$ and $t_{i+1} = (x, y)$.There are four cases to be considered

Case i. If $e_1 = (z, x)$ then $e_1(v)$ contains a triple of the form (x, z, y) and $t_{i+1}(v)$ contains a triple of the form (z, y, x).

Case ii. If $e_1 = (y, z)$ then $e_1(v)$ contains a triple of the form (x, z, y) and $t_{i+1}(v)$ contains a triple of the form (y, x, z).

Case iii. If $e_{k+1} = (x, z)$ then $e_{k+1}^{-1}(w)$ contains a triple of the form (x, z, y) and $t_{i+1}(w)$ contains a triple of the form (z, y, x).

Case iv. If $e_{k+1} = (z, y)$ then $e_{k+1}^{-1}(w)$ contains a triple of the form (x, z, y) and $t_{i+1}(w)$ contains a triple of the form (y, x, z).

Case i and *Case* iv imply that $\{e_1(v), t_{i+1}(v), e_{k+1}^{-1}(w), t_{i+1}(w)\}$ is not consistent, contradicting the hypothesis that $C_i^* \cup \{t_{i+1}(v), t_{i+1}(w)\}$ is consistent. Similarly for cases ii and iii. So the only combinations left for consideration are case i and case iii or case ii and iv, but in both situations we have $e_1 = e_{k+1}^{-1}$, contradicting the fact that no two transpositions used in PATH (v, w) may be inverses to each other because PATH (v, w) is a subset of a skeleton. Q.E.D.

3. An algorithm to generate maximal connected consistent sets in $\langle S_\Sigma, \geqq \rangle$. We will refer to this algorithm as the MCCS algorithm.

The algorithm consists of $\binom{n}{2}$ stages where $n = |\Sigma|$. At each stage a recursive procedure BACKTRACK (P, t, l) is used to find all the permutations which are consistently projectable by t. Theorem 2.2 is the justification for the correctness of this process.

Procedure BACKTRACK (P, t, l)

/*Find all the permutations in C^* which are consistently projectable by the adjacent transposition $t = (x, y) \in \tau(P)$. l is the position of the symbol x in the permutation P. Every visited permutation is marked. C^* and projection are global sets.*/

```
beginproc
  for (each unmarked predecessor R of P in C*) do
          begin
          MARK R;
          If (position of x and y in R = l and l+1 respectively) then
                    begin
                    Projection := Projection ∪ t(R);
                      call BACKTRACK (R, t, l)
                    end
          end
endproc
```

main ( ) /*$\mathcal{P}(P)$ denotes the set of ordered pairs determined by P. Each ordered pair defines a transposition in a natural way. There is no loss of generality in picking the identity I in $\langle S_\Sigma, \geqq \rangle$ as the initial permutation. Initially every permutation is unmarked; $\tau(P)$ is the set of P's admissible adjacent transpositions. C^* and Projection are global sets. At the end C^* contains a maximal connected consistent set as indicated by Theorem 2.1 and Corollaries 2.3 and 2.4.*/

```
    beginmain
 1. Pick up a permutation P_0 in ⟨S_Σ, ≧⟩; P := P_0;
 2. C* := Projection := {P}; τ := τ(P) ∩ 𝒫(P); 𝒮 := 𝒫(P);
 3. while (τ ≠ ∅) do
 4.  begin
 5.      Pick up a transposition t in τ;
 6.      l := position of first symbol of t in P;
 7.      call BACKTRACK (P, t, l);
 8.      P := t(P); C* := C* ∪ Projection ∪ {P};
 9.      Projection := ∅; τ := τ(P) ∩ 𝒮
10.      Unmark all the permutations in C*;
11.  end
12. output(C*)
13. endmain
```

Comments. i. Different choices of t in line 5 of the main procedure may produce maximal consistent sets of different cardinality (the sets presented in Fig. 2 illustrate this point). We do not know of a good optimality criterion to be used at that point which guarantees that the obtained set C^* is of *maximum* cardinality. However, if this algorithm is used to check the maximality of a given connected consistent set, then any choice of t at line 5 will do.

ii. The consistent sets EXP (P), EXP′ (P), EXP″ (P) and EXP‴ (P) constructed by Abello [1] and discussed in Abello and Johnson [10] are generated by this algorithm when the transpositions t at line 5 are chosen appropriately. Each of these sets contains a skeleton which can be determined precisely by certain orderings of the set of adjacent transpositions which are required to transform certain permutation Q into its reverse Q^R. So all that is needed to do is to execute the algorithm with initial permutation Q following the path indicated by one of the skeletons (see the Appendix).

A conjecture. Let T_i be a maximum consistent set on a set S_i of i symbols and let T_{n-i} be a maximum consistent set on $\{1, 2, \cdots, n\} - S_i$ (the complement of S_i). The set $T_i * T_{n-i}$ formed by concatenating each i-permutation on T_i with every $(n-i)$-permutation in T_{n-i} is consistent because T_i and T_{n-i} are. If we assume that T_i and T_{n-i} contain a skeleton of lengths $\binom{i}{2}$ and $\binom{n-i}{2}$ respectively then we have that $T_i * T_{n-i}$ contains a sub-skeleton of length $\binom{i}{2}+\binom{n-i}{2}$. On the other hand, if there is a *connected* consistent set of greater cardinality than any other consistent set such a set may be decomposed into $\#_i$ equivalence classes each being of cardinality $\leqq |T_i| \times |T_{n-i}|$, in other words $|T_n| \leqq \#_i |T_i| \times |T_{n-i}|$. This being the case, a bound for $|T_n|$ is determined by a bound on $\#_i$ which is given by considering how many times $\binom{n}{2}$ contains $\binom{i}{2}+\binom{n-i}{2}$; and (wonder of wonders!) this number is less than or equal to $2+2/n-2$ for any i: $2 \leqq i \leqq n-2$.

Therefore $|T_n| \leqq (2+2/n-2)|T_i| \times |T_{n-i}|$. In particular if $i=2$ we have $|T_n| \leqq (2+2/n-2)2 \times |T_{n-2}|$,

$$|T_n| \leqq 4 \times |T_{n-2}| + (4/n-2)|T_{n-2}|. \tag{i}$$

For large n the fraction in the second term of (i) is insignificant, which together with the assumption that $|T_{n-2}| < 2^{n-2}$ will give us that

$$|T_n| < 2^n(1+\varepsilon).$$

This analysis together with the fact that all the known consistent sets are of cardinality less than 2^n suggest the following conjecture.

If M_n denotes the cardinality of a maximum consistent set, then $2^{n-1} < |M_n| < 2^n$ for $n \geqq 4$.

Appendix. *Two sample sets generated by the algorithm.*

EXP (P).
For $x \in \Sigma$, EXP $(x) \equiv x$.
For $P = p_1 p_2 \cdots p_n \in S_\Sigma$,

$$\text{EXP}(P) \equiv p_1 \text{EXP}(p_2 \cdots p_n) \cup p_n \text{EXP}(p_1 \cdots p_{n-1}).$$

Figure 7 illustrates EXP (P) with $P = 12345$.

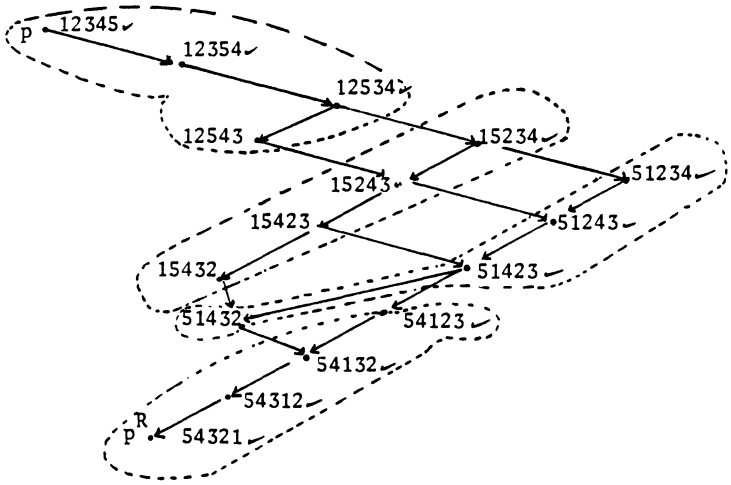


FIG. 7. *Graphical representation of* EXP (12345). • *Each arrow represents an adjacent transposition.* • $|\text{EXP}(12345)| = 16$ *and* $\text{LC}(12345) \subset \text{EXP}(12345)$. • *The elements of* LC (12345) *have a check mark at their right.* • *We have encircled those elements which have a common prefix of length two, namely* 12 EXP (345), 15 EXP (234), 51 EXP (234) *and* 54 EXP (123).

This set is generated by the MCCS algorithm if the transpositions t chosen at line 5 are those corresponding to the permutations which have a check mark at their right in Fig. 7. These permutations form a skeleton denoted here by LC(12345) (see [10]).

EXP$'''(P)$

$$\text{EXP}'''(P) \equiv \text{EXP}(P) \cup p_2 p_1 \text{EXP}(p_3 \cdots p_n) \cup p_{n-1} p_n \text{EXP}(p_1 \cdots p_{n-2}) - \{P, (p_2 p_1) p_3 \cdots p_n, p_{n-1} p_n (p_1 \cdots p_{n-2})^R\}.$$

Figure 8 illustrates EXP$'''(P)$ with $P = 12345$. This set is generated by the MCCS algorithm if the path marked with check marks is followed with initial permutation 21354.

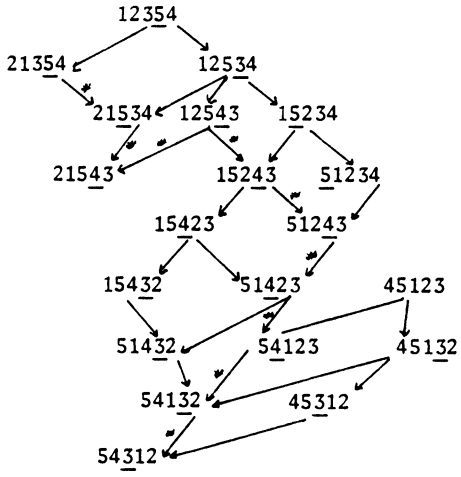


FIG. 8. • *A maximal consistent set of cardinality* $(3/2)2^{n-1}-4$. • *Here,* $n=5$, *so* $|\mathrm{EXP}'''(12345)|=20$.

Acknowledgments. Thanks to Drs. Eugene Johnsen, Charles Johnson and John Bruno for their helpful comments and valuable suggestions and to Ms. June Finney for the wonderful job typing the manuscript.

REFERENCES

[1] J. M. ABELLO, *Toward a maximum consistent set*, Technical report TRCS 11-81, Computer Science Dept, Univ. California, Santa Barbara, 1981.

[2] K. J. ARROW, *Social Choice and Individual Values*, John Wiley, New York, 1951.

[3] V. J. BOWMAN AND C. S. COLANTONI, *The extended Condorcet condition: A necessary and sufficient condition for the transitivity of majority decision*, J. Math. Soc., 2 (1972), pp. 267–283.

[4] P. C. FISHBUNN, *Conditions on preferences that guarantee a simple majority winner*, J. Math. Soc., 2 (1972), pp. 105–112.

[5] K. INADA, *The simple majority decision rule*, Econometrica, 37 (1969), pp. 491–506.

[6] C. R. JOHNSON, *Remarks on mathematical social choice*, Working paper 78–25, Dept. Economics and Institute for Physical Science and Technology, Univ. Maryland, College Park, 1978.

[7] J. S. KELLY, *Voting anomalies, The number of voters and the number of alternatives*, Econometrica, 42 (1974), pp. 239–251.

[8] D. KLAHR, *A computer simulation of the paradox of voting*, American Pol. Sci. Rev., 60 (1966), pp. 384–390.

[9] H. RAYNAUD, Technical report No. 331, Institute for Mathematical Studies in the Social Sciences, Stanford Univ., Stanford, CA, 1981.

[10] J. M. ABELLO AND CHARLES R. JOHNSON, *How large are transitive simple majority domains?*, this Journal, 5 (1984), pp. 603–618.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 1, January 1985

ON THE CHARACTERISTIC EQUATIONS OF THE CHARACTERISTIC POLYNOMIAL*

MILAN RANDIĆ†

Dedicated to Professor Danilo Blanuša, University of Zagreb

Abstract. The characteristic equations for a graph are defined as the system of equations for the coefficients of the characteristic polynomial. The construction of the equations is related to the method of Krylov, and the coefficients of these characteristic equations represent random walks of different length for pairs of vertices. Some properties of the characteristic equations, as revealed on illustrations, are discussed. These illustrations include trees on six vertices, isospectral graphs, and selected graphs having some unusual structural features.

AMS subject classification. 05C75

Introduction. The characteristic polynomial of a graph is defined as $\det(A - xI)$, where A is the adjacency matrix for the graph considered and I is the unit matrix of the same dimension. It represents one of the very important graph invariants. In the early development of quantum chemistry, the interaction matrix, the Hamiltonian, included only the nearest neighbor contributions; the characteristic polynomial played the role of the secular determinant [1]. This explains the continued interest of chemists in spectral properties of graphs even though Bloch's approximation [2] on the importance of the nearest neighbors no longer represents a viable mathematical model for computation of the electronic structure of molecules. Nevertheless, the characteristic polynomial emerges in some other chemical problems, such as the qualitative description of molecular orbitals, particularly their nodal characteristics, as related to Woodward–Hoffmann rules for cyclization in chemical reactions [3], [4]; for descriptions of numerous problems of chemical kinetics; as the starting point for construction of acyclic polynomials [5], [6] and other similar contraptions. Coulson [7] appears to be the first to recognize the role of selected subgraphs, enumeration of which gives the coefficients of the characteristic polynomial. More recently, several people have outlined graph theoretical constructions, showing that it suffices to use only K_2 and C_n as subgraphs (i.e., disjoint edges or disjoint cycles respectively). The most complete analysis is due to Sachs [8]. However, in applications the combinatorial explosion of terms associated with Sachs' method makes the scheme impractical already for graphs having less than a dozen vertices. With a rather pessimistic appraisal of computational difficulties associated with the construction of the characteristic polynomial, Harary, King, Mowshowitz and Read point out [9]: "... the calculation of characteristic polynomials for graphs of any size is usually extremely tedious ...". Hence, it is not surprising to see a revived interest in the challenging problem of construction of the characteristic polynomial. Modifications considering a vertex, edge or a ring removal reduced the amount of labor in many instances [10]–[15]. An elegant approach was described by Balasubramanian [16] in which the idea of pruning trees has been

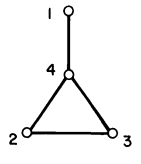
* Received by the editors July 6, 1983, and in final revised form March 10, 1984. This work was presented at the SIAM Second Conference on the Applications of Discrete Mathematics, held at Massachusetts Institute of Technology, Cambridge, Massachusetts, June 27–29, 1983.

† Department of Mathematics and Computer Science, Drake University, Des Moines, Iowa 50311, and Ames Laboratory, Iowa State University, Ames, Iowa 50011. Ames Laboratory is operated by Iowa State University for the U.S. Department of Energy under contract W-7405-ENG-82. The work was supported in part by the Office of the Director.

incorporated directly into a modified and reduced adjacency matrix, the approach which can be exteneded to cyclic graphs with pending bonds [17].

A need for a general approach which is computationally straightforward and bypasses the proliferation of components is clearly desirable, particularly since the topic of isospectral graphs, which has received attention in the mathematical [18], [19], physical [20], [21] and chemical [22]–[25], literature is still of interest and may continue to be so as long as the question of the complete characterization of isospectral graphs remains unresolved. It is then somewhat suprising that all the past efforts on construction of the characteristic polynomial have overlooked the so-called method of Krylov [26] for this construction. This particular approach appears well known in the area of numerical analysis of matrices and linear equations. A review article on indirect methods of expansion of secular determinants into algebraic equations was prepared already in 1945 by Weyland [27]. The method of Krylov essentially uses powers of the adjacency matrix which, when multiplied by a unit (column) vector, give equations for the coefficients of the characteristic polynomial. In a recent book [28] on the symmetric eigenvalue problem we read: "The idea of the power method is a very natural one. The civil engineers call it *Stodola's Iteration.* In (Krylov [26], 1931) the sequence, $(x, Ax, A^2x, \cdots)$ is actually used to find the coefficients of the characteristic polynomial and, despite that unfortunate goal, Krylov's name became securely attached to the sequence." In another book [29] on latent roots and latent vectors, there are further comments on Krylov's method. These comments point out the shortcomings in an example in which the full characteristic equation (polynomial) is not found. "In certain cases a different starting vector may yield the characteristic equation, but the uncertainty makes the method of little practical value," summarizes Hammerling [29] on the method. A good introduction to Krylov's method can also be found in Berezin and Zhidkov [30] and Gantmacher [31]. We will use essentially Krylov's method for computing the characteristic polynomial and will *extend* the approach by using all possible starting vectors. We will discuss the properties of a so-augmented system of equations which we will call *characteristic equations*, and will avoid referring to the characteristic polynomial equation $\mathrm{Ch}(x)=0$ as the characteristic equation. We will see that the characteristic equations have a useful role, not perhaps excluding possible practical value in numerical computations.

Brief outline of Krylov's method. Consider the graph G_1 with adjacency matrix A shown in Fig. 1.



G_1

$$A=\begin{bmatrix}0&0&0&1\\0&0&1&1\\0&1&0&1\\1&1&1&0\end{bmatrix}$$

FIG. 1. *A simple graph for illustration of the Cayley–Hamilton theorem.*

The characteristic polynomial in this case is

$$(-1)^n \det(A-xI)=x^4-4x^2-2x+1.$$

The factor $(-1)^n$ is introduced to make the leading term, x^n, positive. The Cayley–Hamilton theorem says that in this case $A^4-4A^2-2A+1=0$, that is, the adjacency matrix A satisfies its own characteristic polynomial. The above becomes a matrix equation which is satisfied for any element $a_{i,j}$ of the matrices A, A^2, A^4 and the corresponding i,j elements of the unity and zero matrices. The graph considered is

simple enough that one can easily verify the validity of the theorem. The required powers of A are:

$$A^2=\begin{bmatrix}1&1&1&0\\1&2&1&1\\1&1&2&1\\0&1&1&3\end{bmatrix}, \qquad A^4=\begin{bmatrix}3&4&4&2\\4&7&6&6\\4&6&7&6\\2&6&6&11\end{bmatrix}.$$

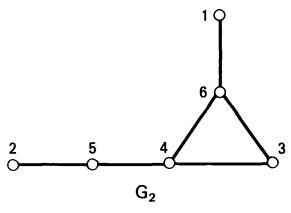
The Cayley–Hamilton theorem then becomes:

$$(A^4)_{i,j}-4(A^2)_{i,j}-2(A)_{i,j}+(1)_{i,j}=0,$$

and should hold for any i, j. For instance, for $i=2$, $j=1$, the 2, 1 elements of the matrices A^4, A^2 and A are, respectively, 4, 1, and 0; thus we see that the above relation is satisfied.

We will consider now a somewhat generalized form of the Cayley–Hamilton theorem. Consider the following products: AX, $A(AX)$, $A(A(AX))$, etc, in which we select a vector (column X with the components $x_1, x_2, x_3, \cdots, x_n$) and make repeated products with the matrix A on derived vectors. We can write the result of the first multiplication as X' (i.e., $AX=X'$), the result of the second multiplication as X'' (i.e., $AX'=X''$) and so on. These are equivalent to the products AX, A^2X, A^3X etc., but we need not have various powers A^n, because we can use X', X'' etc. with A rather than constructing A^2 and higher powers of A to be used with the initially selected column-vector X. The basis for our approach is the fact that vectors A^kX also satisfy the Cayley–Hamilton theorem; since they can be easily derived and considered known they allow us to determine the unknown coefficients of the characteristic polynomial which also appear in the Cayley–Hamilton theorem. For the graph G_1 the characteristic polynomial is: $x^4+a_2x^2+a_3x+a_4$ with $a_2=-4$, $a_3=-2$ and $a_4=1$. The Cayley–Hamilton theorem becomes $A^4+a_2A^2+a_3A+a_4=0$. Formally, we see that by multiplying the above matrix equation (from the right) by the column vector X, we derive a valid vector equation: $A^4X+a_2A^2X+a_3AX+a_4X=0$. Since A is known and X can be chosen at will, we see that we can generate a system of linear equations for the coefficients a_i which when solved, allows one to write the characteristic polynomial. Solving linear equations poses no problem; thus the suggested procedure is computationally practical while conceptually simple. Observe the difference: in the usual eigenvalue problem, one diagonalizes the secular determinant and finds eigenvalues and eigenvectors *without* explicitly obtaining the characteristic polynomial; in the present approach one solves equations for the coefficients which define the secular equation. The information on the characteristic polynomial is of interest *per se* as it allows the study of various structural features which are reflected in the magnitudes of the coefficients. This is particularly of interest when families of structurally related systems are considered [32], [33].

An example. We will illustrate some details of the procedure on the graph G_2 in Fig. 2.



$$\begin{bmatrix}0&0&0&0&0&1\\0&0&0&0&1&0\\0&0&0&1&0&1\\0&0&1&0&1&1\\0&1&0&1&0&0\\1&0&1&1&0&0\end{bmatrix}$$

FIG. 2. *The smallest graph with all vertices being nonequivalent.*

In this particular graph all vertices are nonequivalent. The characteristic polynomial is of degree six: $x^6+a_1x^5+a_2x^4+a_3x^3+a_4x^2+a_5x+a_6$. It is known that for graphs: (a) $a_1=0$; and (b) $a_2=-$(the number of edges). Graph G_2 is simple enough that one can determine by inspection a few other coefficients, but in order to illustrate the procedure we will assume that the coefficients are not known. Since X can be chosen at will, we may take a vector with all but one component zero. As will be seen, this will lead to fewer computations in constructing the vectors X', X'', etc. Hence we assume X to be a column vector with $x_1=1$ and all other $x_i=0$. The column vectors X, AX, $A(AX)$) make an array A^kX with n rows and $n+1$ columns:

$$\begin{bmatrix} 1 & 0 & 1 & 0 & 3 & 2 & 12 \\ 0 & 0 & 0 & 0 & 1 & 1 & 6 \\ 0 & 0 & 1 & 1 & 4 & 7 & 19 \\ 0 & 0 & 1 & 1 & 5 & 7 & 25 \\ 0 & 0 & 0 & 1 & 1 & 6 & 8 \\ 0 & 1 & 0 & 3 & 2 & 12 & 16 \end{bmatrix}.$$

Each row of the array A^kX defines one of the linear equations for the coefficients of the characteristic polynomial because the elements of each row satisfy the Cayley–Hamilton theorem when extended to vector multiplication. Therefore, in this case we obtain the following set of linear equations for the coefficients a_i:

(A) $12+2a_1+3a_2+a_4+a_6=0,$

(B) $6+2a_1+3a_2=0,$

(C) $19+7a_1+4a_2+a_3+a_4=0,$

(D) $25+7a_1+5a_2+a_3+a_4=0,$

(E) $8+6a_1+a_2+a_3=0,$

(F) $16+12a_1+2a_2+3a_3+a_5=0.$

Strictly, a_0 appears as the factor of the constant terms and the above would then correspond to a system of homogenous linear equation, which could then be transformed into the above set by assuming $a_0=1$. The above equations can be simplified by little manipulations. Thus (D)–(C) gives $6+a_2=0$, which determines a_2. Substitution of this result into (B) gives a_1, and with a_1 and a_2 known one can determine a_3 from (E); other coefficients are then easily found. The solution is

$$a_1=0, \quad a_2=-6, \quad a_3=-2, \quad a_4=7, \quad a_5=2, \quad a_6=1,$$

and the characteristic polynomial for G_2 is

$$x^6-6x^4-2x^3+7x^2+2x-1.$$

Since for all graphs $a_1=0$ and $a_2=-$(number of edges), we can assume these known results, view the set of equations as somewhat redundant, and speed up the solution process.

The solution to the system of equations, the coefficients a_i, are integers, because they represent a result of *enumeration* of qualified subgraphs. However, given a set of equations, like the equations (A)–(F), it is by no means obvious that the solution in integers exists, so an interesting question can be raised [34] on the characterization of conditions that ensure solutions in integers for a system of linear equations. Because

the proposed approach for computing the characteristic polynomial leads to a system of linear equations, one has to be concerned with the possibility that the system of equations is incomplete. In principle, a system of equations may be inconsistent, but because the characteristic polynomial *exists* we need not consider that issue. Incompleteness is a result of linear dependence between the equations or appearance of the same equation more than once. As will be seen, in cases of graphs with symmetry we will obtain duplicate equations and the situation may arise of having an insufficient number of equations to determine the coefficients.

The complete set of characteristic equations. The system of equations for the coefficients a_k in the previous section were sufficient for obtaining the solution. However, the array $A^k X$ with X the column vector $(1, 0, 0, 0, 0, 0)$ is only one of many such possible arrays that can be generated by vector multiplications we have outlined. In particular, one can consider all vectors of the form $(\cdots, 0, 1, 0, \cdots)$ with 1 at different rows of the column.

In Table 1 we show the remaining arrays for the graph G_2. Because the entries in each row of an $A^k X_i$ (i indicating the nonzero position in the unit column vectors) represent the count of walks of length k between the vertex i and the vertex j, where j is the row j of the $A^k X_i$ arrays, one sees that each row of the arrays $A^k X_i$ summarizes the random walks of different lengths for a pair of vertices (i, j). In a graph with n vertices there are $\frac{1}{2}n(n+1)$ such pairs ((i, i) also has to be included, representing self-returning walks of length k). Hence in the case of G_2 with 6 vertices, we have in all 21 different pairs, and therefore at most 21 different equations. These are listed in Table 2, together with the assignment of vertices (i, j). As one sees on inspection, most of the derived equations are different, at least for the case considered. Only the equations (1, 5) and (2, 6) are duplicate, indicating equinumerosity for all walks of the corresponding length between the two pairs of nonequivalent vertices.

The collection of nonduplicate equations from the possible total of $\frac{1}{2}n(n+1)$ will be called the *characteristic system or characteristic equations of a graph.* The concept of characteristic equation is more general than the concept of characteristic polynomial, in the sense that the latter, as a rule, will be contained in the former (provided that in the case of deficiency we can find additional conditions which yield a_i coefficients). Importantly, the characteristic equations *contain more information* about a graph. Even in the case of a restricted number of such equations we have the information that many pairs of vertices produce the same count for random walks.

A trivial reason for such an occurrence is equivalence of vertices (vertices belong to the same orbits of the automorphism group). But as will be seen, there are other, nontrivial, situations that require a better understanding. Already for the graph G_2, which was selected for illustration because it is the smallest graph having no equivalent vertices (identity being the only symmetry operation), we see that the pair (1, 5) yields the same count of random walks as the pair (2, 6). Why? We hope to partially answer this question and similar ones for more general cases. In order to clarify these situations we will first examine the characteristic equations for acyclic graphs (trees) on six vertices, as they offer a fair number of typical coincidental counts of walks for nonequivalent pairs of vertices.

Before proceeding with the results for selected graphs, let us point to some properties of the arrays $A^k X_i$ of Table 1 and the characteristic equations of Table 2. Observe that in a few instances the second column of an array $A^k X_i$ has all zeros except for single 1 entry and thus the second column can be considered as another unit column vector X_j. For example, in the case of the first array $A^k X_1$, we find that

TABLE 1

The arrays A^kX_i for graph G_2.

X_1							X_2						
1	0	1	0	3	2	12	0	0	0	0	1	1	6
0	0	0	0	1	1	6	1	0	1	0	2	0	6
0	0	1	1	4	7	19	0	0	0	1	1	5	8
0	0	1	1	5	7	25	0	0	1	0	4	2	17
0	0	0	1	1	6	8	0	1	0	2	0	6	2
0	1	0	3	2	12	16	0	0	0	1	1	6	8
X_3							X_4						
0	0	1	1	4	7	19	0	0	1	1	5	7	25
0	0	0	1	1	5	8	0	0	1	0	4	2	17
1	0	2	2	8	14	39	0	1	1	4	7	20	41
0	1	1	4	7	20	41	1	0	3	2	13	16	62
0	0	1	1	5	8	25	0	1	0	4	2	17	18
0	1	1	4	7	19	41	0	1	1	5	7	25	43
X_6							X_5						
0	1	0	3	2	12	16	0	0	0	1	1	6	8
0	0	0	1	1	6	8	0	1	0	2	0	6	2
0	1	1	4	7	19	41	0	0	1	1	5	8	25
0	1	1	5	7	25	43	0	1	0	4	2	17	18
0	0	1	1	6	8	31	1	0	2	0	6	2	23
1	0	3	2	12	16	56	0	0	1	1	6	8	31

TABLE 2

The characteristic equations of graph G_2 extracted from A^kX_i arrays.

(A)	$a_6 + a_4 + 3a_2 + 2a_1 + 12 = 0$	(1, 1)
(B)	$a_2 + a_1 + 6 = 0$	(1, 2)
(C)	$a_4 + a_3 + 4a_2 + 7a_1 + 19 = 0$	(1, 3)
(D)	$a_4 + a_3 + 5a_2 + 7a_1 + 21 = 0$	(1, 4)
(E)	$a_3 + a_2 + 6a_1 + 8 = 0$	(1, 5)
(F)	$a_5 + 3a_3 + 2a_2 + 12a_1 + 16 = 0$	(1, 6)
(G)	$a_6 + a_4 + 2a_2 + 6 = 0$	(2, 2)
(H)	$a_3 + a_2 + 5a_1 + 8 = 0$	(2, 3)
(I)	$a_4 + 4a_2 + 2a_1 + 17 = 0$	(2, 4)
(J)	$a_5 + 2a_3 + 6a_1 + 2 = 0$	(2, 5)
(*)	$a_3 + a_2 + 6a_1 + 8 = 0$	(2, 6)
(K)	$a_6 + 2a_4 + 2a_3 + 8a_2 + 14a_1 + 39 = 0$	(3, 3)
(L)	$a_5 + a_4 + 4a_3 + 7a_2 + 20a_1 + 41 = 0$	(3, 4)
(M)	$a_4 + a_3 + 5a_2 + 8a_1 + 25 = 0$	(3, 5)
(N)	$a_5 + a_4 + 4a_3 + 7a_2 + 19a_1 + 41 = 0$	(3, 6)
(O)	$a_6 + 3a_4 + 2a_3 + 13a_2 + 16a_1 + 62 = 0$	(4, 4)
(P)	$a_5 + 4a_3 + 2a_2 + 17a_1 + 18 = 0$	(4, 5)
(Q)	$a_5 + a_4 + 5a_3 + 7a_2 + 25a_1 + 43 = 0$	(4, 6)
(R)	$a_6 + 2a_4 + 6a_2 + 2a_1 + 23 = 0$	(5, 5)
(S)	$a_4 + a_3 + 6a_2 + 8a_1 + 31 = 0$	(5, 6)
(T)	$a_6 + 3a_4 + 2a_3 + 12a_2 + 16a_1 + 56 = 0$	(6, 6)

the second column represents the unit column vector X_6. Consequently, the array A^kX_6 is identical to A^kX_1 if shifted by one column to the right. This can be used in computing A^kX_6 to reduce the amount of computation by simply continuing to calculate A^kX_1 for one more iteration. Normally one computes A^kX_i for $k=0, 1, 2, \cdots, n$ but now one should also find the column corresponding to $k=n+1$.

In Table 2 we include equations involving all the coefficients (assuming $a_0=1$), even though we know that in the case of graphs $a_1=0$. A number of equations can produce this result trivially: cf. the equations (E) and (H) for instance, or (L) and (N) and the pair (D) and (N). By incorporating the information on $a_1=0$ we would reduce the number of different equations from the present 20 to 17, but since we are primarily interested in the equations, rather than in their solution (the characteristic polynomial) we may keep all the different equations for the purpose of characterization of a graph. Immediately a number of questions may be asked: Is the characterization unique? Can a graph be reconstructed from a known system of equations? What is the meaning of duplicate equations, and why do they appear?

Characteristic equations for trees with $n=6$ vertices. In order to answer the questions posed above, it is instructive to consider a number of selected examples. We start by examining the characteristic equations for trees (acyclic graphs) on $n=6$ vertices. Because of the bipartitie character of trees we report only the characteristic equations for the coefficients a_{2k}, since necessarily all a_{2k+1} $(k=0, 1, 1, 2)$ are, in this case, zero. The results are shown in Table 3. Graphs G_3-G_8 are illustrated in Fig. 3 where an arbitrary labeling of vertices has been assumed. We have included *all* possible equations; duplicates *within* a graph are indicated by (*) and duplicate equations between different graphs have been indicated as (**).

Inspection of Table 3 points to some regularities in the characteristic equations. The simplest equation, (c): $a_2+5=0$, which corresponds to the well-known property

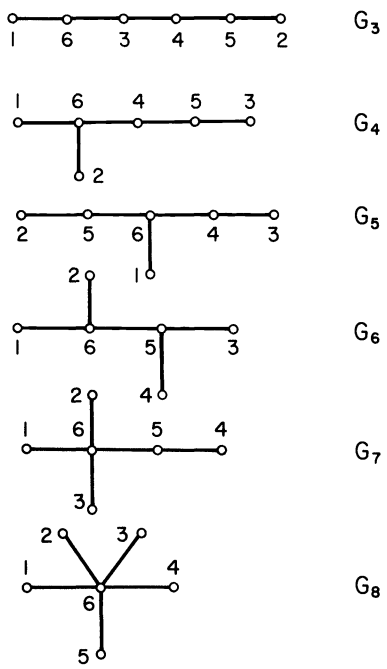


FIG. 3. *Trees on $n=6$ vertices and labeling of their vertices.*

of the characteristic polynomial that a_2 counts the number of edges in a graph, appears in all cases that have chains of length 4. Similarly, all graphs that have an edge of the type [35] (1, 3) have two equations with all corresponding coefficients identical except for a_6 which is either 1 or 0 (cf. equations (g) and (h)). Hence all such graphs necessarily

TABLE 3

The characteristic equations for trees with $n=6$ vertices. A duplicate equation within a graph is marked as (*), the occurrence of a same equation in different members is shown as (**).

G_3:	(a)	$a_6+a_4+2a_2+5=0$	(1, 1)	
	(b)	$a_4+3a_2+9=0$	(1, 3)	
	(c)	$a_2+5=0$	(1, 5)	
	(d)	$a_6+2a_4+6a_4+19=0$	(3, 3)	
	(e)	$a_4+4a_2+14=0$	(3, 5)	
	(f)	$a_6+2a_4+5a_2+14=0$	(5, 5)	
G_4:	(g)	$a_6+a_4+3a_2+10=0$	(1, 1)	
	(h)	$a_4+3a_2+10=0$	(1, 2)	
	(i)	$a_4+4a_2+15=0$	(1, 4)	
	(**)	$a_2+5=0$	(1, 3)	cf. (c)
	(**)	$a_6+a_4+2a_2+5=0$	(3, 3)	cf. (a)
	(*	$a_4+3a_2+10=0$	(3, 4)	cf. (h)
	(j)	$a_6+2a_4+7a_2+24=0$	(4, 4)	
	(k)	$a_6+2a_4+5a_2+15=0$	(5, 5)	
	(l)	$a_4+5a_2+20=0$	(5, 6)	
	(m)	$a_6+3a_4+10a_2+35=0$	(6, 6)	
G_5:	(a)	$a_6+a_4+3a_2+11=0$	(1, 1)	
	(**)	$a_4+4a_2+15=0$	(1, 4)	
	(o)	$a_6+2a_4+6a_2+21=0$	(4, 4)	
	(**)	$a_4+5a_2+20=0$	(4, 5)	cf. (l)
	(p)	$a_6+a_4+2a_2+6=0$	(2, 2)	
	(**)	$a_2+5=0$	(2, 3)	cf. (c)
	(*), (**)	$a_4+4a_2+15=0$	(2, 6)	cf. (i)
G_6:	(**)	$a_6+a_4+3a_2+11=0$	(1, 1)	cf. (n)
	(q)	$a_4+3a_2+11=0$	(1, 2)	
	(r)	$a_4+5a_2+21=0$	(1, 5)	
	(s)	$a_6+3a_4+11a_2+43=0$	(5, 5)	
G_7:	(t)	$a_6+a_4+4a_2+17=0$	(1, 1)	
	(u)	$a_4+4a_2+17=0$	(1, 2)	
	(v)	$a_4+5a_2+22=0$	(1, 5)	
	(w)	$a_6+2a_4+7a_2+29=0$	(5, 5)	
	(x)	$a_6+a_4+2a_2+7=0$	(4, 4)	
	(*)	$a_4+5a_2+22=0$	(4, 6)	cf. (v)
	(y)	$a_6+4a_4+17a_2+73=0$	(6, 6)	
G_8:	(z)	$a_6+a_4+5a_2+25=0$	(1, 1)	
	(¢)	$a_4+5a_2+25=0$	(1, 2)	
	($)	$a_6+5a_4+25a_2+125=0$	(6, 6)	

have an eigenvalue $x=0$, another well-known result of spectral graph theory. The highest coefficient a_6 appears only in the equations corresponding to count of self-returning walks (i.e., those having label (i, i)). The edge type $(1, 3)$ or in general (m, n) indicates the valencies of the two vertices i, j making the edge [35].

Another important observation is that the number of distinctive equations appears to depend strongly on the symmetry properties of the graphs: fewer equivalence classes, fewer equations. In the case of the star graph G_8 we have only three equations for the three coefficients, but since $a_6=0$ we see that the equations are linearly dependent (proportional). The case illustrates one of the limitations of Krylov's method, which in this case gives only two equations:

$$a_6=0, \qquad a_4+5a_2+25=0.$$

Without additional information or some modification of the procedure in this case we cannot determine the characteristic polynomial.

The case indicates a potential difficulty that may occur in highly symmetrical graphs, but the fact that we have obtained only three different equation (from 21 theoretically possible equations) is in itself *information*, and the question to consider is *how* can this kind of information be used to derive the characteristic polynomial. Elsewhere we will discuss this problem in more detail [36], so let us only point out that in such cases one should use the information on the *equivalence* and introduce appropriate linear combinations of vertex labels that will factor the adjacency matrix. Then we may proceed to construct arrays B^kY_i analogous to the arrays A^kX_i but corresponding to subspaces of the factored adjacency matrix. As a result, one obtains a system of equations for each factor and, consequently, the factors of the characteristic polynomial consistent with the automorphism group of the graph.

Isospectral graphs. The first critical test for the characteristic equations as being unique comes from comparison of such equations for isospectral graphs. These are graphs that have identical characteristic polynomials and thus represent a potential case for identical sets of characteristic equations. If graphs are not isospectral then they will necessarily differ in at least one equation, since one of the equations has to account for a different root factor $(x-x_i)$. But since the characteristic equations contain more information than the characteristic polynomial, it may happen that isospectral graphs have different characteristic equations. As we will see, indeed this appears to be a rule, rather than exception. The characteristic polynomial, which represents the expanded secular determinant in the well-known Hückel MO method,[1] is related to random walks, the relationship which has been pointed out and discussed by Marcus [39]. The coefficients of the x^{n-j} term of the characteristic polynomial can be derived from known counts of self-returning walks of length j. The intimate relationship is also reflected in the equivalence of the information given by a characteristic polynomial and spectral moments.[2] In comparison, the characteristic equations, besides incorporating the same information on self-returning walks also contain information on the count of random walks (non self-returning walks). The number of non self-returning walks is generally much greater than the number of self-returning walks, thus with the abundance of structural information concealed in a collection of all random walks, rather than using only self-returning walks and limited structural information involved

[1] The earliest recognition of the mathematical equivalence of the Hückel MO method and the graph theory eigenvalue problem appears in [37], [38].

[2] Spectral moments are defined as $M^k=\sum_i x_i^k$ or alternatively, because of the invariant properties of the trace of A^k of adjacency matrix, as $M^k=\mathrm{Tr}\,A^k$. For more see the papers of Baker and Fisher [20], [21].

there, it appears much less likely to encounter so many coincidences that are typical for characteristic polynomials.

In Table 4 we list the characteristic equations for a simplest pair of acyclic isospectral graphs, which immediately confirms our expectations. Not only are the collection of the characteristic equations different, but in this particular instance *none* of the equations are the same! Graphs are shown in Fig. 4.

TABLE 4

The characteristic equations for a pair of isospectral graphs.

G_9:	$a_8 + a_6 + 4a_4 + 19a_2 + 97 = 0$	(1, 1)
	$a_6 + 4a_4 + 19a_2 + 97 = 0$	(1, 2)
	$a_6 + 7a_4 + 40a_2 + 217 = 0$	(1, 7)
	$a_8 + 4a_6 + 19a_4 + 97a_2 + 508 = 0$	(7, 7)
G_{10}:	$a_8 + a_6 + 5a_4 + 26a_2 + 137 = 0$	(1, 1)
	$a_6 + 5a_4 + 26a_2 + 137 = 0$	(1, 2)
	$a_4 + 7a_2 + 40 = 0$	(1, 5)
	$a_6 + 6a_4 + 33a_2 + 177 = 0$	(1, 6)
	$a_8 + a_6 + 2a_4 + 5a_2 + 17 = 0$	(5, 5)
	$a_6 + 3a_4 + 12a_2 + 57 = 0$	(5, 6)
	$a_8 + 2a_6 + 9a_4 + 45a_2 + 234 = 0$	(6, 6)
	$a_8 + 2a_6 + 5a_4 + 17a_2 + 74 = 0$	(7, 7)
	$a_6 + 7a_4 + 40a_2 + 177 = 0$	(7, 8)
	$a_8 + 5a_6 + 26a_4 + 137a_2 + 725 = 0$	(8, 8)

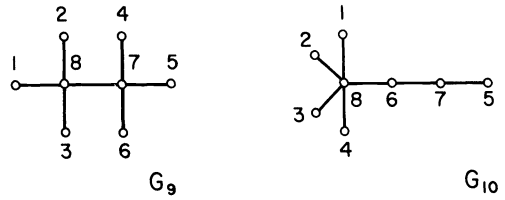


FIG. 4. *The smallest isospectral trees.*

We have examined a fair number of isospectral graphs and will report our finding elsewhere [40]. The preliminary results confirm that, as a rule, one can expect *different* characteristic equations for isospectral graphs although in numerous cases one finds the same equation in two isospectral graphs. But the latter is true also in case of graphs which are not isospectral, so it should not be viewed as an alarming sign. With increasing symmetry, however, the number of equations is reduced considerably, as can already be seen with graph $K_{1,5}$ (G_8); hence the chance should not be overlooked that for highly regular and symmetric graphs we may have isospectral pairs with the same collection of characteristic equations, should not be overlooked. In fact, as Schwenk confirmed [41], highly regular transitive graphs with the same characteristic equations exist, they are illustrated in Schwenk's paper on spectral reconstruction problems [42]. Not only do these graphs have the same characteristic polynomials, but their subgraphs obtained by deleting vertex or by deleting an edge are also isospectral.

Another pair of highly regular graphs which are isospectral and whose study may provide interesting results is the pair of graphs of Fig. 5, considered by Fisher [21].

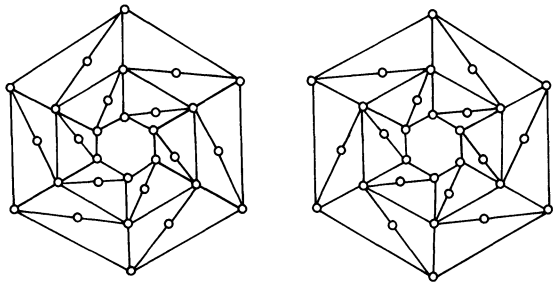


FIG. 5. *Isospectral graphs of Fisher.*

As Fink and Morris [43] observed, here both graphs have the same collection of vertex codes, where "code" represents the history of the path connectedness of a vertex i and is defined to be a sequence $p_1, p_2, p_3, \cdots, p_k, \cdots$, with $p_k = (A^k)_{ii}$. Another such pair of graphs has been identified by Slater, and the topic expanded by Quintas and Slater; see [44]–[46].

Endospectral graphs. Graphs of the type shown in Fig. 6 we call *endospectral* (the name proposed here for the first time, from the Greek *endo* meaning *within* or *inner.* The first such graph G_{11}, was studied by Schwenk [23], who demonstrated that there are two symmetry nonequivalent vertices whose removal produces disconnected subgraphs with the property that the characteristic polynomial for the system is the same,

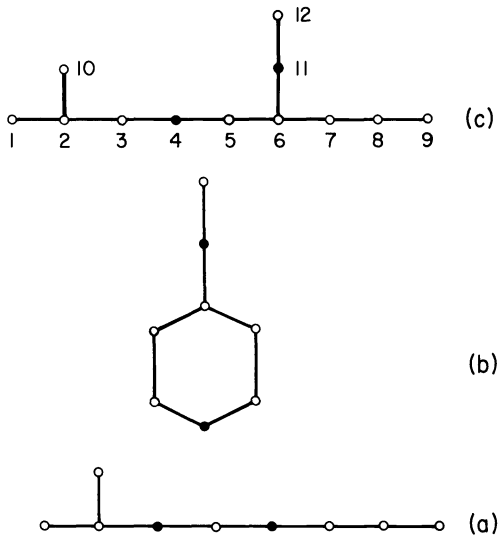


FIG. 6. *Endospectral graphs.*

regardless which of the two vertices has been removed. The consequences of this is that one may attach to the two special vertices suitably named (by Herndon [24] as *isospectral points* (nodes or vertices)) any fragment F and construct, in this way, an isospectral pair of graphs. Another such endospectral graph is G_{12}, which has been identified in the chemical literature [22], [24] as the source of the isospectrality of well-known chemical graphs of divinylbenzene and 2-phenylbutadiene (see Fig. 7).

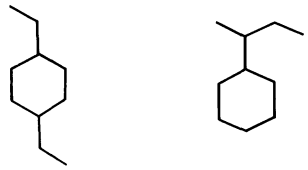


FIG. 7. *Molecular graphs of divinylbenzene and phenylbutadiene.*

It is of interest to mention that over 40 years of chemical studies of such molecules by the simple Hückel MO method isospectrality among molecules was unknown until 1973, when Živković [48][3] discerned the above pair as such. This is somewhat embarrassing, particularly because Živkovic discovered the above two molecules as isospectral by examining well-known tables. Coulson and Streitwieser's compilation, *Dictionary of pi-Electron Calculations* [49], published in 1965, contains eigenvalues and eigenvectors for some 350 molecular structures. The two isospectral molecules *p*-divinylbenzene and 2-phenylbutadiene are tabulated only 5 pages apart!

In the area of chemical documentation occurrence of isospectral graphs had been discussed by Balaban and Harary in 1971 [50]. For the purposes of chemical documentation the work of Balaban and Harary doomed the use of the characteristic polynomial as hazardous and useless, even though today we may be somewhat less pessimistic *if* we are willing to supplement use of the characteristic polynomial with information on structurally related systems [32]. If one uses the characteristic equations rather than the polynomial itself, we may resurrect spectral graph theory as a viable basis for characterization of graphs; the topic will be elaborated elsewhere [51]. Finally a word of caution—the mathematical isospectrality and associated properties and the actual molecule pi-electron spectroscopy should not be confused, even though the simple MO method suggests that the two are the same. The Hückel MO method has been known for too long a time to be *deficient* in predictions of molecular spectral properties. Hence the report of Heilbronner and Jones [52] on differences in the actual spectra of divinylbenzene and 2-phenylbutadiene was hardly warranted as a demonstration of the limitations of the Hückel MO approach (see [53]). This can be established already by examining the spectrum of a *single* molecule. Actual photo-electron spectra are of interest per se as a source for testing other theoretical models, but the conclusion that isospectral graphs are of no relevance to chemistry only fuels confusion about chemical graph theory among less informed readers.

The last of the three endospectral graphs shown in Fig. 6 has been found by the present author [15]. The isospectral points (i.e., those which when erased still leave the same characteristic polynomial) 4 and 11 have the same count of self-returning walks: 2, 6, 22, 88, 365, $\cdots$. Higher walks necessarily have to coincide, because of the Cayley–Hamilton theorem, which can be viewed as a recursive relation. In Table 5 we list *all walks* for the two isospectral points. As one sees, the two isospectral points can nevertheless be distinguished: while the sequences of self-returning walks are the same for the vertices, the sequences representing random walks are different. The situation can be contrasted with that associated with the so called *unusual walks* [54], which represent a coincidental count of self-returning walks in two *different* graphs, which need not be even isospectral (see the next section). The results in Table 5 are

[3] The manuscript of Živkovic was submitted to Croatica Chemica Acta but never appeared. It was subsequently resurrected and enlarged with the collaboration of Trinajstić and Randić and appeared in Molecular Physics [22].

TABLE 5

Selected equations for the endospectral graph corresponding to the isospectral vertices (4) *and* (11).

one step

(4, 3):	1,	3,	11,	43,	173,	708,
(4, 5)	1,	3,	11,	43,	192,	831,
(11, 6)	1,	4,	16,	66,	277,	1174,
(11, 12)	1,	2,	6,	22,	88,	363,

two steps

(4, 2)	0,	1,	5,	21,	85,	343,	1394,
(4, 6)	0,	1,	5,	23,	104,	466,	2074,
(11, 5)	0,	1,	5,	22,	95,	410,	1773,
(11, 7)	0,	1,	5,	22,	94,	399,	1695,

three steps

(4, 1)	0,	1,	5,	21,	85,	343,
(4, 7)	0,	1,	6,	30,	141,	644,
(4, 11)	0,	1,	6,	29,	133,	599,
(11, 8)	0,	1,	6,	28,	122,	521,

Combined walks:

$(4, 3) + (4, 5) = (11, 6) + (11, 12)$

2, 6, 22, 88, 365, 1539,

$(4, 2) + (4, 6) = (11, 5) + (11, 7)$

0, 2, 10, 44, 189, 809, 3468,

$(4, 1) + (4, 10) + (4, 7) + (4, 11) + 3(4, 3) + 3(4, 5) = (11, 4) + (11, 8) + 4(11, 6) + 2(11, 12)$

6, 22, 84, 365, 1539, 6546

combined in the lower part of the table by adding all walks of length 1, then adding separately all walks of length 2, length 3, etc. for each of the two isospectral vertices.

As one sees, the combined results produce sequences which are the same, whether we consider walks originating at vertex 4 or vertex 11. Thus the regularity observed for unusual vertices hold here also, if self-returning walks are considered and weighting of walks is taken properly into account. Moreover, we find that the same regularity holds also for random walks (i.e., walks originating at one vertex but not necessarily ending at the same vertex), and we will see that the same is true for unusual vertices (vide infra). The weighting is determined by the number of ways one can walk from i to j. For instance, in the case of the pairs of vertices (4, 3) and (4, 5) and count of walks of length 3 we have weighting factor 3 (there being three walks of length 3 between vertices 4 and 3 or 4 and 5). However, in the case of vertex 11 there are four walks between 11 and 6 and only two walks between 11 and 12, hence the corresponding factors are 4 and 2 respectively.

In summary, endospectral graphs have different counts of random walks for the isospectral points. However, properly combined, such sequences of counts of random walks produce for the isospectral points the same resulting overall sequence.

Unusual walks. In Fig. 8 we show graphs having unusual walks. If for two vertices walk sequences W_1 and W_2 are equal, then the corresponding vertices are said to have *equipotent walks* [54]. If two nonequivalent vertices have equipotent walks, then we call them unusual walks. The vertices have also been called *isocodal* [55] in view of the synonymous use of the terms code and sequence (when no confusion results). Table 6 lists several of the characteristic equations associated with the two graphs of

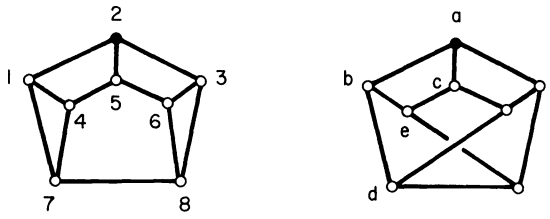


FIG. 8. *A pair of graphs with unusual walks.*

TABLE 6

Unusual random walks in graphs of Fig. 8.

(2, 1)	0,	1,	0,	6,	4,	45,	56,	358,	616,	(a, b)
(2, 5)	0,	1,	0,	7,	2,	51,	42,	396,	532,	(a, c)
(2, 7)	0,	0,	1,	3,	10,	30,	91,	273,	820,	(a, d)
(2, 4)	0,	0,	2,	1,	16,	16,	126,	189,	1024,	(a, e)
(2, 2)	1,	0,	3,	0,	19,	10,	141,	154,	1109,	(a, a)

Fig. 7. We report only the equations corresponding to the unusual vertices. The labelling of vertices for the two graphs has been selected so that the characteristic equations have common labels. Thus the occurrence of unusual graphs allows one to induce numbering of vertices in one of the graphs when labels in the other have been selected (arbitrarily). In the case of equations having (i, i) labels, one obtains the sequence of self-returning walks, which reproduces the observation previously known for these graphs. In fact the unusual graphs have been detected by examination of the counts of self-returning walks. However, the novelty here reported is that the regularity to hold for self-returning walks is also true for random walks, if one of the vertices is the special vertex.

Discussion. Recently Powers and Sulaiman [56] considered walk partitions of vertices in a graph in relation to the coloration of a graph. Their sequences $(e, Ae, A^2e, \cdots, A^{k-1}e)$, where A^le for $l \geqq 0$ is a list of the number of walks of length l starting from each of the vertices, and is equivalent (except for notation) to our arrays $A^k X_i$. They detected isospectral points, as these would partition into the same class but the partitioning would not coincide with an orbit partition. Similarly, Ellzey and Davis [57] in their approach to detection of the automorphism of graphs came across points (atoms in a molecular graph) which would not be differentiated initially. The partitioning "process" is perturbed; nonequivalent vertices eventually do separate into different orbits.

All these more recent investigations clearly show that many graphs, and frequently relatively simple graphs, possess intriguing structural features that we have overlooked in the past or have not investigated thoroughly. The emphasis in different works is different: coloration in the work of Powers and Sulaiman and automorphism in the work of Davis and Ellzey. Our emphasis is on the *novel concept*, the characteristic equations, while the early work of Krylov, Stodola, and possibly others rediscovering the approach was more concerned with numerically solving a matrix eigenvalue problem. Observe that Krylov considered general matrices with real elements, in contrast to binary matrices with zero and ones as elements. Possibly the lack of interest in graph theory for more general matrices is a reason that the work of Krylov, or that of Frame [58], was generally overlooked, despite an otherwise intense interest in graph

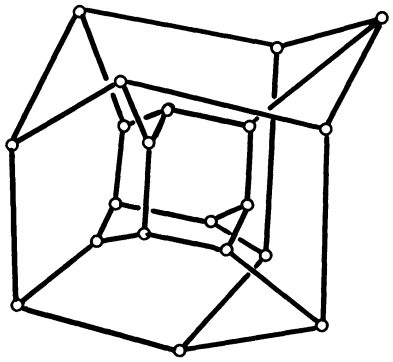


FIG. 9. *A well-known graph depicted differently (showing tetrahedral symmetry).*

spectra [59]. Our interest is primarily with the characteristic equations as such. It appears that such interest and concern may lead to novel directions in reviving the quest for graph *isomorphism* and graph *recognition* in particular. Here we *distinguish* between the two. The graph isomorphism problem is well known and has received considerable attention [60]–[62]. We call the graph recognition problem one in which there is a *single graph* to be considered, and the question is, can the graph be identified with those previously known. For example, is the graph of Fig. 9 some known graph just depicted differently?[4]

But even if some may consider the problem of graph isomorphism as resolved,[5] the search for new ways and possibly simpler ones will remain, and the characteristic equations appear to offer some new avenues to this old problem.

Let us point to one important aspect of the collection of characteristic equations. Suppose we consider a graph with n vertices, which can have $n!$ different ways of labeling. In contrast there are at most $\frac{1}{2}n(n+1)$ different equations, hence much fewer comparisons to be made to verify that two different adjacency matrices of $n \times n$ size represent the same graph. In case of highly regular graphs we may have too few equations, however, and this approach may not be suitable in such instances. Excluding these rare "pathological" cases we see how an apparently $n!$ problem may eventually be circumvented and possibly shown not to be so complex, even perhaps permitting an algorithm which is polynomial in respect to n.

It yet remains to be seen under which circumstances one can expect the above approach to be used as a test for isomorphism, but it certainly has eliminated most of isospectral graphs as obstacles to such applications. The new counterexamples form a less populous class of "isopotent" graphs, graphs characterized by the same set of characteristic equations, the first members of the new class being the two isospectral graphs of Schwenk, having 16 vertices, each of degree six. Use of A^kX_i arrays for resolving the isomorphism problem appears attractive: it possesses all the elegance and simplicity initially hoped for when the characteristic polynomial was considered

[4] The graph is known as Desargues–Levi, it is closely related to the well-known Petersen graph, as pointed out by A. T. Balaban in [63]. The particular pictorial form shown was suggested by this author in [64], where more familiar alternatives are shown, due to H. S. M. Coxeter (Univ. Toronto), K. Mislow (Chemistry, Princeton Univ.) and others. We may point out that the particular graph is the first graph in the chemical literature in which vertices and edges are not related to atoms and bonds but rather to molecules (isomers) and rearrangement routes (reactions) (see [65]). The graph was subsequently called Balaban's graph, but on an initiative of Balaban the name "Desargues–Levi" has been suggested, approved (Coxeter) and accepted (Mislow).

[5] For example, through the use of computer techniques.

as a possibility for such a task; yet even being so intimately connected to the characteristic polynomial, the characteristic equations are more powerful, devoid of frequent coincidental situations. The problem deserves closer scrutiny before being recommended for such application. Much may depend on our ability to better understand the occurrence of highly specialized cases (like the two graphs of Schwenk) that do not qualify for such analysis. This task is outside the scope of the present manuscript.

Acknowledgments. Correspondence with Professor A. Schwenk (Annapolis, Maryland) is greatly appreciated. Professor K. Balasubramanian (Tempe, Arizona) drew my attention to a brief outline of the expansion of the secular determinant in powers of the matrix in [66], which in turn lead me to the article of Weyland [27], to which Professor E. Bright Wilson kindly directed me. Finally, I also wish to thank Professor Balaban (Bucharest, Roumania) for correspondence clearing up some details about the name of the Desargues–Levi graph (see footnote 4).

REFERENCES

[1] C. A. COULSON, *Valence*, Oxford Univ. Press, London, 1961.
[2] F. BLOCH, *Uber die Quantenmechanik der Elektronen in Kristallgittern* Z. Für Phys., 52 (1929) pp. 555–600 (in particular p. 563).
[3] R. HOFFMANN AND R. B. WOODWARD, *The conservation of orbital symmetry*, Acc. Chem. Res., 1 (1968) pp. 17–22.
[4] R. B. WOODWARD AND R. HOFFMANN, *Die Erhaltung der Orbitalsymmetrie*, Angew. Chem., 81 (1969) pp. 797–869. English; *The conservation of orbital symmetry*, Angew. Chem. International Edit., 8 (1969) pp. 781–853.
[5] I. GUTMAN, M. MILUN AND N. TRINAJSTIĆ, *Graph theory and molecular orbitals*, 18, *On topological resonance energy*, Croat. Chem. Acta, 48 (1976), pp. 87–95.
[6] J. AIHARA, *A new definition of Dewar-type resonance energies*, J. Amer. Chem. Soc., 98 (1976) pp. 2750–2758.
[7] C. A. COULSON, *Notes on the secular determinant in molecular orbital theory*, Proc. Cambridge Phil. Soc., 46 (1950), pp. 202–205.
[8] H. SACHS, *Beziehungen zwischen den in einem Graphen enthaltenten Kreisen und seinem charakteristischen Polynom*, Publ. Math. (Debrecen) 11 (1964), pp. 119–134.
[9] F. HARARY, C. KING, A. MOWSHOWITZ AND R. C. READ, *Cospectral graphs and digraphs*, Bull. London Math. Soc., 3 (1971), pp. 321–328.
[10] A. GRAOVAC AND I. GUTMAN, *The determinant of the adjacency matrix of a molecular graph*, Croat. Chem. Acta, 51 (1978), pp. 133ff.
[11] Y. S. KIANG, *Calculation of the determinant of the adjacency matrix and the stability of conjugated molecules*, Int. J. Quant. Chem., 18 (1980), pp. 331–338.
[12] G. S. YAN, *The graph theoretical formulas for determinant expansions*, Int. J. Quant. Chem. Symp., 14 (1980), pp. 549–555.
[13] Q. ZHANG, L. LIN AND N. WANG, *Graphical method of the Hückel matrix*, Sci. Sin., 22 (1979), pp. 1169–1184.
[14] H. HOSOYA, *Graphical enumeration of the coefficients of the secular polynomials of the Hückel molecular orbitals*, Theor. Chim. Acta, 25 (1972), pp. 215–222.
[15] M. RANDIĆ, *On evaluation of the characteristic polynomial for large molecules*, J. Comput. Chem., 3 (1982), pp. 421–435.
[16] K. BALASUBRAMANIAN, *Spectra of chemical trees*, Int. J. Quant. Chem., 21 (1982), pp. 581–590.
[17] K. BALASUBRAMANIAN AND M. RANDIĆ, *The characteristic polynomials of structures with pending bonds*, Theor. Chim. Acta, 61 (1982), pp. 307–323.
[18] L. A. COLATZ AND U. SINOGOWITZ, *Spektren endlicher Grafen*, Abh. Math. Sem. Univ. Hamburg, 21 (1957), pp. 63–77.
[19] F. HARARY, *The determinant of the adjacency matrix of a graph*, SIAM Rev., 4 (1962), pp. 202–210.
[20] G. A. BAKER, JR., *Drum shapes and isospectral graphs*, J. Math. Phys., 7 (1966), pp. 2238–2242.
[21] M. E. FISHER, *On hearing the shape of a drum*, J. Comb. Theory, 1 (1966), pp. 105–125.
[22] T. ŽIVKOVIĆ, N. TRINAJSTIĆ AND M. RANDIĆ, *On conjugated molecules with identical topological spectra*, Mol. Phys., 30 (1975), pp. 517–533.

[23] M. RANDIĆ, N. TRINAJSTIĆ AND T. ŽIVKOVIĆ, *Molecular graphs having identical spectra*, J. Chem. Soc. Faraday Trans. II, 72 (1976), pp. 244–256.
[24] W. C. HERNDON, *Isospectral molecules*, Tet. Letters, 8 (1974), pp. 671–674.
[25] W. C. HERNDON AND M. L. ELLZEY, JR., *Isospectral graphs and molecules*. Tetrahedron, 31 (1975), pp. 99–107.
[26] A. N. KRYLOV, Izv. Akad. Nauk SSSR, Ser. 7, Fiz.-Mat., 4 (1931), pp. 491ff. (In Russian.)
[27] H. WEYLAND, *Expansion of determinantal equations into polynomial form*, Quart. Appl. Math., 2 (1945), pp. 277–306.
[28] B. N. PARLETT, *The Symmetric Eigenvalue Problem*, Prentice-Hall, Englewood Cliffs, NJ, 1980.
[29] S. J. HAMMARLING, *Latent Roots and Latent Vectors*, Univ. Toronto Press, Toronto, Ontario, 1970.
[30] I. S. BEREZIN AND N. P. ZHIDKOV, *Computing Methods*, vol. II, Pergamon Press, Oxford, 1965, Chapter 8.
[31] F. R. GANTMACHER, *Matrix Theory*, Chelsea, New York, 1964.
[32] M. RANDIĆ, *On alternative forms of the characteristic polynomial and the problem of graph recognition*, Theor. Chim. Acta, 62 (1983), pp. 485–498.
[33] H. HOSOYA AND M. RANDIĆ, *Analysis of the topological dependency of the characteristic polynomial in its Chebyshev expansion*, Theor. Chim. Acta, 63 (1983), pp. 473–495.
[34] W. L. WOODWORTH, (Drake University), private communication, 1982.
[35] M. RANDIĆ, *On characterization of molecular branching*, J. Amer. Chem. Soc., 97 (1975), pp. 6609–6615.
[36] M. RANDIĆ, G. IZMIRLIAN AND K. BALASUBRAMANIAN (work in progress).
[37] HS. H. GÜNTHARD AND H. PRIMAS, *Zusammenghung von Graphentheorie und* MO-*Theorie von Molekeln mit Systemen konjugierter Bindungen*, Helv. Chim. Acta, 39 (1956), pp. 1645–1653.
[38] K. RUEDENBERG, *Quantum mechanics of mobile electrons in conjugated systems* III. *Topological matrix as generatrix of bond orders*, J. Chem. Phys., 34 (1961), pp. 1884–1891.
[39] R. A. MARCUS, *Additivity of heats of combustion*, LCAO *resonance energies and bond orders of conformal sets of conjugated compounds*, J. Chem. Phys., 43 (1976), p. 2643.
[40] M. RANDIĆ, A. F. KLEINER AND W. L. WOODWORTH, to be published.
[41] A. J. SCHWENK (Annapolis, Maryland), private correspondence, 1982.
[42] ———, *Spectral reconstruction problems*, Ann. New York Acad. Sci., 328 (1979), pp. 183–189.
[43] A. M. FINK AND W. MORRIS (Ames, Iowa), private information.
[44] P. J. SLATER, *Counterexample to Randić's conjecture on distance degree sequences for trees*, J. Graph Theory, 6 (1982), pp. 89–92.
[45] L. V. QUINTAS AND P. J. SLATER, *Pairs of non-isomorphic graphs having the same path degree sequence*, MATCH, 12 (1981), pp. 75–86.
[46] P. J. SLATER, *The origin of extended degree sequences of graphs*, preprint, 1983.
[47] A. J. SCHWENK, *Almost all trees are cospectral*, in New Directions in the Theory of Graphs, F. Harary, ed., Academic Press, New York, 1973, pp. 275–307.
[48] T. ŽIVKOVIĆ, reported at the Quantum Chemistry School, Leningrad, USSR, December 1973.
[49] C. A. COULSON AND A. STREITWEISER, JR. (with extensive help from M. D. Poole and J. I. Brauman) *Dictionary of pi-Electron Calculations*, W. H. Freeman, San Francisco 1965.
[50] A. T. BALABAN AND F. HARARY, *The characteristic polynomial does not uniquely determine the topology of a molecule*, J. Chem. Docum., 11 (1971), pp. 258–259.
[51] M. RANDIĆ, J. Chem. Inf. & Comput. Sci., submitted.
[52] E. HEILBRONNER AND T. B. JONES, *Spectral differences between "Isospectral" molecules*, J. Amer. Chem. Soc., 100 (1978), pp. 6506–6507.
[53] H. HOSOYA, *Topological index, a newly proposed quantity characterizing the topological nature of structural isomers of saturated hydrocarbons*, Bull. Chem. Soc. Japan, 44 (1971), pp. 2332–2339.
[54] M. RANDIĆ, W. L. WOODWORTH AND A. GRAOVAC, *Unusual random walks*, Int. J. Quant. Chem., 24 (1983), pp. 435–452.
[55] J. V. KNOP, W. R. MÜLLER, K. SZYMANSKI, M. RANDIĆ AND N. TRINAJSTIĆ, *Note on acyclic structures and their self-returning walks*, Croat. Chem. Acta, 56 (1983), pp. 405–409.
[56] D. L. POWERS AND M. M. SULAIMAN, *The walk partition and colorations of a graph*, Linear Algebra Appl., 48 (1982), pp. 145–159.
[57] M. L. ELLZEY, JR. AND M. I. DAVIS, *A technique for determining the symmetry properties of molecular graphs*, J. Comput. Chem., 4 (1983), pp. 267–275.
[58] K. BALASUBRAMANIAN, *The use of Frame's method for the characteristic polynomials of chemical graphs*, preprint.
[59] D. M. CVETKOVIĆ, M. DOOB AND H. SACHS, *Spectra of Graphs—Theory and Applications*, Deutscher Verlag der Wiss., Berlin, 1980.
[60] R. C. READ AND D. G. CORNEIL, *The graph isomorphism disease*, J. Graph Theory, 1 (1977), pp. 339–363.

[61] G. GATI, *Further annotated bibliography on the isomorphism disease*, J. Graph Theory, 3 (1979), pp. 95–109.

[62] C. J. COLBOURN, Technical Report No. 123/78, Department of Computer Science, Univ. Toronto, Toronto, Ontario, 1978.

[63] A. T. BALABAN, *Chemical graphs.* XIX, *intramolecular isomerizations of trigonal-bipyramidal structures with five different ligands*, Rev. Roum. Chim., 18 (1973), pp. 855–862.

[64] M. RANDIĆ, *Symmetry properties of graphs of interest in chemistry*, II, *Desargues-Levi graphs*, Int. J. Quant. Chem., 15 (1979), pp. 663–682.

[65] A. T. BALABAN, D. FARCASIU AND R. BANICA, *Graphs of multiple* 1,2*-shifts in carbonium ions and related systems*, Rev. Roum. Chim., 11 (1966), pp. 1205–1212.

[66] E. B. WILSON, J. C. DECIOUS AND P. C. CROSS, *Molecular Vibrations*, McGraw-Hill, New York, 1965.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 1, January 1985

015

OPTIMAL SET PARTITIONING*

F. K. HWANG†, J. SUN‡ AND E. Y. YAO§

Abstract. We consider the problem of partitioning a set of elements into unlabeled subsets to minimize cost, where the cost of a partition is essentially the sum of costs contributed by the component subsets. We give several results which specify conditions on the cost functions such that there always exists an optimal partition which is an "ordered partition" (an optimal ordered partition can be determined in quadratic time). We also give several applications to illustrate the usefulness of our results.

1. Introduction. The problem of finding a minimum "cost" partition of a given set of elements arises in many applications. One such model which has recently been studied [1],[2],[3],[4],[5] is:

$$\text{minimize } F(P) = \sum_{i=1}^{K} f(S_i)$$

where $P = S_1 \cup S_2 \ldots \cup S_K$ is a partition of a given set Z of n real numbers: $Z = \{z_1, \ldots, z_n\}$, $f: 2^Z \to R$ is a function defined on the power set of Z.

When K is arbitrary, we refer to the above problem simply as the *partition problem.* When K is fixed, we refer to the problem as the *K-partition problem.* A more special case is that not only K, but the set $M = \{|S_1|, \ldots, |S_k|\}$, which is called the *shape* of P, is also fixed, it is then referred to as the *shape-partition problem.*

Define an ordered partition to be one in which for any two subsets S_i and S_j, either no number in S_i exceeds any number in S_j, or vice versa. We say that f has the property $OP = OOP$ if there always exists an optimal partition which is ordered. Hwang [4] noted that an optimal ordered partition can always be determined in $O(n^2)$ time by a straightforward application of dynamic programming and studied certain classes of f functions for which $OP = OOP$. Chakravarty, Orlin and Rothblum [2] considered the K-partition problems. They gave an elegant result for $OP = OOP$ and showed that an optimal ordered K-partition can be obtained in $O(n^2 \cdot K)$ time.

It should be noted that we can prove $OP = OOP$ for a K-partition problem by proving it for an arbitrary shape partition since a K-partition must assume some shape. Similarly, we can prove $OP = OOP$ for a partition problem by proving it for an arbitrary K-partition. This is the approach used in [2] and [4] and will be continued in this paper. It should also be noted that all our results apply to the case

*Received by the editors August 1, 1983, and in revised form February 1, 1984. This work was presented at the SIAM Second Conference on the Applications of Discrete Mathematics, held at Massachusetts of Technology, Cambridge, Massachusetts, June 27-29, 1983. This paper was typeset at Bell Laboratories, Murray Hill, New Jersey, using the **troff** program running under the Unix™ operating system. Final copy was produced on April 12, 1984.

†AT&T Bell Laboratories, Murray Hill, New Jersey 07974.
‡University of Washington, Seattle, Washington 98195.
§Zhejiang University, Hangzhou, Zhejiang, China.

that $F(P) = L(\Sigma_{i=1}^{K} f(S_i))$ where L is a nondecreasing function which can depend on K for K-partition and on shape for shape-partition.

2. Some definitions. Let G be a set function defined on subsets of Z. G is said to be *minimum−(maximum)−ordered* if for arbitrary S_1 and S, $S_1 \subseteq S \subseteq Z$, there always exists an ordered partition S_1', $S\backslash S_1'$ such that

$$G(S_1') + G(S\backslash S_1') \leqslant (\geqslant)\ G(S_1) + G(S\backslash S_1) \quad \text{with} \quad |S_1'| = |S_1| .$$

A minimum-ordered function is called minimum-ordered-*proper* if $\min\{G(S_1'), G(S\backslash S_1')\} \leqslant \min\{G(S_1), G(S\backslash S_1)\}$ for all $S_1 \subseteq S \subseteq Z$, $|S_1'| = |S|$, and *improper* if the inequality is reversed, Similarly, we can define maximum-ordered-proper and improper by interchanging the two words "minimum" and "maximum" and reversing the inequality. Note that G is minimum-ordered-proper (improper) if and only if $-G$ is maximum-ordered-proper (improper).

A proper function can be further divided into short-proper or long-proper depending on whether it is always the shorter or the longer set achieving the minimum (or maximum). This short or long property is very useful for shape partitions for then the optimal ordered partition is explicit once the z_i's are ordered (otherwise we have to compare $K!/\Pi_j(m_j!)$ ordered partition where m_j is the multiplicity of size j in the shape).

3. Minimum-(maximum)-ordered functions. The following lemma was proved in [2],[4].

LEMMA 1. *Suppose that the problem is to minimize (maximize)* $F(P) = \Sigma_{i=1}^{K} f(S_i)$ *for a shape-partition. Then* $OP = OOP$ *if* f *is a minimum−(maximum)−ordered function. We now state a self-evident but useful result.*

THEOREM 1 (*the relabeling theorem*). *Suppose that* $OP = OOP$ *for* f. *Let* f' *be obtained from* f *by replacing every number* $z \in S$ *by the number* $g(z)$ *where* g *is an arbitrary function. Then* $OP = OOP$ *for* f' (*the ordering is on* $g(z)$).

A set function G is said to be *nondecreasing* (*nonincreasing*) if for any $z_1 \in S \subseteq Z$ and $z_2 \in Z$, $z_1 \leqslant z_2$, $G(S) \leqslant (\geqslant)\ G(S \cup \{z_2\}\backslash\{z_1\})$.

THEOREM 2 (*the separation theorem*). *Suppose that*

$$f(S_1) + f(S\backslash S_1) - f(S_1') - f(S\backslash S_1') = [H(S_1\backslash\{z_i\}) - H(S\backslash S_1\backslash\{z_j\})]G(z_i, z_j)$$

where H *is nonincreasing (nondecreasing),* $G(z_i, z_j)$ *preserves the sign of* $(z_i - z_j)$, *and* $\{S_1', S\backslash S_1'\}$ *is obtained from* $\{S_1, S\backslash S_1\}$ *by interchanging* $z_i \in S_1$ *and* $z_j \in S\backslash S_1$. *Then* $f(S)$ *is minimum−(maximum)−ordered.*

Proof. Let $\{S_1, S\backslash S_1\}$ be an optimal but not ordered partition. x_1 and $x_{|S_1|}$ denote the smallest and largest element in S_1, y_1 and $y_{|S\backslash S_1|}$ denote the smallest and largest in $S\backslash S_1$. Then $x_1 < y_{|S\backslash S_1|}$ and $y_1 < x_{|S_1|}$. Now

$$0 \geqslant f(S_1) + f(S\backslash S_1) - f(S_1 \cup \{y_{|S\backslash S_1|}\}\backslash\{x_1\}) - f(S\backslash S_1 \cup \{x_1\}\backslash\{y_{|S\backslash S_1|}\})$$

$$= [H(S_1\backslash\{x_1\}) - H(S\backslash S_1\backslash\{y_{|S\backslash S_1|}\})]\ G\ (x_1, y_{|S\backslash S_1|}) .$$

Since $G(x_1, y_{|S\setminus S_1|}) < 0$ we have $H(S_1\setminus\{x_1\}) - H(S\setminus S_1\setminus\{y_{|S\setminus S_1|}\}) \geqslant 0$. Thus, letting $S_1' = S_1 \cup \{y_1\}\setminus\{x_{|S_1|}\}$, we have

$$0 \geqslant f(S_1) + f(S\setminus S_1) - f(S_1') - f(S\setminus S_1')$$

$$= [H(S_1\setminus\{x_{|S_1|}\}) - H(S\setminus S_1\setminus\{y_1\})]\, G\ (x_{|S_1|}, y_1)$$

$$\geqslant [H(S_1\setminus\{x_1\}) - H(S\setminus S_1\setminus\{y_{|S\setminus S_1|}\})]\, G\ (x_{|S_1|}, y_1) \geqslant 0\ ,$$

i.e., we can always interchange y_1 with $x_{|S_1|}$ without affecting optimality. Eventually, we obtain an optimal ordered partition.

A function $g(x, y)$ is called an *interval function* if g is increasing in $\max\{x, y\}$ and decreasing in $\min\{x, y\}$.

THEOREM 3. *Suppose that* $f(S) = \Sigma_{z\in S}\, g(z, u)$ *where* g *is an interval function and* u *satisfies* $\Sigma_{z\in S}\, g(z, u) = \min_v \Sigma_{z\in S}\, g(z, v)$. *Then OP = OOP for the general or K-partition problems with* $F(P) = \Sigma_{i=1}^{K} f(S_i)$ *to be minimized.*

Proof. Consider arbitrary S_1, S satisfying $S_1 \subseteq S \subseteq Z$. Let u_1 and u_2 be defined in

$$\sum_{z\in S_1} g(z, u_1) = \min_v \sum_{z\in S_1} g(z, v)\ ,$$

$$\sum_{z\in S\setminus S_1} g(z, u_2) = \min_v \sum_{z\in S\setminus S_1} g(z, v)\ .$$

Suppose that $\{S_1, S\setminus S_1\}$ is optimal but not ordered. If $u_1 = u_2$, then there exists $x \neq u_1$. Now

$$f(S_1)+f(S\setminus S_1) = \sum_{z\in S} g(z, u_1) > \sum_{z\in S\setminus\{x\}} g(z, u_1)+g(x, x) \geqslant f(S\setminus\{x\})+f(\{x\})\ ,$$

a contradiction to the optimality of $\{S, S\setminus S_1\}$. If $u_1 \neq u_2$, assume without loss of generality that $u_1 < u_2$. Then there exist $x \in S_1$ and $w \in S\setminus S_1$ with $x > w$. Furthermore, at least one of the following two inequalities is true:

$$g(w, u_1) < g(w, u_2)\ ,$$

$$g(x, u_2) < g(x, u_1)\ .$$

Without loss of generality, assume the first inequality is true. Define

$$S_1' = S_1 \cup \{w\}\ ,$$

$$\sum_{z\in S_1'} g(z, u_1') = \min_v \sum_{z\in S_1'} g(z, v)\ ,$$

$$\sum_{z\in S\setminus S_1'} g(z, u_2') = \min_v \sum_{z\in S\setminus S_1'} g(z, v)\ .$$

Then

$$G(S_1) + G(S\backslash S_1) = \sum_{z \in S_1} g(z, u_1) + \sum_{z \in S\backslash S_1} g(z, u_2)$$

$$> \sum_{z \in S_1'} g(z, u_1) + \sum_{z \in S\backslash S_1'} g(z, u_2)$$

$$\geqslant \sum_{z \in S_1'} g(z, u_1') + \sum_{z \in S\backslash S_1'} g(z, u_2') = G(S_1') + G(S\backslash S_1') ,$$

a contradiction to the optimality of the partition $\{S_1, S\backslash S_1\}$. Hence $OP = OOP$.

Remark. Unfortunately this result is not applicable for the shape-partition case because the shape of the partition is changed when rearranging the elements of S_1 and $S\backslash S_1$. We now show that a stronger condition on G will extend Theorem 3 to shape-partition.

An interval function is called *super-additive* if it satisfies the additional condition that for x and y both lying in the range (z, w), $g(z, w)+g(x, y) \geqslant g(x, z)+g(y, w)$.

COROLLARY. *If g is super-additive, then f is minimum-ordered.*

Proof. Let x, w, u_1, u_2 be defined as before $(u_1 \leqslant u_2)$. Then by superadditivity

$$g(x, u_1)+g(w, u_2) \geqslant g(x, u_2)+g(w, u_1) .$$

Therefore, analogous to Theorem 3, we have

$$G(S_1)+G(S\backslash S_1) \geqslant G(S_1')+G(S\backslash S_1') .$$

Suppose that $u_1' \leqslant u_2'$. Then we can keep on interchanging larger elements into $S\backslash S_1$ and eventually obtain an ordered optimal partition. We now prove $u_1' \leqslant u_1 \leqslant u_2 \leqslant u_2'$.

Suppose the contrary that there exists a $u_1' > u_1$ such that

$$\sum_{z \in S_1\backslash\{x\}} g(z, u_1')+g(w, u_1') < \sum_{z \in S_1\backslash\{x\}} g(z, u_1)+g(w, u_1) .$$

But by the definition of u_1,

$$\sum_{z \in S_1\backslash\{x\}} g(z, u_1)+g(x, u_1) \leqslant \sum_{z \in S_1\backslash\{x\}} g(z, u_1')+g(x, u_1') .$$

Adding up both sides, we obtain

$$g(w, u_1')+g(x, u_1) < g(w, u_1)+g(x, u_1')$$

a contradiction to the superadditivity of g. Similarly, we can prove $u_2 \leqslant u_2'$. The proof is complete.

We give another result on super-additive functions.

THEOREM 4. *Let S denote the set $\{z_1 \leqslant \ldots \leqslant z_t\}$ and g a super-additive interval function. Then $f(S) = \sum_{i=1}^{t-1} g(z_i, z_{i+1})$ is minimum-ordered, for $t \geqslant 2$.*

Proof. Let S_1 be the set $\{x_1 \leqslant x_2 \leqslant \ldots \leqslant x_\ell\}$ and let $S\backslash S_1$ be the set $\{y_1 \leqslant y_2 \leqslant \ldots \leqslant y_m\}$. Furthermore, let $S = \{z_1 \leqslant z_2 \leqslant \ldots \leqslant z_{\ell+m}\}$ be the elements in S. Without loss of generality, assume $z_1 = x_1$. We show that $(S_1, S\backslash S_1)$ cannot

be an optimal partition of S if $x_\ell > y_1$. Define $S_1' = \{z_1, z_2,..., z_\ell\}$ and $S/S_1' = \{z_{\ell+1}, z_{\ell+2},..., z_{\ell+m}\}$.

Let (u_1, u_2), (u_3, u_4),..., (u_{2i-1}, u_{2i}), $u_1 \geqslant u_2 \geqslant ... \geqslant u_{2i}$, denote the adjacent pairs in S_1(u_{2k} and u_{2k+1} can denote the same x element) not adjacent in S. Similarly, let (v_1, v_2), (v_3, v_4),..., (v_{2j-1}, v_{2j}) $v_1 \geqslant v_2 \geqslant ... \geqslant v_{2j}$ denote the adjacent pairs in $S\backslash S_1$ (v_{2k} and v_{2k+1} can denote the same y element) not adjacent in S. Without loss of generality, assume $u_1 \geqslant v_1$. Then necessarily,

$$j + 1 \geqslant i \geqslant j ,$$

$$u_k \geqslant v_k \quad \text{for } k = 1, 2,..., j, \text{ and}$$

$$v_k \geqslant v_{k+2} \quad \text{for } k = 1, 2,..., i-1 .$$

Furthermore, (v_{2k-1}, u_{2k}) and (u_{2k+1}, v_{2k}) for all k above are adjacent in S. Therefore

$$\begin{aligned} f(S_1) + f(S\backslash S_1) - f(S_1') - f(S\backslash S_1') \\ = \sum_{k=1}^{i} g(u_{2k-1}, u_{2k}) + \sum_{k=i}^{j} g(v_{2k-1}, v_{2k}) + g(z_\ell, z_{\ell+1}) \\ - \sum_{k=1}^{i} g(u_{2k-1}, v_{2k-2}) - \sum_{k=1}^{j} g(v_{2k-1}, u_{2k}) - (i-j)g(v_{2j+1}, u_{2j+2}) \\ - (j+1-i)g(u_{2i+1}, v_{2i}) , \end{aligned}$$

where v_0, v_{2j+1} and u_{2i+1} are defined to be the z elements adjacent to u_1, u_{2j+2} and v_{2i}, respectively, in S, and $u_1 \geqslant v_0$, $v_{2j+1} > u_{2j+1}$, $u_{2i+1} \geqslant v_{2i}$. Note that the last two terms in the equation contain a single g term since i is either j or $j+1$. Without loss of generality, assume that $u_{2h-1} \leqslant z_\ell \leqslant z_{\ell+1} \leqslant u_{2h}$. We have

$$\begin{aligned} g(u_{2k-1}, u_{2k}) &\geqslant g(u_{2k-1}, v_{2k-2}), \quad k = 1, 2,..., h-1 , \\ &\geqslant g(v_{2k-1}, u_{2k}), \quad k = h+1,..., i , \\ g(v_{2k-1}, v_{2k}) &\geqslant g(v_{2k-1}, u_{2k}), \quad k = 1, 2,..., h-1 , \\ &\geqslant g(u_{2k+1}, v_{2k}), \quad k = h,...,j , \end{aligned}$$

and

$$g(u_{2h-1}, u_{2h}) + g(z_\ell, z_{\ell+1}) \geqslant g(u_{2h-1}, v_{2h-2}) + g(v_{2h-1}, u_{2h}) .$$

Therefore

$$f(S_1) + f(S\backslash S_1) - f(S_1') - f(S\backslash S_1') \geqslant 0 .$$

4. Composition functions. Suppose that G is ordered. Let HG be the composition of G and H. We would like to know under what conditions that HG is also ordered.

THEOREM 5. *The following relations between G, H and HG are valid.*

G / H / HG	minimum-proper	minimum-improper	maximum-proper	maximum-improper
convex nondecreasing		minimum-improper	maximum-proper	
convex nonincreasing	maximum-proper			minimum-improper
concave nondecreasing	minimum-proper			maximum-improper
concave nonincreasing		maximum-improper	minimum-proper	

Proof. We prove only for the case that G is minimum-proper and H is concave nondecreasing since the other cases are similar.

Consider S_1 and S such that $S_1 \subseteq S \subseteq Z$. Since G is minimum-proper

$$G(S_1) + G(S\backslash S_1) = G(S_1') + G(S\backslash S_1') + d \quad \text{for some } d \geqslant 0 .$$

Without loss of generality, assume

$$G(S_1') \leqslant \min\{G(S_1), G(S\backslash S_1)\} .$$

By using the concave and nondecreasing property of H, we have

$$HG(S_1) + HG(S\backslash S_1) \geqslant HG(S_1') + H[G(S\backslash S_1') + d]$$

$$\geqslant HG(S_1') + HG(S\backslash S_1') .$$

Hence HG is minimum-proper.

COROLLARY. *The word "proper" in Theorem 5 can be replaced by "short-proper" ("long proper").*

If we strengthen the condition for G, then the condition for H can be weakened. The following theorem is a straight-forward extension of an elegant result of Chakravarty, Orlin and Rothblum [2].

THEOREM 6. *Suppose that $H_{|S|}G$ is concave (convex) in y for every fixed $|S|$ and $G(S) = \sum_{z\in S} g(z)$. Then $H_{|S|}G$ is minimum−(maximum)−ordered.*

Chakravarty, Orlin and Rothblum proved the case for $g(z) = z$. Theorem 6 follows immediately from Theorem 1.

5. Examples.

Example 1. A system reliability problem. Consider a system consisting of k parallel components where component i is a series combination of n_i elements. The problem is to assign $n = \Sigma_{i=1}^{k} n_i$ elements with working probabilities $q_1,\ldots, q_n$ to the k

components to maximize the system reliability.

We may define the cost of a partition as the probability of system down, i.e.,

$$F(P) = \prod_{i=1}^{k} f(S_i) = \prod_{i=1}^{k} \left(1 - \prod_{j=1}^{n_i} q_j\right).$$

We make the log transformation (which is nondecreasing) to obtain the standard form

$$\log F(P) = \sum_{i=1}^{k} \log f(S_i) = \sum_{i=1}^{k} \log \left(1 - \prod_{j=1}^{n_i} q_j\right).$$

Since

$$f(S_i) = 1 - \prod_{j=1}^{n_i} q_j$$

is easily verified to be minimum-short-proper and log is a concave nondecreasing function, by the corollary of Theorem 5, $\log f(S_i)$ is minimum- short-proper. Hence the most reliable system is obtained by assigning the n_1 most reliable elements to component 1, the next n_2 most reliable elements to component 2, and so on, assuming $n_1 \leqslant n_2 \leqslant \ldots \leqslant n_k$.

Example 2. Symmetric functions. We first consider the power mean case

$$f(S) = |S| \left(\sum_{z \in S} z^r\right)^{1/s}.$$

Define $g(z) = z^r$, $H_{|S|}(y) = |S| y^{1/s}$. For $s \leqslant 1$, $H_{|S|}$ is convex. Hence f is minimum-ordered in $g(z)$ by Theorem 6. For $s \geqslant 1$, $H_{|S|}$ is concave. Then f is maximum-ordered. The special case $r = 1$ and $s = \frac{1}{2}$ have been studied in [1],[3].

Next we consider the y^{th}-order cross product case. Define

$$C_y(S) = \sum_{\substack{Y \subseteq S \\ |Y|=y}} \prod_{z \in Y} z.$$

Let

$$f(S) = |S|^c C_y(S).$$

Then for $S_1 \subseteq S$, $z_i \in S_1$, $z_j \in S_2 = S \backslash S_1$,

$$\begin{aligned} & f(S_1) + f(S\backslash S_1) - f(S_1') - f(S\backslash S_1') \\ &= |S_1|^c[C_y(S_1-\{z_i\}) + x_i C_{y-1}(S_1-\{z_i\})] + |S_2|^c[C_y(S_2-\{z_j\}) + x_j C_{y-1}(S_1-\{z_j\})] \\ &\quad - |S_1|^c[C_y(S_1-\{z_i\}) + x_j C_{y-1}(S_1-\{z_i\})] - |S_2|^c[C_y(S_2-\{z_j\}) + x_i C_{y-1}(S_2-\{z_j\})] \\ &= [|S_1|^c C_{y-1}(S_1-\{z_i\}) - |S_2|^c C_{y-1}(S_2-\{z_j\})](z_i - z_j). \end{aligned}$$

By Theorem 2, f is maximum-ordered.

Example 3. A scheduling problem. We have k identical machines to do n jobs while job i requires setting the machine at level z_i. Assume that the cost of setting a machine from level x to level y is $g(x, y)$ where g is convex nondecreasing in $|x-y|$. It is easily verified that g is super-additive. By Theorem 4 an optimal scheduling is

obtained by the following steps:

(i) Order jobs according to z_i.

(ii) Obtain an optimal ordered partition.

(iii) For each machine schedule the jobs in order of z_i (either way).

Example 4. A clustering problem. Suppose that we want to partition a set of numbers into clusters with the following goals:

(i) The smaller the number of clusters the better.

(ii) Clusters should not vary too much in sizes.

(iii) Numbers in a cluster should be close.

It is easily seen that we can write

$$F(P) = C_M + \sum_{i=1}^{k} f(S_i)$$

where any criteria of the first two goals affect only C_M (which is a function of the shape M), and any criterion of the last goal affects only f. By our comments in the last paragraph of Sec. 1, it suffices to prove $OP = OOP$ for the shape-partition problem $F_M(P) = \Sigma_{i=1}^{k} f(S_i)$ with an arbitrarily given M. We consider two subproblems.

In the first subproblem $f(S)$ is the range of S. Let $S = \{z_1 \leqslant z_2 \leqslant \ldots \leqslant z_t\}$. Then $f(S) = \Sigma_{i=1}^{t-1} g(z_i, z_{i+1}) = \Sigma_{i=1}^{t-1} (z_{i+1} - z_i)$.

Since g is clearly superadditive, f is minimum-ordered by Theorem 4. In the second subproblem, $f(S)$ is the variance of S. Then

$$f(S_i) = \sum_{z \in S_i} g(z, \bar{z}_i) = \sum_{i=1}^{k} \sum_{z \in S_i} \frac{(z - \bar{z}_i)^2}{|S_i|}$$

where $\bar{z}_i$ is the mean of z in S_i. Since $\bar{z}_i$ minimizes $\Sigma_{z \in S_i} g(z, v)$ overall v and g is easily verified to be superadditive, f is minimum-ordered by the Corollary of Theorem 3.

6. Conclusion. Tanaev [5] considered optimal set partitions for labeled subsets. Chakravarty, Orlin and Rothblum [3] considered optimal set partitions for multivariate elements. In this paper we study optimal set partitions for unlabeled subsets and single variate elements.

REFERENCES

[1] BARNES, E. E. AND HOFFMAN, A. J., *Partitioning, spectra and linear programming,* Proc. Silver Jubilee Conf. Combinatories, Univ. of Waterloo, Ontario, 1982.

[2] CHAKRAVARTY, A. K., ORLIN J. B. AND ROTHBLUM, U. G., *A partitioning problem with additive objective with an application to optimal inventory groupings for joint replenishment,* Oper. Res. 30 (1982), 1018-1022.

[3] CHAKRAVARTY, A. K., ORLIN, J. B. AND ROTHBLUM, U. G., *Consecutive optimizers for a partitioning problem with applications to optimal inventory groupings for joint replenishment,* to appear.

[4] HWANG, F. K., *Optimal partitions,* J. Optimization Theory and Applications 34 (1981), 1-10.

[5] TANAEV, V. S., *Optimal subdivision of finite sets into subsets,* Akad. Nank BSSR Doklady 23 (1979), 26-28.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 2, April 1985

MAPPINGS AND FACETS FOR NON-ABELIAN GROUP PROBLEMS*

JULIAN ARÁOZ† AND ELLIS L. JOHNSON‡

Abstract. The main result is a generalization of two of Gomory's results for Abelian groups. The first result shows how to get a facet for a given problem from a facet of a problem on the homomorphic image of the given problem. For non-Abelian groups, this result gives facets having zero-valued coefficients on a normal subgroup. The second result characterizes all facets having zero-valued coefficients. For general subgroups, not normal, we must define an image which is more general than a group; we define multigroups and show that the subadditive characterization of facets still holds for such systems. Our motivation for introducing this additional generality is to explain facets of group problems having zero-valued coefficients on subgroups which are not normal.

AMS subject classifications. 20D99, 52A25, 90C10

1. Introduction. An important idea in deriving valid inequalities for an integer program has been to use mappings onto smaller or somehow simpler problems. An example is to map from an integer program onto an Abelian group problem and "lift" strong valid inequalities (e.g. facets) for the group problem back to the integer program to use as cuts (see [3, pp. 23-27]).

Gomory's results on homomorphisms and facets for the group problem are particularly interesting because he shows both that the lifted inequalities are facets [2, Thm. 19] and that all facets having zero coefficients, as the lifted facets do, come from lifting [2, Thm. 20]. The main result of this paper is to generalize these two theorems to non-Abelian groups.

We begin with a description of the non-Abelian group problem and some examples. In order to generalize to non-Abelian groups, we have to generalize the notion of homomorphisms and factor groups. The images of our mappings need not be groups, and we introduce the notion of a multigroup. The result that the subadditive characterization of facets [6] holds even for such systems is an additional, interesting result which was motivated by the attempt to characterize all facets having zero coefficients.

2. The non-Abelian group problem. A *group* is a set G with an addition $+$ such that for every g and h in G, $g+h$ is also in G and such that the following properties hold.

Property 2.1. $g+(h+k)=(g+h)+k$, for all g, h, k in G (associativity).

Property 2.2. $g+0=0+g$, for all $g \in G$ (zero element).

Property 2.3. $g+(-g)=(-g)+g=0$, for all $g \in G$ and some $-g \in G$ (negation).

The zero element 0 of G is easily seen to be unique, as is the negative $-g$ for a given g.

An *Abelian group* satisfies, in addition:

Property 2.4. $g+h=h+g$, for all g, h in G (commutativity).

A non-Abelian group is a group which is not commutative. Our results include the Abelian case but are not new there [2]. Thus, our main interest is in non-Abelian groups.

The *group problem* is determined by a group G, a *right-hand side* $b \in G$, and an *objective function* $c(g)$, $g \in G$. A *solution expression* to the group problem is an expression

* Received by the editors July 24, 1981, and in final revised form February 15, 1984.
† Universidad Simón Bolivar, Caracas, Venezuela.
‡ T. J. Watson Research Center, Yorktown Heights, New York 10598.

whose sum is b:

$$g+h+\cdots+k=b.$$

An *optimum solution expression* is a solution expression $g+h+\cdots+k$ which minimizes $c(g)+c(h)+\cdots+c(k)$ over all solution expressions.

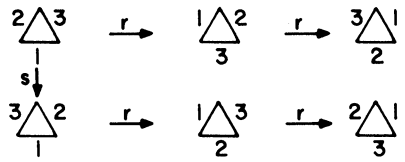


FIG. 1

Example. Consider as an example the smallest non-Abelian group: the trihedral group D_6 of positions of the triangle. This group has two generators: the rotation r and the reflection s; and is determined by $r^3=0$, $s^2=0$, $sr=r^2s$. Its addition table is given in Table 1. We have taken $g_1=r$, $g_2=r^2$, $g_3=s$, $g_4=rs$, and $g_5=r^2s$.

TABLE 1

	0	g_1	g_2	g_3	g_4	g_5
0	0	g_1	g_2	g_3	g_4	g_5
g_1	g_1	g_2	0	g_4	g_5	g_3
g_2	g_2	0	g_1	g_5	g_3	g_4
g_3	g_3	g_5	g_4	0	g_2	g_1
g_4	g_4	g_3	g_5	g_1	0	g_2
g_5	g_5	g_4	g_3	g_2	g_1	0

Let us take as right-hand side g_5 and as objective function

$$c(g_1)=c(g_3)=1,$$

$$c(0)=c(g_2)=c(g_4)=c(g_5)=100.$$

Then, $g_1+g_1+g_3\,(=g_5)$, is a solution expression with objective function value equal to 3, while $g_3+g_1\,(=g_5)$ is a solution expression with objective function value equal to 2. In fact, g_3+g_1 is an optimum solution expression. This problem can be stated as one of finding the smallest number of rotations and reflections (about a given axis) which will take us from the given initial position to the final specified position. With different objective functions, we get different optimum solution expressions, of course.

In this example, we effectively limited the group elements being considered to g_1 and g_3 by taking a large objective function value on other group elements. We always assume all group elements are present in the problem.

Assumption 2.5. *Master group assumption.* We assume that the group problems are *master group problems*, i.e. all group elements can be used in solution expressions. For example, if $b=g_5$, then b itself is a solution expression, but not an optimum one in our example since $c(b)=c(g_5)=100$.

We list several other assumptions used throughout.

Assumption 2.6. *Finite group assumption.* All of our groups will be finite.

Assumption 2.7. *Nonzero right-hand side.* Assume $b \neq 0$.

Assumption 2.8. *Nonnegative objective coefficients.* Assume $c(g) \geqq 0$ since otherwise there is no optimum solution because we can always use any element g many times in a solution: $b+g+g+\cdots+g=b$ provided the number of g's is a multiple of the order of the group.

Assumption 2.9. *Solution expressions without* 0. Since $c(0) \geqq 0$, every optimum solution expression can be assumed to not include 0.

3. The group polyhedron. Given an expression, the corresponding *incidence vector* is $(t(g), g \in G)$ where $t(g)$ is the number of times the group element g appears in the expression. The incidence vector t is a *solution vector* when it is the incidence vector of a solution expression. By assumption 2.9, $t(0)=0$ in every solution vector, so we delete $t(0)$ from t. Denote

$$G_+ = G - \{0\}. \tag{3.1}$$

Define the *group polyhedron* P to be:

$$P = \operatorname{conv}\{(t(g), g \in G_+ \mid t \text{ is a solution vector}\}. \tag{3.2}$$

The group polyhedron depends on both the group G and the right-hand side $b \in G_+$. When G is an Abelian group, the group polyhedron is called by Gomory [2] the corner polyhedron.

Group polyhedra have been shown [1] to be closed and have recession cone equal to R^d_+. We are interested in the facets of the group polyhedron, i.e., the minimal defining system of inequalities. Since the recession cone is R^d_+, every coefficient $\pi(g)$ in a facet

$$\sum_{g \in G_+} \pi(g)t(g) \geqq \pi_0$$

satisfies $\pi(g) \geqq 0$. Here, let $d = |G_+|$.

As an example, consider the group D_6 whose addition table is given in Table 1 of the previous section. Let the right-hand side be g_5. Then, there are four inequalities, other than $t_j \geqq 0$, $j = 1, \cdots, 5$ needed to define the group polyhedron:

$$\begin{aligned} 2t_1+2t_2+t_3+t_4+3t_5 &\geqq 3,\\ t_3+t_4+t_5 &\geqq 1,\\ t_1+t_2+t_3+t_5 &\geqq 1,\\ t_1+t_2+t_4+t_5 &\geqq 1. \end{aligned}$$

The vertices are

t_1	t_2	t_3	t_4	t_5
				1
1			1	
1		1		
	2		1	
	1	1		
		2	1	
		1	2	

For a right-hand side of $b = g_1$, there are seven facets:

$$\begin{aligned}
4t_1+2t_2+\ t_3+3t_4+3t_5 &\geqq 4,\\
4t_1+2t_2+3t_3+3t_4+\ t_5 &\geqq 4,\\
4t_1+2t_2+3t_3+\ t_4+3t_5 &\geqq 4,\\
2t_1+\ t_2+\ t_3+\ t_4+\ t_5 &\geqq 2,\\
t_1+\ t_2\ \ \ +\ t_4+\ t_5 &\geqq 1,\\
t_1+\ t_2+\ t_3+\ t_4\ \ \ &\geqq 1,\\
t_1+\ t_2+\ t_3+\ \ \ t_5 &\geqq 1.
\end{aligned}$$

The vertices are

t_1	t_2	t_3	t_4	t_5
1				
			1	1
		1		1
		1	1	
	2			
	1	2		
	1		2	
	1			2

The other possibilities for the right-hand side are essentially included since there are automorphisms taking g_5 to either g_3 or g_4 and taking g_1 to g_2.

Notice that this group, D_6, has four subgroups $\{1, 2\}$, $\{3\}$, $\{4\}$, $\{5\}$. In both cases $b = g_1, b = g_5$, there are facets having coefficients equal to zero for any subgroup not containing the right-hand side. This fact is not surprising since we have the results from Abelian groups to that effect and in any case

$$\sum_{g \notin K} t_g \geqq 1$$

is always a valid inequality, but not always a facet, for any subgroup K with $b \notin K$. This inequality reflects the fact that if K is a subgroup with $b \notin K$, then we cannot form an expression from elements in K which will add up to b. However, the resulting inequality need not be a facet, in general, but there is always a facet with coefficients $\pi(g) = 0$ for $g \in K$. This paper is mainly concerned with characterizing such facets.

There is a subadditive characterization of facets [6], which is a tool used here. Define the *subadditive cone* for G to be

$$\begin{aligned}
S = \{(\pi(g), g \in G_+) \,|\, &\pi(g+h) \leqq \pi(g)+\pi(h), \text{ if } g, h, \text{ and } g+h \in G_+,\\
&0 \leqq \pi(g)+\pi(h), \text{ if } g, h \in G_+ \text{ and } g+h = 0\}.
\end{aligned} \tag{3.3}$$

THEOREM 3.4 [6, Thm. 2]. *If $t \in P$, the group polyhedron for group G and right-hand side b, and if $\pi \in S$, the subadditive cone for group G, then*

$$\sum_{g \in G_+} \pi(g)t(g) \geqq \pi(b).$$

It should be clear that S is a polyhedral cone. In fact, $S \subseteq R_+^d$, $d = |G_+|$, is true. Thus S is pointed and has a finite number of extreme rays. We say that π is *on an extreme ray* if it is not the origin and is on the half-line making up the extreme ray.

THEOREM 3.5 [6, Thm. 7]. *The facets of P, other than* $t(g) \geqq 0$, *are among the vectors* π *on extreme rays of S. In fact, a vector* π *on an extreme ray of S is a facet of P if and only if it satisfies for each* $g \in G_+$ *at least one of* $\pi(b) = \pi(g') + \pi(g) + \pi(g'')$, *for some* $g', g'' \in G$ *satisfying* $b = g' + g + g''$.

The following proposition is an easy consequence of Theorem 3.5 and $S \subseteq R_+^d$.

PROPOSITION 3.6. *If* π *is a facet of P and if* $\pi(g) = 0$ *for any* $g \in G_+$, *then* $\pi(g) = 0$ *on some subgroup K of G and* $\pi(g) > 0$ *for* $g \notin K$.

Proof. All we need to show is that if $\pi(g) = 0$ and $\pi(h) = 0$, then $\pi(g+h) = 0$. This follows, in one direction, from

$$\pi(g+h) \leqq \pi(g) + \pi(h) = 0,$$

by $\pi \in S$ and Definition (3.3). On the other hand $\pi(g+h) \geqq 0$ follows from $S \subseteq R_+^d$.

4. Normal subgroups. For a group G (Abelian or non-Abelian) and any subgroup K, the *left* (or *right*) *cosets* can be formed by fixing any $g \in G$ and taking all $h = g + k$ (or $h = k + g$) for all $k \in K$. It is well known (see, e.g., [4, pp. 10–15]) that the resulting left cosets are disjoint and, hence, partition G. Another way to define cosets is to define g and h to be *left-equivalent* if $h = g + k$. Left-equivalence is, then, an equivalence relation and the left cosets are the equivalent classes.

The subgroup K is a normal subgroup if and only if the partition of G given by left cosets is identical to the partition given by right cosets. In case K is normal, we simply refer to cosets. When G is Abelian, every subgroup is normal. For a normal subgroup K, the *factor group* G/K is defined to have elements the cosets and addition defined from the addition $+$ of G. That is, if $\hat{g}$ is the coset containing g and $\hat{h}$ is the coset containing h, then

$$\hat{g} \hat{+} \hat{h} = \hat{k} \quad \text{where } \hat{k} \text{ is the coset containing } k = g + h,$$

defines an addition $\hat{+}$ on $\hat{G} = G/K$. The pair $\hat{G}$ and $\hat{+}$ can easily be shown to be a group. The mapping $\phi : G \to G/K$ defined by $\phi(g) = \hat{g}$, the coset containing g, is a homomorphism, i.e. satisfies

$$\phi(g+h) = \phi(g) \hat{+} \phi(h).$$

In fact, every homomorphism is such a mapping onto a factor group [4, p. 28]. The first of the two main results of Gomory [2, Thm. 19] on lifting facets for Abelian groups using homomorphisms is generalized in the theorem below to non-Abelian groups. We further generalize it in § 7.

THEOREM 4.1. *If K is a normal subgroup of G and if* $\hat{\pi}$ *is a facet for the factor group G/K with right-hand side* $\hat{b} \neq \hat{0}$, *the zero of G/K, then a facet for G with right-hand side b, where* $\phi(b) = \hat{b}$, *is given by*

$$\pi(g) = \hat{\pi}(\phi(g)),$$

where ϕ *is the homomorphism from G to G/K.*

The proof of Theorem 4.1 is interesting but is not given here because it is not too different from the Abelian case (for a different proof, see [5, p. 33]) and because it is a special case of Theorem 7.1, whose proof, however, is a bit more complicated because of its more general setting.

The second theorem of Gomory [2, Thm. 20] which we generalize is a converse of Theorem 4.1. As in the case of Theorem 4.1, we state it here for the normal subgroup case without proof. Theorem 8.1 generalizes the result to any subgroup.

THEOREM 4.2. *If π is a facet for the group G and right-hand side b and if $\pi(g)=0$ for all $g\in K$, where K is a normal subgroup of G, then π satisfies*

$$\pi(g)=\hat{\pi}(\phi(g)),$$

where ϕ is the homomorphism from G to G/K and $\hat{\pi}$ is some facet for G/K with right-hand side $\hat{b}=\phi(b)$.

We follow Gomory and call the derivation of π from $\hat{\pi}$ a *lifting* of the facet $\hat{\pi}$ from G/K to G.

From Proposition 3.6, if $\pi(g)=0$ for any $g\in G_+$, then $\pi(g)$ must be zero on a subgroup of G. Theorem 4.2 says that if that subgroup is normal (or contains a normal subgroup), then π comes from lifting some facet from a homomorphic image of G. The next two sections are a digression to develop the structure needed to generalize that result to any facet having some $\pi(g)=0$, $g\in G_+$. In the process, we generalize Theorem 4.1 since we define a more general lifting of facets.

Before proceeding, let us give a small example. For the dihedral group D_6, given in Table 1, the subgroup $\{0, g_1, g_2\}$ is normal and its factor group is isomorphic to C_2, the cyclic group of order 2. For a right-hand side $b=g_5$, the facet

$$t_3+t_4+t_5\geqq 1$$

comes from lifting the facet $\hat{t}_1\geqq 1$ from C_2 with right-hand side $\hat{g}_1$.

A subgroup of D_6 which is not normal is $\{0, g_3\}$. Yet, there is a facet for $b=g_5$, namely

$$t_1+t_2+t_4+t_5\geqq 1,$$

for which $\pi(g_3)=0$ and $\pi(g_i)=1$, $i\neq 3$. This facet is an example of one which we seek to explain.

5. Subgroups, submorphisms, and multigroups. Let K be any subgroup of a group G. In order not to be in a previously discussed case, think of G as being non-Abelian and K being a subgroup which is not normal.

DEFINITION 5.1. Two group elements g and h are *K-equivalent* if

$$g=k+h+k' \quad \text{for some } k \text{ and } k' \text{ in } K.$$

Since K is a subgroup, this relation is easily seen to be reflexive and transitive, so is an equivalence relation. Let us call the equivalence classes *K-classes* or *double cosets* [7].

One way to construct K-classes is to form left-cosets and right-cosets, and merge (take the union of) any two intersecting cosets until a partition of G is reached.

In the case of D_6 and subgroup $\{0, g_3\}$, the K-classes are just $\{0, g_3\}$ and $\{g_1, g_2, g_4, g_5\}$.

The dihedral group D_8 of order 8 is a more interesting case. Its addition table is given in Table 2. The subgroup $K=\{0, g_4\}$ gives K-classes $\{0, g_4\}$, $\{g_1, g_3, g_5, g_7\}$ and $\{g_2, g_6\}$. This example shows that K-classes need not all be of the same size.

PROPOSITION 5.2. *Let G be a group and K be a subgroup. Then, K is always itself a K-class, and every other K-class has order $i|K|$ where $1\leqq i\leqq |K|$.*

Proof. That K is itself a class is clear because if $h\in K$, then so does $k+h+k'$ for any $k, k'\in K$. That the maximum order of any class is $|K|^2$ is also clear because the

TABLE 2

	0	g_1	g_2	g_3	g_4	g_5	g_6	g_7
0	0	g_1	g_2	g_3	g_4	g_5	g_6	g_7
g_1	g_1	g_2	g_3	0	g_5	g_6	g_7	g_4
g_2	g_2	g_3	0	g_1	g_6	g_7	g_4	g_5
g_3	g_3	0	g_1	g_2	g_7	g_4	g_5	g_6
g_4	g_4	g_7	g_6	g_5	0	g_3	g_2	g_1
g_5	g_5	g_4	g_7	g_6	g_1	0	g_3	g_2
g_6	g_6	g_5	g_4	g_7	g_2	g_1	0	g_3
g_7	g_7	g_6	g_5	g_4	g_3	g_2	g_1	0

class is generated by fixing any h in the class and taking $k+h+k'$ over all $k, k' \in K$. To see that the order of any class is a multiple of K is more difficult. One way to form a K-class is to take the union of any two left-cosets which have a nonempty intersection with the same right-coset. In this way, we see that K-classes are unions of left-cosets, each of which has order $|K|$, completing the proof.

Let us return to the example of D_8. Let

$$\hat{0} = \{0, g_4\}, \quad \hat{g}_1 = \{g_1, g_3, g_5, g_7\}, \quad \hat{g}_2 = \{g_2, g_6\}.$$

Then, we can form the table

	$\hat{0}$	$\hat{g}_1$	$\hat{g}_2$
$\hat{0}$	$\hat{0}$	$\hat{g}_1$	$\hat{g}_2$
$\hat{g}_1$	$\hat{g}_1$	$\hat{0}, \hat{g}_2$	$\hat{g}_1$
$\hat{g}_2$	$\hat{g}_2$	$\hat{g}_1$	$\hat{0}$

This table would be an addition table except that

$$\hat{g}_1 \hat{+} \hat{g}_2 = \{\hat{0}, \hat{g}_2\}.$$

Thus we cannot define homomorphisms using K-classes. Yet our characterization of facets forces us to consider tables such as the one above. We resolve the difficulty by defining multigroups.

DEFINITION 5.3. *Multigroups.* Let $\hat{G}$ be a set of elements with an operation $\hat{+}$ mapping pairs $\hat{g}, \hat{h}$ onto a subset $\hat{g} \hat{+} \hat{h}$ of elements of $\hat{G}$. That is, the "sum" $\hat{g} \hat{+} \hat{h}$ of two elements is a set of elements, not just one element. The pair $\hat{G}, \hat{+}$ must satisfy:

(i) $\hat{g} \hat{+} (\hat{h} \hat{+} \hat{i}) = (\hat{g} \hat{+} \hat{h}) \hat{+} \hat{i}$ (associativity);

(ii) $\hat{0} \hat{+} \hat{g} = \hat{g} \hat{+} \hat{0} = \{\hat{g}\}$, for all $\hat{g} \in \hat{G}$ (zero);

(iii) for each $\hat{g} \in \hat{G}$, there exist exactly one $\hat{h} \in \hat{G}$ such that $\hat{0} \in \hat{g} \hat{+} \hat{h}$ and exactly one $\hat{i} \in \hat{G}$ such that $\hat{0} \in \hat{i} \hat{+} \hat{g}$, and $\hat{h} = \hat{i}$ (negative denoted $-\hat{g}$);

(iv) if $\hat{g} \in \hat{h} \hat{+} \hat{i}$, then $\hat{h} \in \hat{g} \hat{+} (-\hat{i})$ and $\hat{i} \in (-\hat{h}) \hat{+} \hat{g}$.

The element $\hat{0} \in \hat{G}$ is the zero element of $\hat{G}$, and the $\hat{h}$ in (iii) is the negative of $\hat{g}$, denoted $-\hat{g}$.

Condition (i) is a set-equality. In order to define expressions, such as in (i), in a multigroup, define

$$S \hat{+} T = \bigcup_{\substack{s \in S \\ t \in T}} (s \hat{+} t), \quad S \subseteq \hat{G}_+, \quad T \subseteq \hat{G}_+. \tag{5.4}$$

Thus, $\hat{+}$ is extended to operating on subsets of $\hat{G}$. In fact, we could be consistent by saying that $\hat{+}$ is only defined on subsets, and $\hat{g} \hat{+} S$ means $\{\hat{g}\} \hat{+} S$.

PROPOSITION 5.5. *Given a group G and a subgroup K, let ϕ be the mapping from G onto K-classes defined by $\phi(g) = \hat{g}$, the K-class containing g. Let $\hat{+}$ be defined on K-classes by*

$$\hat{g} \hat{+} \hat{h} = \bigcup_{\substack{g' \in \hat{g} \\ h' \in \hat{h}}} \phi(g' + h').$$

Then $\hat{G} = \{K\text{-classes of } G\}$ with $\hat{+}$ is a multigroup.

Proof. We must show 5.3 (i), (ii), (iii), and (iv). To prove (i), we show

$$\hat{g} \hat{+} (\hat{h} \hat{+} \hat{i}) = \{\phi(g' + (h' + i')) \mid g' \in \hat{g}, h' \in \hat{h}, \text{ and } i' \in \hat{i}\}, \tag{5.6}$$

and then (i) follows by associativity of G. To show (5.6) we use

$$\phi(g + h) \in \phi(g) \hat{+} \phi(h), \tag{5.7}$$

which should be clear from the definition of $\hat{+}$ in the statement of Proposition 5.5. In one direction, the proof of (5.6) is then easy:

$$\phi(g' + (h' + i')) \in \phi(g') \hat{+} \phi(h' + i') \subseteq \phi(g') \hat{+} (\phi(h') \hat{+} \phi(i')) = \hat{g} \hat{+} (\hat{h} \hat{+} \hat{i}).$$

In the other direction, let $\hat{l} \in \hat{g} \hat{+} (\hat{h} \hat{+} \hat{i})$. By the definition in Proposition 5.5 of $\hat{+}$ applied to $\hat{h} \hat{+} \hat{i}$,

$$\hat{l} \in \hat{g} \hat{+} \phi(h' + i') \quad \text{for some } h' \in \hat{h} \text{ and } i' \in \hat{i}.$$

Using the definition of $\hat{+}$ again gives

$$\hat{l} = \phi(g' + j) \quad \text{for some } g' \in \hat{g} \text{ and } j \in \phi(h' + i').$$

The proof will now be completed if we can show

$$j \in \phi(h' + i') \text{ implies } j = h'' + i'', \ h'' \in \hat{h} \text{ and } i'' \in \hat{i}.$$

If $j \in \phi(h' + i')$, then

$$j = k + h' + i' + k' \quad \text{for some } k, k' \in K.$$

Let $h'' = k + h'$ and $i'' = i' + k'$. Then, $h'' \in \hat{h}$ and $i'' \in \hat{i}$, completing the proof of (i).

The proof of (ii) is easy from Proposition 5.2 since $\hat{0} = K$.

To prove (iii), given g let g^r be the right-negative of g, i.e., $0 = g + g^r$. Then, clearly $\hat{0} \in \phi(g) + \phi(g^r)$. To show uniqueness, let $g' = k + g + k'$, k and k' in K. We must show that the right negative of g' is in $\phi(g^r)$. Let g'' be this right negative: $0 = g' + g''$. Then

$$0 = k + g + k' + g'', \quad \text{so}$$

$$k'' = g + k' + g'' \quad \text{where } k'' + k = 0, \text{ so } k'' \in K,$$

$$0 = g + k' + g'' + k \quad \text{again by } k'' + k = 0.$$

Hence, $g^r = k' + g'' + k$ and g^r and g'' are in the same K-class.

The proof of (iv) follows easily from (5.7).

PROPOSITION 5.8. *The multigroup $\hat{G}$ defined in Proposition 5.5 is a group (i.e., $|\hat{g} \hat{+} \hat{h}| = 1$ for all $\hat{g}, \hat{h} \in \hat{G}$) if and only if K is a normal subgroup of G.*

Proof. An alternative characterization of a normal subgroup is that

$$K = g + K + (-g) \quad \text{for all } g \in K.$$

All we need to prove here is that if K is not normal, then some $|\hat{g} \hat{+} \hat{h}| \geqq 2$. If K is not normal, then there is some $g \in K$ such that $g + K + (-g) \neq K$. Then, clearly,

$$g + K + (-g) \not\subseteq K, \quad \text{i.e., } g + k + (-g) \notin K, \text{ for some } k \in K.$$

Consider $|\hat{g} \hat{+} (-\hat{g})|$. Clearly, $K = \hat{0} \in \hat{g} \hat{+} (-\hat{g})$. But there are other elements in $\hat{g} \hat{+} (-\hat{g})$ because there is some $k \in K$ such that $g + k + (-g) \notin K$, and $g + k \in \hat{g}$. Hence, the proposition is proven.

PROPOSITION 5.9. *For $g \in G$ and $\hat{G}$ as defined in Proposition 5.5, $|\hat{g} \hat{+} \hat{h}| = |\hat{h} \hat{+} \hat{g}| = 1$ for all $\hat{h} \in \hat{G}$ if and only if*

$$K = g + K + (-g).$$

In that case, $|(-\hat{g}) \hat{+} \hat{h}| = |\hat{h} \hat{+} (-\hat{g})| = 1$, for all $\hat{h} \in \hat{G}$, and $\hat{g} \hat{+} \hat{x} = \{\hat{h}\}$ is uniquely solvable for all $\hat{h}$ and $\hat{x} \hat{+} \hat{g} = \{\hat{h}\}$ is uniquely solvable for all $\hat{h}$.

Proof. If $K \neq g + K + (-g)$, then $|\hat{g} \hat{+} (-\hat{g})| \geqq 2$. Conversely, if $K = g + K + (-g)$, then $|\hat{g} \hat{+} \hat{h}| = 1$ (the proof for $\hat{h} \hat{+} \hat{g}$ is similar) because $g' \in \hat{g}$ and $h' \in \hat{h}$ implies

$$\begin{aligned} g' + h' &= k^1 + g + k^2 + k^3 + h + k^4 \\ &= k^1 + g + k^5 + h + k^4, \quad \text{where } k^5 = k^2 + k^3 \\ &= k^1 + k^6 + g + h + k^4, \quad \text{where } k^6 \in K \text{ exists by } K = g + K + (-g) \\ &= k^7 + g + h + k^4 \in \phi(g + h), \quad \text{where } k^7 = k^1 + k^6. \end{aligned}$$

To show unique solvability,

$$\{x\} = (-\hat{g}) \hat{+} \hat{h}$$

clearly solves $\hat{g} \hat{+} x = \{\hat{h}\}$. To show uniqueness, if x is a solution, then

$$\begin{aligned} \{x\} = \hat{0} \hat{+} x &= ((-\hat{g}) \hat{+} \hat{g}) \hat{+} x, \quad \text{by } \{\hat{0}\} = (-\hat{g}) \hat{+} \hat{g} \\ &= (-\hat{g}) \hat{+} (\hat{g} \hat{+} x), \quad \text{by associativity,} \\ &= (-\hat{g}) \hat{+} \hat{h}, \quad \text{by } \hat{g} \hat{+} x = \{\hat{h}\}. \end{aligned}$$

The proof for the left-solution is similar.

We give one other result which is used in proving Theorem 7.1 to follow.

PROPOSITION 5.10. *For G and $\hat{G}$ as defined in Proposition 5.5, if $\hat{i} \in \hat{g} \hat{+} \hat{h}$ and $i \in \hat{i}$, then there exists some $g' \in \hat{g}$ and $h' \in \hat{h}$ such that $i = g' + h'$.*

Proof. If $i \in \hat{i}$ and $\hat{i} \in \hat{g} \hat{+} \hat{h}$, then there exist some $j \in \hat{i}$ such that

$$j = (k' + g + k^2) + (k^3 + h + k^4) \quad \text{for some } k', k^2, k^3, k^4 \in K.$$

Further, there is some $k^5, k^6 \in K$ such that $i = k^5 + j + k^6$. Hence,

$$i = k^5 + k' + g + k^2 + k^3 + h + k^4 + k^6, \quad \text{or}$$

$$i = g' + h', \quad \text{where } g' = (k^5 + k') + g + k^2 \in \hat{g} \text{ and } h' = k^3 + h + (k^4 + k^6) \in \hat{h},$$

completing the proof.

6. Subadditive characterization for multigroups. The results in this section could be stated more generally since Definition 5.3 (i), (ii), (iii), and (iv) need not hold in order for the subadditive characterization to be valid. Since we do not prove the results

here, in any case, we state only what is needed here. We follow the development of Johnson [6] (see also Aráoz and Johnson [1]).

Let $\hat{G}$ with $\hat{+}$ be a multigroup (as defined in 5.3) and let $\hat{b} \in \hat{G}$, $\hat{b} \neq \hat{0}$.

DEFINITION 6.1. An *expression* in $\hat{G}$ is defined recursively as

(i) $E = (g)$, $g \in \hat{G}$, is a primitive expression;

(ii) given two expressions E_1 and E_2, form expression E by $E = (E_1 \hat{+} E_2)$.

We also allow the empty string as an expression. A *subexpression* of E is defined by: the empty string and E itself are always subexpressions of E, and if E is defined by (ii), then any subexpression of E_1 or E_2 is a subexpression of E.

DEFINITION 6.2. The *evaluation* of E is a function γ defined on expressions by:

(i) $\gamma(E) = \{g\}$, if $E = (g)$ is a primitive expression;

(ii) $\gamma(E) = \gamma(E_1) \hat{+} \gamma(E_2)$, if $E = E_1 \hat{+} E_2$, where $\hat{+}$ here is defined on sets by 5.4;

(iii) $\gamma(\text{empty string}) = \{\hat{0}\}$.

Thus, γ maps from expressions to subsets of $\hat{G}$.

DEFINITION 6.3. An *incidence vector* $(\hat{t}(\hat{g}), \hat{g} \in \hat{G})$ of an expression E is defined by $t(g)$ equals the number of times (g) appears as a primitive subexpression of E.

DEFINITION 6.4. A *solution expression* is an expression E such that $\hat{b} \in \gamma(E)$. A *solution vector* is an incidence vector of a solution expression. Since $\hat{0}$ can be deleted from a solution expression without affecting its evaluation, we assume that $t(\hat{0}) = 0$ in any solution vector and leave out $\hat{0}$ from $\hat{t}$; that is, $\hat{t}(g)$ is defined for $g \in \hat{G}_+$.

DEFINITION 6.5. *The multigroup polyhedron* $\hat{P}$ is defined by

$$\hat{P} = \text{conv}\,\{(\hat{t}(g), g \in \hat{G}_+) \mid \hat{t} \text{ is a solution expression}\}$$

Thus, $\hat{P}$ depends on $\hat{G}$ and on $\hat{b}$.

THEOREM 6.6. *$\hat{P}$ is a polyhedron whose recession cone is R^d_+.*

This theorem's proof is similar to that of [1, Thms. 7.2, 8.7].

THEOREM 6.7. *The facets*

$$\sum_{\hat{g} \in G_+} \hat{\pi}(\hat{g})\hat{t}(\hat{g}) \geqq \hat{\pi}(\hat{b})$$

of $\hat{P}$, other than $\hat{t}(\hat{g}) \geqq 0$, are on the extreme rays of $\hat{S} = \{(\hat{\pi}(\hat{g}), \hat{g} \in \hat{G}_+) \mid \hat{\pi}(\hat{i}) \leqq \hat{\pi}(\hat{g}) + \hat{\pi}(\hat{h})$, if $\hat{i}, \hat{g}, \hat{h} \in \hat{G}$ and $\hat{i} \in \hat{g} \hat{+} \hat{h}$, and $0 \leqq \hat{\pi}(\hat{g}) + \hat{\pi}(\hat{h})$, if $\hat{0} \in \hat{g} \hat{+} \hat{h}\}$. The extreme rays of $\hat{S}$ giving facets are precisely those giving minimal valid inequalities.

The proof of this theorem follows closely the development for additive systems [6]. The main use of this result here is to show that facets for multigroup problems have the same sort of characterization as for group problems so that we can prove lifting theorems.

Example 6.8. We close this section with an example from D_{12}, the dihedral group of order 12. Without detailing the derivation, the multigroup below comes from the subgroup of D_{12} consisting of 0 and the reflection.

	$\hat{0}$	$\hat{g}_1$	$\hat{g}_2$	$\hat{g}_3$
$\hat{0}$	$\hat{0}$	$\hat{g}_1$	$\hat{g}_2$	$\hat{g}_3$
$\hat{g}_1$	$\hat{g}_1$	$\hat{0}, \hat{g}_2$	$\hat{g}_1, \hat{g}_3$	$\hat{g}_2$
$\hat{g}_2$	$\hat{g}_2$	$\hat{g}_1, \hat{g}_3$	$\hat{0}, \hat{g}_2$	$\hat{g}_1$
$\hat{g}_3$	$\hat{g}_3$	$\hat{g}_2$	$\hat{g}_1$	$\hat{0}$

Thus, the inequalities on $\hat{\pi}_1$, $\hat{\pi}_2$, $\hat{\pi}_3$ are

$\hat{\pi}_1$	$\hat{\pi}_2$	$\hat{\pi}_3$
2		$\geqq 0$
2	−1	$\geqq 0$
1	1	$-1 \geqq 0$
1	−1	$1 \geqq 0$
	2	$\geqq 0$
−1	1	$1 \geqq 0$
		$2 \geqq 0$

.

The extreme rays are given by

$\hat{\pi}_1$	$\hat{\pi}_2$	$\hat{\pi}_3$
1		1
1	1	
1	2	1
1	2	3

.

Which ones are facets depends now on the choice of the right-hand side. If $\hat{b} = \hat{g}_2$, then the facets are

$\hat{\pi}_1$	$\hat{\pi}_2$	$\hat{\pi}_3$
1	1	
1	2	1

or $t_1 + t_2 \geqq 1$,
$t_1 + 2t_2 + t_3 \geqq 2$

and the vertices are

$\hat{t}_1$	$\hat{t}_2$	$\hat{t}_3$
	1	
1		1
2		

.

Figure 2 shows the polyhedron P.

7. Lifting facets.

THEOREM 7.1. *Let G be a group, K a subgroup of G, and $b \in G - K$. If $\hat{\pi}$ is a facet for the multigroup $\hat{G}$ of K-classes with right-hand side $\hat{b} = \phi(b)$, for the mapping of g to*

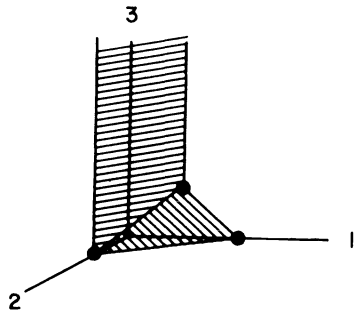


FIG. 2

the K-class containing g, then π *given by*

$$\pi(g) = \hat{\pi}(\phi(g)),\ g \in G_+,$$

is a facet

$$\sum_{g \in G_+} \pi(g)t(g) \geqq \pi(b)$$

for the group polyhedron P for group G and right-hand side b.

Proof. We rely heavily upon the subadditive characterization for this proof. First, π as defined here will be shown to be subadditive. This result follows from

$$\begin{aligned}\pi(g+h) &= \hat{\pi}(\phi(g+h)) \\ &\leqq \hat{\pi}(\phi(g)) + \hat{\pi}(\phi(h)) \quad \text{by (5.7) and Theorem 6.7} \\ &= \pi(g) + \pi(h) \quad \text{by definition of } \pi.\end{aligned}$$

When $g+h=0$, the same argument shows $0 \leqq \pi(g)+\pi(h)$. Hence, π is subadditive.

To show that π is extreme in the subadditive cone, we follow the argument in [5, p. 33]. Let S be the matrix with $|G_+|$ rows and a column for each subadditive inequality holding with equality:

$$\pi(g+h) = \pi(g)+\pi(h), \quad \text{or} \quad 0 = \pi(g)+\pi(h), \quad \text{where } 0 = g+h.$$

In a typical column of S, we put a $+1$ in rows g and h and a -1 in row $g+h$. In order for π to be extreme in the subadditive cone, the matrix S must have rank $|G_+|-1$. Thus, if we can show that the solution set to

$$\lambda S = 0$$

has rank 1, we will be done. That is, we must show that every solution λ is some multiple of π, which does satisfy $\lambda S = 0$. Suppose that λ is a solution to $\lambda S = 0$.

We first show that $\lambda(k) = 0$ must hold for all $k \in K$. This result follows from the fact that $\pi(k) = 0$ for all $k \in K$ so every addition relation $k'' = k + k'$ for k and k' in K (and hence k'' in K) corresponds to a column in S. But K is a subgroup, and there is no real solution to $\lambda(k'') = \lambda(k) + \lambda(k')$ for all $k, k' \in K$ except $\lambda(k) = 0$ all $k \in K$.

Next, we show that $\lambda(g) = \lambda(h)$ whenever g and h are in the same K-class. But then $g = k + h + k''$, for k and k'' in K, so $g + k' = k + h$, for $k' = -k'' \in K$.

Let $g' = g + k'$. Since g, g', and h all are in the same K-class

$$\pi(g) = \pi(g') = \pi(h) \quad \text{by definition of } \pi.$$

Hence, the relations

$$\pi(g') = \pi(g) + \pi(k') \quad \text{and} \quad \pi(g') = \pi(k) + \pi(h)$$

must hold since $\pi(k) = \pi(k') = 0$ by $k, k' \in K$. Hence,

$$\lambda(g') = \lambda(g) + \lambda(k') \quad \text{and} \quad \lambda(g') = \lambda(k) + \lambda(h)$$

must hold. We have just shown that

$$\lambda(k) = \lambda(k') = 0 \quad \text{for } k, k' \in K.$$

Hence, $\lambda(g) = \lambda(g') = \lambda(h)$. Thus, $\lambda(g) = \lambda(h)$ for any two g, h in the same K-class.

If λ is not some multiple of π, then we can define $\hat{\lambda}$ on $\hat{G}_+$ by

$$\hat{\lambda}(\hat{g}) = \lambda(g) \quad \text{for any } g \text{ with } \hat{g} = \phi(g),$$

since $\lambda(g)$ is the same on every K-class. This $\hat{\lambda}$ will not be a multiple of $\hat{\pi}$ since λ was not a multiple of π. Yet $\hat{\lambda}$ satisfies

$$\hat{\lambda}\hat{S} = 0$$

where $\hat{S}$ has a column for each inequality holding with equality:

$$\hat{\pi}(\hat{g} \hat{+} \hat{h}) = \hat{\pi}(\hat{g}) + \hat{\pi}(\hat{h}),$$

$$0 = \hat{\pi}(\hat{g}) + \hat{\pi}(\hat{h}) \quad \text{when } \hat{0} = \hat{g} \hat{+} \hat{h}.$$

Thus, a contradiction of extremality of $\hat{\pi}$ in the subadditive cone for $\hat{G}$, is reached. Hence, π must be extreme in the subadditive cone for G.

It remains to show minimality of the valid inequality:

$$\sum_{g \in G_+} \pi(g) t(g) \geqq \pi_0 \quad (= \pi(b)) \quad \text{for all solution vectors } t.$$

Suppose that it were not minimal. That is, suppose some $\pi(h)$ could be lowered to $\pi'(h) < \pi(h)$ and the inequality remained valid for all solution vectors t. We reach a contradiction by showing that $\hat{\pi}(\phi(h))$ could also be lowered while maintaining

$$\sum_{\hat{g} \in \hat{G}_+} \hat{\pi}(\hat{g}) \hat{t}(\hat{g}) \geqq \hat{\pi}_0 \quad (= \hat{\pi}(\hat{b}))$$

valid for all solution vectors $\hat{t}$. What we need to show is that if $\pi(h)$ can be lowered, then so can all other $\pi(g)$ for g in the K-class $\hat{h}$ be lowered at the same time while still maintaining validity of π.

In order to show that all $\pi(g)$, $g \in \hat{h}$, can be lowered at the same time, let us consider solution vectors t and their solution expressions E. If g and h are in the same K-class, then

$$g = k + h + k' \quad \text{where } k \text{ and } k' \text{ are in } K,$$

so any time g appears in E, we could substitute $k + h + k'$ in place of E without changing t except that $t(k)$ and $t(k')$ increase. Since $\pi(k) = \pi(k') = 0$, and $\pi(g) = \pi(h)$ this change does not affect

$$\sum_{g \in G_+} \pi(g) t(g).$$

Thus, there is a certain symmetry in the values of $t(g)$, $g \in \hat{h}$, among all solution vectors t and $\sum \pi(g) t(g)$ is the same among all solution vectors having the same value of

$$\sum_{g \in \hat{h}} t(g).$$

It should be clear that if $\pi(h)$ can be lowered so can every $\pi(g)$, $g \in \hat{h}$. The proof is thus completed.

We conclude this section with an example for D_{12}, the dihedral group of order 12. If we take the rotation r and the reflection m, then this group is given by 0; $g_i = r^i$, $i = 1, \cdots, 5$; $g_{6+i} = r^i m$, $i = 0, \cdots, 5$. The defining relations are $r^6 = 0$, $mr = r^5 m$, $m^2 = 0$. The subgroup $K = \{0, g_6\}$ gives K-classes.

$$\hat{0} = \{0, g_6\}, \quad \hat{g}_1 = \{g_1, g_5, g_7, g_{11}\}, \quad \hat{g}_2 = \{g_2, g_4, g_8, g_{10}\}, \quad \hat{g}_3 = \{g_3, g_7\}.$$

From Example 6.8, we have a facet for the multigroup $\hat{0}$, $\hat{g}_1$, $\hat{g}_2$, $\hat{g}_3$ given by

$$\hat{t}_1 + 2\hat{t}_2 + \hat{t}_3 \geqq 2$$

for a right-hand side $\hat{b} = \hat{g}_2$. The lifting theorem says that

$$t_1 + 2t_2 + t_3 + 2t_4 + t_5 + t_7 + 2t_8 + t_9 + 2t_{10} + t_{11} \geqq 2$$

is a facet for D_{12} with right-hand side $b = g_2$ (or g_4, or g_8, or g_{10}).

8. Facets with zero coefficients.

THEOREM 8.1. *Let G be a group, $b \in G_+$ be the right-hand side, and $(\pi(g), g \in G_+)$ give a facet (other than $t(g) \geqq 0$);*

$$\sum_{g \in G_+} \pi(g) t(g) \geqq \pi(b)$$

of the group polyhedron P. If any $\pi(g) = 0$, $g \in G_+$, then $\pi(g) = 0$ if and only if $g \in K$, for some subgroup K, and π is derived by lifting a facet $\hat{\pi}$, as in Theorem 7.1, from the polyhedron for multigroup $\hat{G}$ equal to the K-classes and right-hand side $\hat{b}$ to the K-class containing b.

Proof. Proposition 3.6 has already shown that $\pi(g) = 0$ for g in some subgroup K.

We next show that $\pi(g) = \pi(h)$ if g and h are in the same K-class, that is, if

$$g = k + h + k''.$$

Let $-k'' = k'$ so that

$$g + k' = k + h.$$

Denote $g' = g + k'$ $(= k + h)$. Then, subadditivity of π assures

$$\pi(g') \leqq \pi(g) + \pi(k') = \pi(g) \quad \text{and} \quad \pi(g') \leqq \pi(k) + \pi(h) = \pi(h),$$

by $\pi(k) = \pi(k') = 0$ because $k, k' \in K$. Clearly,

$$\pi(g) = \pi(g' + (-k')) \leqq \pi(g') + \pi(-k') = \pi(g'),$$

and similarly $\pi(h) \leqq \pi(g')$. Hence, $\pi(g) = \pi(g') = \pi(h)$, showing that π is constant on K-classes.

We remark that $\pi(b) > 0$ so $b \notin K$, and $\hat{b} \neq \hat{0}$.

We next show that $\hat{\pi}$ defined on $\hat{G}$ by

$$\hat{\pi}(\hat{g}) = \pi(g) \quad \text{for any } g \in \hat{g}$$

is subadditive. We can define $\hat{\pi}$ in this way because π is the same for every $g \in \hat{g}$, for $\hat{g}$ any K-class. To show subadditivity of $\hat{\pi}$, let g and $h \in G_+$ and let $i = g + h$. Then,

$$\hat{\pi}(\hat{i}) = \pi(i) \leqq \pi(g) + \pi(h) = \hat{\pi}(\hat{g}) + \hat{\pi}(\hat{h}).$$

However, there may be other K-classes $\hat{j} \in \hat{g} \hat{+} \hat{h}$ whenever there is some $j \in G_+$ given

by

$$j = k + g + k' + h + k'', \qquad k, k', k'' \in K,$$

not in the same K-class as i. By subadditivity of π, generalized to more than two elements,

$$\begin{aligned}\hat{\pi}(\hat{j}) = \pi(j) &\leqq \pi(k) + \pi(g) + \pi(k') + \pi(h) + \pi(k'') \\ &\leqq \pi(g) + \pi(h) \quad \text{by } \pi(k) = 0, \text{ all } k \in K \\ &= \hat{\pi}(\hat{g}) + \hat{\pi}(\hat{h}).\end{aligned}$$

Hence, $\hat{\pi}$ is subadditive, and

$$\sum_{\hat{g} \in \hat{G}_+} \hat{\pi}(\hat{g}) \hat{t}(\hat{g}) \geqq \hat{\pi}(\hat{b})$$

is a valid inequality for $\hat{G}$ with right-hand side $\hat{b}$.

It should also be clear that this inequality is minimal because if any $\hat{\pi}(g)$ could be lowered, we could lift it back to a valid inequality smaller than π contradicting π being minimal. It remains to show that $\hat{\pi}$ is extreme in its subadditive cone.

Suppose that $\hat{\pi}$ is not extreme in the subadditive cone for $\hat{G}$. Then

$$\hat{\pi} = \hat{\pi}^1 + \hat{\pi}^2$$

for $\hat{\pi}^1$ and $\hat{\pi}^2$ in the subadditive cone and not equal to a multiple of $\hat{\pi}$. We can lift $\hat{\pi}^1$ and $\hat{\pi}^2$ by defining

$$\pi^i(g) = \hat{\pi}^i(\hat{g}), \qquad i = 1, 2, \quad \hat{g} \text{ the } K\text{-class containing } g.$$

As in the proof of Theorem 7.1, both π^1 and π^2 are in the subadditive cone for G. Clearly,

$$\pi = \pi^1 + \pi^2,$$

and π^1 and π^2 are not multiples of π, contradicting π being extreme in the subadditive cone for G. The theorem is thus proven.

Appendix. Double cosets and multigroups. Our motivation for this work has been to characterize facets of group polyhedra having zero coefficients. In order to do so, we have been forced to consider algebraic objects consisting of a set and an addition table with multiple entries for sums. Having done so, we could now set the whole question in terms of such objects. Here, we discuss the nature of this extension and give an example.

To begin, let us consider addition + to be over subsets of G, where G is, for our purposes, a finite set. Then, $g + h$ is defined to mean $\{g\} + \{h\}$ for singletons $\{g\}$ and $\{h\}$. One already has used + in this way, e.g. in defining cosets to be $g + K$. Addition need only be defined on singletons since we want to define

$$S + T = \cup \{s + t \mid s \in S \text{ and } t \in T\}.$$

We now allow the sum of two singletons to be an arbitrary subset of group elements, but always require 5.3 (i), (ii), (iii) and (iv) to hold; that is, (i) associativity; (ii) existence of zero; (iii) existence of negative; and (iv) solvability. Thus, when + is single-valued, G must be a group.

Let us now define a homomorphism from one multigroup G to $\hat{G}$ to be a mapping ϕ such that

(A.1) $\phi(0) = \hat{0}$, the zero of $\hat{G}$.

(A.2) $\phi(j) \in \hat{g} + \hat{h}$, $\hat{g}, \hat{h} \in \hat{G}$, if and only if $j \in g' + h'$ for some g', $h' \in G$ with $\phi(g') = \hat{g}$ and $\phi(h') = \hat{h}$.

(A.3) For $\phi(j) \in \hat{g} \hat{+} \hat{h} \hat{+} \hat{k}$ and a given $h \in G$ with $\phi(h) = \hat{h}$, there exists g and $k \in G$ with $\phi(g) = \hat{g}$, $\phi(k) = \hat{k}$, and such that $j = g + h + k$.

The multigroups we are interested in are *double-coset groups* $\hat{G}$ whose elements are of the form $K + g + K$ for some subgroup K of G and all $g \in G$. The homomorphisms we are interested in are mappings from a group G to one of its double-coset groups $\hat{G} = \{K + g + K \mid g \in G$ and K a fixed subgroup of $G\}$. In fact, the multigroups which are homomorphic images of groups are precisely these double-coset groups. Even more is true, as stated in the next proposition. First some definitions are needed.

DEFINITION A.4. For a multigroup G, a subset H is a *subgroup of* G if

(i) $g \in H$ and $h \in H$ implies $g + h \subseteq H$;

(ii) $h \in H$ implies the negative (see Definition 5.3 (iii)) of h is in H.

DEFINITION A.5. If G is a multigroup and K is a subgroup, then the *double coset group* of G from K has elements $K + g + K$, $g \in G$, with addition $\hat{+}$ defined from G by

$$(K + g + K) \hat{+} (K + h + K) = \bigcup_{k \in K} \{K + g + k + h + K\}.$$

It is true that the double cosets $K + g + K$ in this more general setting still partition G, because K-equivalence:

$$g \overset{K}{\sim} h \quad \text{if } g \in K + h + K,$$

is still an equivalence relation. For this result, Property 5.3 (iv) is needed.

PROPOSITION A.6. *If G is a double-coset group of G^0 from K^0 and if ϕ is a homomorphism from G to $\hat{G}$, for $\hat{G}$ any multigroup, then $\hat{G}$ must be a double-coset group of G^0 from some subgroup H^0 of G^0, $H^0 \supseteq K^0$.*

Proof. Let $K \subseteq G$ be defined by

$$K = \{k \in G \mid \phi(k) = \hat{0}, \text{ the zero of } \hat{G}\}.$$

Then K is a subgroup to G because if $k_1, k_2 \in K$, then $\phi(k_1) \hat{+} \phi(k_2) = \hat{0} + \hat{0} = \{\hat{0}\}$. For any $k \in k_1 + k_2$, we must show that $k \in K$, i.e. $\phi(k) = \hat{0}$. By $k \in k_1 + k_2$, (A.1), and $\phi(k_1) = \phi(k_2) = \hat{0}$, we have

$$\phi(k) \in \{\hat{0}\}, \quad \text{so } \phi(k) = \hat{0}.$$

Next, we must show that the negative $(-k)$ of k is also in K, i.e. $\phi(-k) = \hat{0}$. From (A.1), $\phi(0) = \hat{0}$. From $k + (-k) = 0$,

$$\phi(0) \in \phi(k) + \phi(-k) = \{\phi(-k)\},$$

by $\phi(k) = \hat{0}$ and 5.3 (ii). Hence, $\phi(-k) = \hat{0}$, and K is a subgroup.

We next show that $\phi(g)$ is equal for all $g \in K + h + K$, for any fixed $h \in G$. For any such g,

$$g \in k_1 + h + k_2, \quad \text{some } k_1, k_2 \in K,$$

and hence

$$\phi(g) \in \phi(k_1) \hat{+} \phi(h) \hat{+} \phi(k_2), \quad \text{from (A.2) applied twice.}$$

By $\phi(k_1) = \phi(k_2) = \hat{0}$, $\phi(g) \in \{\phi(h)\}$ so $\phi(g) = \phi(h)$.

We next show that $\phi(g) = \phi(h)$ implies that $g \in K + h + K$. Suppose that $\phi(g) = \phi(h)$. Clearly, $\phi(g) \in \hat{0} \hat{+} \phi(h) \hat{+} \hat{0}$, and by Property (A.3) of homomorphisms, there exist $k, k' \in G$ such that $\phi(k) = \hat{0}$, $\phi(k') = \hat{0}$, and $g = k + h + k'$. Thus $g \in K + h + K$. In fact, the only use made of (A.3) is here so we could weaken it to $\hat{g} = \hat{k} = \hat{0}$.

It remains to show that addition $\hat{+}$ must be that defined by double-cosets. By (A.2), $\phi(j) \in \hat{g} \hat{+} \hat{h}$ if and only if $j \in g+h$ for some $g, h \in G$ such that $\phi(g) = \hat{g}$ and $\phi(h) = \hat{h}$. If $j \in g+h$ for some $g, h \in G$, then the double-coset $K+j+K$ is in $K+g+K+h+K$ so $\hat{g} \hat{+} \hat{H}$ is a subset of those $K+j+K$ in the sum $(K+g+K)+(K+h+K)$. Conversely, if $j \in K+g+K+h+K$ for $\phi(g) = \hat{g}$ and $\phi(h) = \hat{h}$, then $\phi(k+g+k') = \hat{g}$ and $\phi(k''+h+k''') = \hat{h}$ for all $k, k', k'', k''' \in K$. Hence, $j \in g'+h'$ for some g', h' with $\phi(g') = \hat{g}$ and $\phi(h') = \hat{h}$, and, thus, $\phi(j) \in \hat{g} \hat{+} \hat{h}$.

COROLLARY A.7. *If G is a group and ϕ is a homomorphism from G to $\hat{G}$, for $\hat{G}$ any multigroup, then $\hat{G}$ must be a double-coset group of G for some subgroup K of G. The image $\hat{G}$ of ϕ is another group if and only if the kernel K of ϕ is a normal subgroup of G.*

COROLLARY A.8. *If G^0 is a group and ϕ is a homomorphism from G^0 to $\hat{G}$, for $\hat{G}$ any multigroup, then there are homomorphisms ϕ^1 from G^0 to some multigroup G and ϕ^2 from G to $\hat{G}$ such that $\phi = \phi^2 \circ \phi^1$ if and only if the kernel H^0 of ϕ contains a subgroup K^0.*

The meaning of these results is that we could have begun by considering the class of double-coset groups. Within this class, we can lift facets from homomorphic images and get all facets with zero coefficients. However, for a group G, the only facets we get are from double-coset groups coming from subgroups of that G since the homomorphisms from such double-coset groups are always onto another such double-coset group.

We conclude with an example from D_{12} the dihedral group of order 12. In Example 6.8, we gave an example of a double-coset group with kernel $K = \{0, s\}$. That table is

$\{0, s\}$	$(r, r^5, sr, sr^5\}$	$\{r^2, r^4, sr^2, sr^4\}$	$\{r^3, sr^3\}$
$\{r, r^5, sr, sr^5\}$	$\{0, s\}, \{r^2, r^4, sr^2, sr^4\}$	$\{r^3, sr^3\}, \{r, r^5, sr, sr^5\}$	$\{r^2, r^4, sr^2, sr^4\}$
$\{r^2, r^4, sr^2, sr^4\}$	$\{r^3, sr^3\}, \{r, r^5, sr, sr^5\}$	$\{0, s\}, \{r^2, r^4, sr^2, sr^4\}$	$\{r, r^5, sr, sr^5\}$
$\{r^3, sr^3\}$	$\{r^2, r^4, sr^2, sr^4\}$	$\{r, r^5, sr, sr^5\}$	$\{0, s\}$

On letting $\hat{0} = \{0, s\}$, $\hat{r} = \{r, r^5, sr, sr^5\}$, $\hat{r}^2 = \{r^2, r^4, sr^2, sr^4\}$ and $\hat{r}^3 = \{r^3, sr^3\}$, we have

$\hat{0}$	$\hat{r}$	$\hat{r}^2$	$\hat{r}^3$
$\hat{r}$	$\hat{0}, \hat{r}^2$	$\hat{r}^3, \hat{r}$	$\hat{r}^2$
$\hat{r}^2$	$\hat{r}^3, \hat{r}$	$\hat{0}, \hat{r}^2$	$\hat{r}$
$\hat{r}^3$	$\hat{r}^2$	$\hat{r}$	$\hat{0}$

Now, $\hat{H} = \{\hat{0}, \hat{r}^3\}$ is a subgroup for this table, and forming $\hat{H}\hat{g}\hat{H}$ gives

$\{\hat{0}, \hat{r}^3\}$	$\{\hat{r}, \hat{r}^2\}$
$\{\hat{r}, \hat{r}^2\}$	$\{\hat{0}, \hat{r}^3\}, \{\hat{r}, \hat{r}^2\}$

We get the same table by letting

$$H = \{0, s, r^3, sr^3\}$$

and forming HgH, $g \in G$:

$\{0, s, r^3, sr^3\}$	$\{r, r^2, r^4, r^5, sr, sr^2, sr^4, sr^5\}$
$\{r, r^2, r^4, r^5, sr, sr^2, sr^4, sr^5\}$	$\{0, s, r^3, sr^3\}, \{r, r^2, r^4, r^5, sr, sr^2, sr^4, sr^5\}$.

This example illustrates Corollary A.8 in that the sequence of mappings having kernels $\{0, s\}$ and $\{0, r^3\}$ is equivalent to the mapping having kernel $\{0, s, r^3, sr^3\}$.

There is another interesting subgroup of $\hat{G}$, namely, $H = \{0, r^2\}$. This subgroup is "normal" in $\hat{G}$ in that $\hat{g} \hat{+} \hat{H} = \hat{H} \hat{+} \hat{g}$, all $\hat{g} \in \hat{G}$, and in fact the addition table obtained from $\hat{G}$ by forming double cosets $\hat{H} \hat{+} \hat{g} \hat{+} \hat{H}$ is

$\{\hat{0}, \hat{r}^2\}$	$\{\hat{r}, \hat{r}^3\}$
$\{\hat{r}, \hat{r}^3\}$	$\{\hat{0}, \hat{r}^2\}$

which is the same addition table as C_2, the cyclic group of order two. We thus have illustrated a generalization of Proposition 5.9 to the case where G is already a double-coset group. The proof there carries over to this more general case. That is, the image $\hat{G}$ of a homomorphism from one double-coset group G onto $\hat{G}$ is isomorphic to a group if and only if the kernel K is a normal subgroup of the double-coset group G.

Acknowledgments. This work was done while the second author was supported by the Alexander von Humboldt Foundation as a visiting senior scientist. Our thanks go to that foundation and to Professor Bernhard Korte and the Institute for Operations Research of the University of Bonn.

REFERENCES

[1] J. ARÁOZ AND E. L. JOHNSON, *Some results on polyhedra of semigroup problems*, this Journal, 2 (1981), pp. 244-258.

[2] R. E. GOMORY, *Some polyhedra related to combinatorial problems*, Linear Alg. and Appl., 2 (1969), pp. 451-558.

[3] R. E. GOMORY AND E. L. JOHNSON, *Some continuous functions related to corner polyhedra*, Math. Programming, 3 (1972), pp. 23-85.

[4] M. HALL, JR., *The Theory of Groups*, second ed., Chelsea, New York, 1976.

[5] E. L. JOHNSON, *Integer Programming: Facets, Subadditivity, and Duality for Group and Semigroup Problems*, CBMS-NSF Regional Conference Series in Applied Mathematics 32, Society for Industrial and Applied Mathematics, Philadelphia, 1980.

[6] ———, *On the generality of the subadditive characterization of facets*, Math. Oper. Res., 6 (1981), pp. 101-112.

[7] A. G. KUROSH, *The Theory of Groups*, Vol. 1, second English edition edited by K. A. Hirsch, Chelsea, New York, 1960.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 2, April 1985

PROBABILISTIC ANALYSIS OF GEOMETRIC LOCATION PROBLEMS*

EITAN ZEMEL†

Abstract. We analyze the behavior of the k center and median problems for n points randomly distributed in an arbitrary region A of R^d. Under a mild assumption on the region A, we show that for $k \leqq k(n) = o(n/\log n)$, the objective function values of the discrete and continuous versions of these problems are equal to each other *almost surely*. For the two-dimensional case, both these problems can be solved by placing the centers or medians in an especially simple regular hexagonal pattern (the "honeycomb heuristic" of Papadimitriou). This yields the exact asymptotic values for the k center and median problem, namely, $\alpha(|A|/k)^{1/2}$ and $\beta(|A|/k)^{1/2}$ where $|A|$ denotes the volume of A, α and β are known constants, and the objective of the median problem is given in terms of the average, rather than the usual total, distance. For the three- and four-dimensional case, similar results can be obtained for the center problem to within an accuracy of roughly one percent. As a byproduct, we also get asymptotically optimal algorithms for the two-dimensional p-norm k median problem and for the twin problems of minimizing the maximum number of vertices served by any center and similarly for maximizing the minimum.

Key words. geometric location problems, probabilistic analysis, heuristics, k center, k median

AMS subject classifications. 68A, 90C

Introduction. Many NP-complete optimization problems whose worst case asymptotic behavior is dismaying tend to have asymptotic *average* complexity which is very satisfactory. In particular, this is the case for various geometric optimization problems where the data consists of the Euclidean (or other L_p norm) distances between vertices randomly scattered in a region A of the d-dimensional space. The first algorithm of this type is Karp's work on the travelling salesman problem [11], [12] which builds on the findings of Beardwood, Halton and Hammersley [3] concerning the asymptotic value of the objective function. Since then, several new algorithms and results of this type have emerged. In particular, the k median problem was analyzed by Fisher and Hochbaum [10] and Papadimitriou [16]. Also, a general technique for analyzing the optimal solution values was developed by Steele [19] using the concept of subadditive functionals.

In this paper we discuss in a unified way the k *center* and *median* problems for n points scattered uniformly and independently inside a region A of R^d, under very mild qualifications concerning this region. The motivation for this work, as well as several important elements in the discussion below, are due to Papadimitriou's work on the k median problem inside a *square* region A of the two-dimensional plane [16]. The main result of Papadimitriou's paper is that for k which grows slower than $o(n/\log n)$, one can solve the k median problem by placing the medians in a simple regular hexagonal pattern (the "honeycomb heuristic"). He has shown that the relative error of this heuristic tends to zero with probability one when n tends to infinity. This result is based on two main observations. First, when a large number of points are scattered uniformly and independently in a square region $A \subseteq R^2$, there is a close relationship between the solutions of the discrete and continuous problems. Second,

* Received by the editors September 29, 1982, and in final revised form November 1, 1983. This research was supported, in part, by the National Science Foundation under grant ECS-8121741, and by the Israel Institute of Business Research, Tel Aviv University. Part of this work was done when the author was visiting Tel Aviv University. This paper was presented at the I.I.S.O. Conference on Stochastics and Optimization, Gargnano, Italy, September 1982.

† J. L. Kellogg Graduate School of Management, Department of Managerial Economics and Decision Sciences, Northwestern University, Evanston, Illinois 60201.

for large k, the solution of the continuous problem can be very closely approximated by the above-mentioned hexagonal pattern. We can strengthen and generalize these observations in several directions. We show that the first observation is valid not only for median problems in a square region of R^2, but is satisfied also for both *median* and *center* problems within an arbitrary region of R^d (under very mild qualifications). Furthermore, the closeness of the continuous and discrete versions can be asserted in the strong sense of *almost sure convergence* as opposed to the weaker notion of *convergence in probability* used in [16]. (On the difference between these two notions, and the relevance of the stronger one to optimization problems such as are considered here, the reader is referred to [17] and [19].) The second observation of [16], namely the optimality of the hexagonal pattern for the k median problem in a square region A of R^2, can also be generalized. In fact, both median and center problems in a general region of R^2 (under the same mild qualifications) are solved with high probability within arbitrary accuracy by the honeycomb heuristic. Thus, the main result of this paper for the *two-dimensional case* can be summarized in the following theorem:

MAIN THEOREM. *Let A be a compact region in R^2 of volume $|A|$ (and which satisfies some mild condition to be specified later). Let n points be distributed uniformly and independently in A. Let M^* and m^* be the optimal values of the k center and median problems respectively. Let $\alpha = (2/3\sqrt{3})^{1/2} = 0.620400\cdots$, $\beta = 0.377196\cdots$. Then, for $k \leqq k(n) = o(n/\log n)$:*

(a)
$$M^* = \alpha\left(\frac{|A|}{k}\right)^{1/2} \quad \textit{almost surely,}$$

$$m^* = \beta\left(\frac{|A|}{k}\right)^{1/2} \quad \textit{almost surely.}$$

(b) *The optimal asymptotic values indicated in part (a) are achieved almost surely by placing the centers or medians in a regular hexagonal pattern throughout A. (The same solution is optimal for both center and median problems.)*

The precise details and terms used in the statement of this theorem will be clarified in the subsequent sections. An almost as strong result for the three- and four-dimensional k center problem is also obtained. We note that while the result on the median in R^2 is a relatively straightforward generalization of Papadimitriou's work, the one concerning the center is new and is rather surprising. The objective function of the center problem is of a different type than that of the median problem or Travelling Salesman Problem in that it is not a minisum problem but rather is a minimax problem. Thus, its objective function is potentially sensitive to the location of every single point of V, in contrast to the median problem where we can ignore the position of a small subset of the points. As a consequence, in order to provide a heuristic with diminishing error for the center problem, one needs to take into account the location of *every* individual point inside A. This is accomplished (asymptotically) by Lemma 5. It is this lemma which also restricts the results of this paper to the range $k(n) = o(n/\log n)$. For higher values of growth of k, Papadimitriou provides a different heuristic for the median problem in R^2. However, for the reasons just outlined, his methods do not seem to carry over to the center problem.

While the continuous problem is well solved for the two-dimensional plane, we have only partial results for higher dimensions. Almost-optimal solutions for the center problem can be obtained from the results of [1], [4], [7], [13], [18] concerning the covering of R^d by spheres. However, it is not known whether the solutions obtained are actually optimal and whether the optimal solutions for median and center problems

still coincide in higher dimensional spaces, as they do in the case of the R^2. Several other geometric location problems for which the continuous version is solvable could be addressed using the same methodology. In particular, we consider the p-norm k median problem and the twin problems of minimizing the maximum number of vertices served by any center and similarly for maximizing the minimum.

2. Preliminaries. Let A be an arbitrary but fixed compact region in R^d of volume $|A|$ and let $V=\{v_1, \cdots, v_n\}$ be a given set of points in A. For every pair of two points, x, y in R^d, let $d(x, y)$ be the Euclidean (L_2) distance between them. Similarly, for $y \in R^d$ and for each set $X=\{x_1, \cdots, x_k\}$ of points in R^d let $d(y, X)=\min_{x_i \in X} d(y, x_i)$. Let

$$M_A(X)=\max_{y \in A} d(y, X) \tag{1}$$

and

$$m_A(X)=\frac{1}{|A|}\int_A d(y, X)\, dy \tag{2}$$

be the *maximum* and *average* distance respectively of a point of A from X. Similarly, we can define maximum and average distances with respect to V, namely

$$M_V(X)=\max_{v_i \in V} d(v_i, X), \tag{3}$$

$$m_V(X)=\frac{1}{n}\sum_{v_i \in V} d(v_i, X). \tag{4}$$

The discrete (or, more precisely, *discrete supply/discrete demand*) k center and median problems with respect to V seek a subset $V' \subseteq V$ of cardinality k which minimizes $M_V(V')$ and $m_V(V')$ respectively. We denote the optimal objective values of these problems by M_{VV} and m_{VV} respectively. Three additional versions of the center or median problem can be naturally defined—namely,

$$M_{AA}=\min_{\substack{X \subseteq A \\ |X|=k}} M_A(X), \tag{5}$$

$$M_{AV}=\min_{\substack{X \subseteq A \\ |X|=k}} M_V(X), \tag{6}$$

$$M_{VA}=\min_{\substack{V' \subseteq V \\ |V'|=k}} M_A(V'), \tag{7}$$

and similarly for the median. We refer to these problems as the *continuous supply/continuous demand*, *continuous supply/discrete demand*, and *discrete supply/continuous demand* problems respectively. All four versions of both median and center problems are either prime suspects or known to be, NP-hard. (See Papadimitriou [16] for m_{VV}, Megiddo and Supowit [14] for M_{AV} and m_{AV}, Nasuyama et al. [14] for M_{AV} and M_VV.) Our main interest in this paper is the relationship between the optimal solutions to these four versions when n is very large.

Our starting point in this discussion is the optimal solution to the continuous supply/continuous demand problems (or, "continuous problems" for short). Note that the optimal values for these problems, m_{AA} and M_{AA}, respectively, depend on A but not on V. Lemmas 1 and 2 below yield useful lower and upper bounds on these optimal values:

LEMMA 1. *There exist positive constants* γ *and* δ (*which depend on the dimension* d *but not on* k *or* A) *such that*

$$M_{AA} \geqq \gamma\left(\frac{|A|}{k}\right)^{1/d}, \tag{8}$$

$$m_{AA} \geqq \delta\left(\frac{|A|}{k}\right)^{1/d}. \tag{9}$$

Proof. The dependence on the volume $|A|$ follows from the homogeneity of the functions involved. Assume that $|A| = 1$. The bound on M_{AA} is obvious since from the definition of M_{AA} it follows that k d-dimensional spheres of radius M_{AA} each must cover the entire region A and thus their combined volume must at least equal 1. For the bound on m_{AA}, let $A_1, \cdots, A_k$ be the partition of A into Voronoi cells induced by the optimal solution $x_1, \cdots, x_k \in A$ which yield the value of m_{AA}. Thus

$$m_{AA} = \sum_{i=1}^{k} \int_{A_i} d(y, x_i)\, dy$$

clearly,

$$\int_{A_i} d(y, x_i)\, dy \geqq \int_{S_i} d(y, u_i)\, dy,$$

where S_i denotes a d-dimensional sphere whose volume equals $|A_i|$ and with u_i as its center. Let δ be the value of this integral for a sphere of volume 1. Thus,

$$m_{AA} \geqq \sum_{i=1}^{k} |A_i|^{(d+1)/d}\delta \geqq k \cdot \left(\frac{1}{k}\right)^{(d+1)/d} \delta = \frac{\delta}{k^{1/d}}$$

where the second inequality follows the fact that $\sum_{i=1}^{k} |A_i| = 1$.

Upper bounds on M_{AA} and m_{AA} of the same functional form but with constants $\bar{\gamma}$ and $\bar{\delta}$ which depend on A as well as on d can be easily derived (proof omitted):

LEMMA 2. *There exist positive constants* $\bar{\gamma}$ *and* $\bar{\delta}$ (*which depend on* A) *such that*

$$M_{AA} \leqq \bar{\gamma}\left(\frac{|A|}{k}\right)^{1/2}, \tag{10}$$

$$m_{AA} \leqq \bar{\delta}\left(\frac{|A|}{k}\right)^{1/d}. \tag{11}$$

3. The relation between discrete and continuous problems. Our general strategy is to approximate the optimal solution to a discrete problem by the solution to its continuous counterpart. Obviously, the quality of such approximation depends crucially on the extent to which the discrete set V is "spread evenly" throughout the entire region A. One measure of the extent of "evenness" is the maximal distance between an arbitrary point of A and its closest neighbor in V, $\Delta \equiv M_A(V)$. Obviously, when Δ is small, so is the difference between the continuous and discrete values:

LEMMA 3.

(a) *Center problem. The absolute value of the difference between the objective values of any two versions of the* k *center problem is bounded by* 2Δ.

(b) *Median problem.*

$$|m_{VV} - m_{AV}| \leqq \Delta,$$

$$|m_{AA} - m_{VA}| \leqq \Delta.$$

Proof. Consider two versions of the k median or center problem which share the same demand set (i.e., are both either continuous demand or discrete demand) but whose supply set is different. Consider the solution to the continuous supply problem, say $X=\{x_1, \cdots, x_k\}$, $x_i \in A$, $i=1, \cdots, k$. We can generate from X an approximate solution for the corresponding discrete supply problem by replacing each center $x_i \in A$ by its closest vertex $v_{j(i)} \in V$. Since by assumption, $d(x_i, v_{j(i)}) \leqq \Delta$, we get the assertion of part (b) together with the corresponding assertion on the center problem. To obtain the remaining results concerning center problems consider two problems with the same supply set (V or A) but with different demand sets. We note that for any supply point $x_i \in X$,

$$d(x_i, V) \leqq d(x_i, A) \leqq d(x_i, V) + \Delta,$$

so that the objective functions cannot vary by more than Δ.

In order for Lemma 3 to be useful, Δ must be small. We now proceed to assess the magnitude of Δ when n is large and V is scattered randomly in A. But first we have to make an additional assumption on the region A. As will shortly be revealed, the assumption is mild and is easily seen to be satisfied by any region of practical interest. We note that the assumption is necessary for some of the lemmas which follow; nevertheless, we believe the ultimate results of this paper are true in general.

We now state our assumption. Let $I(s)$ be an infinite grid of cubes of mesh size s, with faces parallel to the coordinate system. For any region A let $A^-(s)$ and $A^+(s)$ be the set of cubes of $I(s)$ which lie entirely in A and those which intersect A, respectively. Obviously, the volume of $A^-(s)$ and $A^+(s)$ tend to that of A (from below and above respectively) when s tends to zero. We require that these sets approximate A in a stronger sense.

Let $B_{x,r}$ be a sphere of radius r centered at x and let $A_{x,r} = A \cap B_{x,r}$. For a set $A \subseteq R^d$ let $|A|$ denote its volume.

Condition A. There exist positive constants r_0, γ, δ, m such that for $r \leqq r_0$, $s \leqq r/m$, and for each $x \in A$ we have

(a) $$|A^+_{x,r}(s)| \leqq \gamma r^d,$$

(b) $$|A^-_{x,r}(s)| \geqq \delta r^d.$$

It is straightforward to demonstrate that condition A is equivalent to the requirement that for each point $x \in A$, the regions $A_{x,r}$ and $B_{(x,r)}$ have volumes which are within a constant of each other and similarly for their surface areas. The condition eliminates from discussion regions A which contain "sections" which are less than full dimensional or where the ratio of surface area to volume is unbounded. The condition is satisfied by every compact convex region in R^d as well as by nonconvex regions of bounded curvature. In particular, it is satisfied by regions bounded by a finite number of planar faces. We call a compact region which satisfies condition A, *proper*, and restrict our attention in the sequel to proper sets without further mention of this qualification.

One simple consequence of condition A is extremely useful for bounding the magnitude of $\Delta = M_A(V)$.

LEMMA 4. *There exist positive constants s_0, γ_0 (which depend on A) such that for every $s \leqq s_0$*

$$M_{A^-(s)}(A) \leqq \gamma_0 s$$

i.e., every point $x \in A$ is within a distance of at most a constant number of cubes from $A^-(s)$.

Proof. Using the notation of condition A, let $r \leqq r_0$ and let $s < r/m$. Then $A^-_{(x,r)}(s)$ contains at least one cell. Thus, the lemma holds with $\gamma_0 = m$.

In order to couple Lemma 4 with Lemma 3, we have to find mesh size s which would ensure that each cube in $A^-(s)$ contains at least one vertex of V. Clearly, if this condition is satisfied, then we can bound Δ by $\alpha \cdot s$ for some appropriate positive constant α. To this end we study the number of points which fall in each cell of $A^-(s)$, when the set V is uniformly and independently scattered in A. Lemma 5 below indicates that for s values which are not too small (unsymptotically with n) each cube of $A^-(s)$ contains more or less its "fair share" of vertices of V, *almost surely.* A similar type of lemma, for a square region $A \subseteq R^2$, and using the weaker notion of *convergence in probability*, is central to Papadimitriou's paper [16] which motivated this research.

We note that Lemma 5 can be obviously tightened in various ways. However, the following simple version is sufficient for our purposes:

LEMMA 5. *Let n points be distributed uniformly and independently in a region $A \subseteq R^d$ of volume 1. Let $A_1, \cdots, A_{t(n)}$ be a partition of A into $t(n)$ equal volume disjoint subregions, and let n_i be the number of points in subregion A_i. Then, the following $t(n)$ inequalities hold simultaneously almost surely:*

$$\left| n_i - \frac{n}{t(n)} \right| \leqq \sqrt{12 \log n} \cdot \sqrt{\frac{n}{t(n)}}.$$

Proof. Let P_i^n denote the probablity that the ith inequality is violated and let P^n be the probability that one or more of these inequalities is violated. Obviously, $P^n \leqq \sum_{i=1}^{t(n)} P_i^n = t(n) P_1^n$. Note that under our assumptions, n_i is a binomial random variable with n trials and probability of success $1/t(n)$. It follows from the normal approximation to the binomial distribution, [17] that

$$P_1^n \leqq 2\, e^{-12 \log n/4} = 2\, e^{-3 \log n} = \frac{2}{n^3}.$$

Thus, $P^n \leqq 2/n^2$ and therefore $\sum_{n=1}^{\infty} P^n < \infty$.

It follows from Lemma 5 that if $t < n/12 \log n$ each subregion A_i, $i = 1, \cdots, t$ contains at least one vertex of V almost surely. Consider a grid of mesh size $s = \sqrt[d]{1/t}$. Each cube of $A^-(s)$ contains at least one vertex of V almost surely. From Lemma 4 we know that each point $x \in A$ is within a distance of αs from $A^-(s)$. Thus, we can conclude:

LEMMA 6. *There exists a positive constant γ_0 (which depends on A) such that*

$$M_A(V) \equiv \Delta \leqq \gamma_0 \left(\frac{\log n}{n} \right)^{1/d} \quad \textit{almost surely.}$$

Lemma 6 bounds the *absolute* error associated with the approximation of a solution of a discrete supply problem by the solution to its continuous supply counterpart. In order to get a bound on the *relative* error, we use the bound of Lemma 1. Let M^* be the optimal solution to any given version of the k center problem and let M_{App} be the objective function value for this problem obtained from the optimal solution for *any of the other three* versions. We have thus shown:

THEOREM 1. *Let $k \leqq k(n) = o(n/\log n)$. Then*

$$\frac{M_{App} - M^*}{M^*} \quad \textit{tends to zero almost surely with } n.$$

We now proceed to derive a similar result for median problems. Note that part (b) of Lemma 3 takes us part of the way towards that goal. However, we still need

to bound the difference between m_{VV} and m_{VA} and similarly between m_{AA} and m_{AV}. This requires some further study of the structure of the optimal solution for median problems. We open the discussion by investigation of the solution for the *continuous case.* Let $X=\{x, \cdots, x_k\}$ be the optimal location of medians for this problem. Lemma 7 below indicates that each point in X is "serving" a region A_i of a volume roughly equal to $|A|/k$. Let

$$R=\max_{y\in A} d(y, X),$$

$$r=\min_{1\leqq i<j\leqq n} d(x_i, x_j)$$

be the maximal distance between a point and its median, and the minimal distance between two medians respectively:

LEMMA 7. *There exist constants* k_0, α_1, α_2, β_1, β_2 (*which depend on* A) *such that for every* $k\geqq k_0$

$$\alpha_1\left(\frac{|A|}{k}\right)^{1/d}\leqq R\leqq \alpha_2\left(\frac{|A|}{k}\right)^{1/d},$$

$$\beta_2\left(\frac{|A|}{k}\right)^{1/2}\leqq r\leqq \beta_1\left(\frac{|A|}{k}\right)^{1/d}.$$

Proof. We may assume without loss of generality that $|A|=1$. Note that k spheres of radius R must cover the region A and hence their combined volume is at least one. This yields the constant $\alpha_2>0$ such that $R\geqq \alpha_2/k^{1/d}$. On the other hand, k spheres of radius r centered at the x_i's are mutually disjoint, and each has at least a given fraction, say δ, of its volume within A (condition A). Thus, the combined volume of these spheres cannot exceed $1/\delta$, and we get $r\leqq \beta_1/k^{1/d}$. To complete the proof we need to show that there exists a constant c_1 such that $r\geqq c_1 R$. Let c denote the volume of the unit sphere in R^d. Consider any two medians, x_1 and x_2 with $d(x_1, x_2)=r$. Let A_1 be the Voronoi cell served by x_1. By the definition of R, the volume of this cell does not exceed cR^d. Thus, the incremental cost of cancelling the center at x_1 (assigning the region A_1 to x_2) satisfies

$$\Delta_+\leqq c\cdot R^d\cdot r.$$

Now consider a point y whose distance from its center is R. Consider the reduction in cost obtained by establishing an additional center at y. Clearly all the region of A which is within a distance $R/3$ from y would be closer to y than to any of the old centers by at least $R/3$. Thus we obtain that the reduction in cost by such assignment satisfies

$$\Delta_-\geqq \delta\left(\frac{R}{3}\right)^{d+1}.$$

The required result follows from the optimality requirement $\Delta_-\leqq\Delta_+$.

Lemma 7 can be used to provide the following interesting result. Let $A_1, \cdots, A_n$ be the k Voronoi cells associated with an optimal solution $x_1, \cdots, x_n$. We say that A_i and A_j are *neighbors,* if they share a common $(d-1)$-dimensional face. Let m_i be the number of neighbors of A_i. The lemma asserts that m_i is bounded by a constant independent of k:

LEMMA 8. *There exists a positive constant* δ (*which depends on* A) *such that*

$$m_i\leqq\delta, \qquad i=1, \cdots, k.$$

Proof. Let r and R be as in Lemma 7. Consider a given Voronoi cell, say A_i. The faces of this cell are generated by the bisectors of the $k-1$ line segments $[x_i, x_j]$, $j \neq i$. By definition $d(y, X) \leqq R$ for every $y \in A$. Thus, all bisectors generated by points x_j, $j \neq i$, with $d(x_i, x_j) \geqq 2R$, do not contribute a face to A_i. Therefore, m_i is bounded by the number of distinct centers x_j, $j \neq i$ which are within a distance of $2R$ from x_i. Let this number be $\bar{m}_i$. Since $d(x_i, x_j) \geqq r$ for $i \neq j$, we get that $\bar{m}_i$ spheres of radius r centered at the neighbors of x_i are disjoint. The combined volume of these $\bar{m}_i$ spheres is $\bar{m}_i c r^d$ and all this volume is contained in a sphere of radius $2R + r$ centered at x_i. Thus,

$$\bar{m}_i c r^d \leqq c(2R + r)^d \leqq c[(2c_1 + 1)r]^d$$

where $R \leqq c_1 r$ as per the proof of Lemma 7. It now follows easily that $\bar{m}_i \leqq (2c_1 + 1)^d \equiv \delta$.

LEMMA 9. *There exists a constant $\theta > 0$ (which depends on A) such that the combined surface area of the Voronoi cells $A_1, \cdots, A_k$ is bounded by $\theta \cdot (|A|^{(d-1)/d} k)^{1/d}$.*

Proof. The surface area of A itself is finite, say S. In addition we have at most δk additional planar faces, where the longest dimension is bounded by $R \leqq \alpha_1 (|A|/k)^{1/d}$. Thus, the total additional surface area is bounded by

$$\delta k R^{d-1} \leqq \delta \alpha_1^{d+1} \cdot k \left(\frac{|A|}{k} \right)^{(d-1)/d} \equiv \theta k^{1/d} \cdot |A|^{(d-1)/d}.$$

Lemmas 7–9 relate to the continuous supply/continuous demand median problem. However, analogous assertions are valid *almost surely* for the other three versions of the k median problem as well. In order to demonstrate this, it suffices to show that assertions analogous to this of Lemma 7 are valid almost surely since Lemmas 8 and 9 are simple consequences of Lemma 7. But the proof of Lemma 7 can be easily adapted to yield the required result by using Lemma 10 below which essentially states that, for large enough spheres, the volume of a given sphere and the number of vertices of V within it, are more or less proportional to each other. We use the notation $f(n) = \Omega(g(n))$ to indicate the relation $\lim_{n \to \infty} f(n)/g(n) = \infty$.

LEMMA 10. *There exist positive constants θ_1, θ_2 such that if*

$$r \geqq r(n) = \Omega\left[\left(\frac{\log n}{n} \right)^{1/d} \right],$$

the number n_r of vertices of V in any sphere of radius r centered at a point of A satisfies almost surely

$$\theta_1 n r^d \leqq n_r \leqq \theta_2 n r^d.$$

Proof. Let $s(n) = \gamma(\log n/n)^{1/d}$ for some $\gamma > \sqrt[d]{12}$ as per Lemma 5. Consider a sphere $B_{x,r}$ of radius r centered at $x \in A$. The number of vertices of V inside $B_{x,r}$ is bounded from above and from below by the number of vertices within $A^+_{x,r}(s)$ and $A^-_{x,r}(s)$ (see condition A for definitions). Condition A ensures that the number of cells within each of these two sets is bounded from above by $\gamma(r/s)^d$ and from below by $\delta(r/s)^d$ respectively. Lemma 5 ensures that each one of these cells contains at least $c_1 n s^d$ and at most $c_2 n s^d$ vertices of V almost surely. The required result now follows by substitution.

Using Lemma 10, the reader should have no difficulty verifying that Lemmas 7–9 can be generalized for all four versions of the k median problem. In the sequel we

refer to these lemmas in this wider context. We now come to the second main theorem of this secton. Let m^* be the optimal solution to *any* version of the k-median problem and let m_{App} be the objective function value for this problem obtained from the optimal solution for any of the other three versions:

THEOREM 2. *Let* $k \leqq k(n) = o(n/\log n)$. *Then*

$$\frac{m_{App} - m^*}{m^*} \quad \textit{tends to zero almost surely with } n.$$

Proof. We consider a region of volume 1. Part (b) of Lemma 3 takes care of errors generated by the difference between discrete supply and continuous supply problems. We now assess the difference between the optimal values of any two problems which vary from each other by their *demand* set. Since Lemmas 7–9 apply to all four problems, it suffices to examine just one case. Let $X = \{x_1, \cdots, x_k\}$ be an optimal solution to the continuous-continuous problem with corresponding Voronoi cells $A_1, \cdots, A_k$, and objective function value:

$$m^* = \sum_{i=1}^{k} \int_{A_i} d(x_i, y)\, dy$$

(i.e., $m^* = m_{AA}$). Let

$$\bar{m} = \frac{1}{n} \sum_{i=1}^{k} \sum_{v_j \in A_i} d(x_i, v_j) = m_X(v)$$

be the value of this solution when assessed with respect to the discrete demand version. Let

$$\gamma = |m^* - \bar{m}|,$$

for we wish to bound the magnitude of γ.

Consider a grid $I(s)$ with $s = \sqrt[d]{1/t}$ where t is to be specified later. Let I^+ and I^- denote respectively the index sets of the cubes of $I(s)$ which fall in $A^+(s)$ and $A^-(s)$. Similarly, for each Voronoi cell A_i, $i = 1, \cdots, k$, we let I_i^+, and I_i^- denote the index sets for the cubes in $A_i^+(s)$ and $A_i^-(s)$ respectively. Let a_{ij} be the volume of cube c_j inside cell A_i and let n_{ij} be the number of vertices there. Finally, for each cell c_j, $j \in N^+$, let k_j be the center point of the cell. Obviously, there exists a constant $\delta > 0$ such that

$$\left| m^* - \sum_{i=1}^{k} \sum_{j \in I_i^+} a_{ij} d(k_j, x_i) \right| \leqq \delta s$$

and similarly,

$$\left| \bar{m} - \sum_{i=1}^{k} \sum_{j \in I_i^+} \frac{n_{ij}}{n} d(k_j, x_i) \right| \leqq \delta s.$$

To complete our assessment of $\gamma = |m^* - \bar{m}|$, we will need a bound on

$$\gamma_0 = \sum_{i=1}^{k} \sum_{j \in I_i^+} \left| \frac{n_{ij}}{n} - a_{ij} \right| d(k_j, x_i).$$

We note that by Lemma 7, $d(k_j x_j)$ is almost surely bounded by $\alpha_1(1/k^{1/d})$. We now

assess the magnitude of

$$\gamma_1 = \sum_{i=1}^{k} \sum_{j=I_i^+} \left| \frac{n_{ij}}{n} - a_{ij} \right| \leqq \sum_{i=1}^{k} \sum_{j \in I_i^-} \left| \frac{n_{ij}}{n} - a_{ij} \right| + \sum_{i=1}^{k} \sum_{j \in I_i^+ I_i^-} \left| \frac{n_{ij}}{n} + a_{ij} \right|$$

$$\leqq \sum_{j \in I^-(s)} \left| \frac{n_{ij}}{n} - s^d \right| + \sum_{i=1}^{k} \sum_{j \in I_i^+ I_i^-} \left(\frac{n_{ij}}{n} + a_{ij} \right)$$

$$\leqq \sum_{j \in I^-} \sqrt{\frac{12(\log n) s^d}{n}} + \sum_{i=1}^{k} p_i (c_1 s^d + s^d)$$

where p_i is the number of cells broken by the boundary of region A_i and c_1 is a positive constant. By Lemmas 8 and 9, we know that

$$\sum_{i=1}^{k} p_i \leqq \delta \cdot k^{1/d} / s^{d-1}$$

for some constant δ. Thus, we get constants δ_1 and δ_2 such that

$$\gamma_1 \leqq \delta_1 \sqrt{\frac{\log n}{n s^d}} + \delta k_2^{1/d} s$$

and therefore for positive constants δ_3, δ_4

$$\gamma_0 \leqq \delta_3 \sqrt{\frac{\log n}{n \cdot s^d}} \cdot \frac{1}{k^{1/d}} + \delta_4 s.$$

Note that γ has the same asymptotic form. Now choose t such that $s = \sqrt[d]{1/t}$ satisfies $s = o((1/k)^{1/d})$, $s = \Omega((\log n/n)^{1/d})$. This choice ensures that $\gamma = O((1/k)^{1/d})$ and the result now follows Lemma 1.

4. Solution of the continuous problem. We are finally faced with the last phase in our chain of approximations. Theorems 1 and 2 enable us to approximate the solution of the discrete problem by the solution to its continuous counterpart. We now wish to solve the continuous problem. For the two-dimensional plane, the situation is especially attractive. Both center and median problems can be solved by the *same* heuristic, namely, placing the centers $x_1, \cdots, x_k$ in a regular hexagonal pattern. The optimality of the hexagonal pattern for various continuous location problems over the entire 2-D plane has been known for some time, e.g., [5], [6], [13]. The asymptotic optimality of this pattern for the median in a square region of R^2 is proven by Papadimitriou [15]. His proof can be generalized to any proper region $A \subseteq R^2$ for both median and center problems. We summarize these results in the following theorem which we bring here without a proof. The constants $\alpha = \sqrt{2}/3\sqrt{3} = 0.6204003 \cdots$ and $\beta = 0.3771967 \cdots$ used in this theorem correspond to the optimal values of the continuous *one* center and median problems respectively in the regular hexagon of area one.

THEOREM 3. *Let* $A \subseteq R^2$, $k \leqq k(n) = o(n/\log n)$. For every $\varepsilon > 0$, *there exists* $k_0 > 0$ *such that for* $k \geqq k_0$ *the following relations hold almost surely*:

(a)
$$(\alpha + \varepsilon)\left(\frac{|A|}{k}\right)^{1/2} \geqq M_{AA} \geqq (\alpha - \varepsilon)\left(\frac{|A|}{k}\right)^{1/2},$$

$$(\beta + \varepsilon)\left(\frac{|A|}{k}\right)^{1/2} \geqq m_{AA} \geqq (\beta - \varepsilon)\left(\frac{|A|}{k}\right)^{1/2}.$$

(b) *The lower bounds of part* (a) *are realized by the "honeycomb" heuristic.*

For higher dimensions, the results are somewhat weaker. Nevertheless, for $d=3$ or 4, we can use the findings of Bambah, Barnes, Few and Coxeter, Few and Rogers [1], [2], [7] (see [18] for a summary) concerning the density of covering of space by spheres. Let r_d denote the radius of a sphere of volume one in R^d. Let $\alpha_3 = 1.1268\cdots$, $\bar{\alpha}_3 = 1.1354\cdots$, $\alpha_4 = 1.1347\cdots$, $\bar{\alpha}_4 = 1.1526\cdots$. The reader may note that for $d=3$, 4, α and $\bar{\alpha}$ are within 1% or so from each other.

THEOREM 4. *Let $A \subseteq R^d$, $d=3$ or 4, $k \leqq k(n) = o(n/\log n)$. For every $\varepsilon > 0$ there exists $k_0 > 0$ such that for $k \geqq k_0$ the following relation holds almost surely*:

$$(\alpha_d r_d - \varepsilon)\left(\frac{|A|}{d}\right)^{1/d} \leqq M_{AA} \leqq (\bar{\alpha}_d r_d + \varepsilon)\left(\frac{|A|}{d}\right)^{1/d}.$$

It seems likely that Theorem 4 can be generalized to higher than 4 dimensions and to the median problem as well. It is interesting to check whether an exact asymptotic value for the objective function exists for 3, 4 or higher dimensions, and whether the optimal solutions for the median and center problem coincide as they do in R^2.

5. Extensions. The results of this paper can be obviously generalized to various other geometric location problems for which the continuous problem is solvable. A particularly straightforward example is that of minimizing the maximum number of vertices in each Voronoi cell or maximizing the minimum. More specifically, let $X = \{x_1, \cdots, x_k\}$, and let $A_1, \cdots, A_k$ be the set of induced Voronoi cells. Let n_i be the number of vertices of V in A_i. Let

$$N_{VV} = \min_{\substack{V' \subseteq V \\ V' = |k|}} \max_{i=1,\cdots,k} n_i,$$

$$n_{vv} = \max_{\substack{V' \subseteq V \\ V' = k}} \min_{i=1,\cdots,k} n_i.$$

Obviously, $N_{VV} \geqq n/k$, $n_{VV} \leqq n/k$. Theorem 5 below indicates that for $k \leqq k(n) = o(n/\log n)$, these bounds can actually be achieved. (Any positioning of the x_i's such that the volumes of the cells A_i are "close" to $|A_i|/k$ will do; in particular, the centers can be chosen so that these cells correspond to identical cubes of $I(s)$ for some appropriate s.)

THEOREM 5. *Let $A \subseteq R^d$, $k \leqq k(n) = o(n/\log n)$. For every $\varepsilon > 0$ there exists $k_0 > 0$ such that for $k \geqq k_0$ the following relations hold almost surely*

$$\frac{n}{k} \leqq N_{VV} \leqq \frac{n}{k}(1+\varepsilon),$$

$$(1-\varepsilon)\frac{n}{k} \leqq n_{VV} \leqq \frac{n}{k}.$$

The proof of Theorem 5 follows easily from Lemma 5.

We finally mention the p-norm k median problem, namely the problem

$$Z_p = \min_{\substack{V' \subseteq V \\ |V'| = k}} \left(\sum_{v_i \in V} \min_{v_j \in V'} d(v_i v_j)^p \right)^{1/p}.$$

It is straightforward to demonstrate that the analogues of Theorems 1 and 2 are valid for this problem as well. Also, for the two-dimensional case, we have the following analogue of Theorem 3.

THEOREM 6. *Let* $A \subseteq R^2$, $k \leqq k(n) = o(n/\log n)$. *For every* $\varepsilon > 0$ *there exists* $k_0 > 0$ *such that for* $k \geqq k_0$ *the following relations hold almost surely*:

$$(1-\varepsilon)c_p n^{1/p}\left(\frac{|A|}{k}\right)^{1/d} \leqq Z_p \leqq (1+\varepsilon)c_p n^{1/p}\left(\frac{|A|}{k}\right)^{1/d}$$

where c_p *is the value of the continuous p-norm* <u>one</u> *center problem inside a regular hexagon of area one.*

REFERENCES

[1] R. P. BAMBAH, *On lattice coverings by spheres*, Proc. Nat. Inst. Sci. India, 20 (1954), pp. 25-52.

[2] E. S. BARNES, *The coverings of space by spheres*, Canad. J. Math., 8 (1957), pp. 293–304.

[3] J. BEARDWOOD, J. H. HALTON AND J. M. HAMMERSLEY, *The shortest path through many points*, Proc. Cambridge Philo. Soc., 55 (1959), pp. 299–327.

[4] H. S. M. COXETER, L. FEW AND C. A. ROGERS, *Covering space with equal spheres*, Mathematica, 6 (1959), pp. 247–257.

[5] L. FEJES TOTH, *Lagerungen in der Ebene, auf der Kugel und im Raum*, Berlin, 1953.

[6] ———, *Sum of moments of convex polygons*, Acta Math. Acad. Scient. Hungaricae, 24(3–4), (1973), pp. 417–421.

[7] L. FEW, *Covering space by spheres*, Mathematica, 7 (1960), pp. 136–139.

[8] M. L. FISHER AND D. S. HOCHBAUM, *Probabilistic analysis of the Euclidean K-median problem*, Math. Oper. Res., to appear.

[9] P. J. HEAWOOD, *Map colour theorem*, Quart. J. Math., 24 (1890), pp. 332–338.

[10] D. S. HOCHBAUM, *The probabilistic asymptotic properties of some combinatorial geometric problems*, manuscript, Carnegie-Mellon Univ., Pittsburgh, November, 1979.

[11] R. M. KARP, *The probabilistic analysis of some combinatorial search algorithms, in* Algorithms and Complexity: New Directions and Recent Results, J. F. Traub, ed., Academic Press, New York, 1977.

[12] ———, *Probabilistic analysis of partitioning algorithms for the Travelling Salesman Problem*, Math. Oper. Res., 2 (1977), pp. 209–224.

[13] R. KERSHNER, *The number of circles covering a set*, Amer. J. Math., 61 (1939), pp. 665–671.

[14] S. MASUYAMA, T. IBARAKI AND T. HASEGAWA, *The computational complexity of the m-center problem on the plane*, Trans. IECE of Japan, 64(2) (1981), pp. 57–64.

[15] N. MEGIDDO AND K. J. SUPOWIT, *On the complexity of some common geometric location problems*, Manuscript, 1981.

[16] C. H. PAPADIMITRIOU, *Worst case and probabilistic analysis of a geometric location problem*, SIAM J. Comput., 10(3) (1981), pp. 542–557.

[17] A. RENYÉ, *Foundations of Probability*, Holden-Day, San Francisco, 1970.

[18] C. A. ROGER, *Packing and Covering*, Cambridge Tracts in Mathematics and Mathematical Physics, 1964.

[19] MICHAEL J. STEELE, *Subadditive Euclidian functionals and non-linear growth in geometric probability*, Ann. Probab., 9 (1981), pp. 365–376.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 2, April 1985

THE EFFECT OF THE PERTURBATION OF HERMITIAN MATRICES ON THEIR EIGENVECTORS*

JOHN DE PILLIS† AND MICHAEL NEUMANN‡

Abstract. We show that under some appropriate normalization, the eigenvectors corresponding to the maximal and minimal eigenvalues of a hermitian matrix subjected to a small perturbation by a positive semidefinite matrix decrease and increase in length, respectively. It is also shown that an eigenvector of a general matrix corresponding to an eigenvalue which increases in modulus must, if normalized in some particular fashion, eventually decrease in length if the matrix undergoes a sufficiently large perturbation.

AMS(MOS) subject classification. 15A18

1. Introduction. Despite the broad literature on the theory and applications of the spectral properties of hermitian matrices, there do not seem to be many papers found specifying the effect on the eigenvectors of the perturbation of a hermitian matrix by a positive semidefinite matrix. In a sequence of papers by Davis [1], [2] and by Davis and Kahan [3], and in Parlett's book [5], studies are conducted of the angular gap between the eigenspaces corresponding to, say, the maximal eigenvalue of the unperturbed matrix and the maximal eigenvalue of the perturbed matrix, respectively. Most of the results in these works either do not require that the eigenvectors under examination be normalized in some particular fashion or, if normalization is applied, then the (euclidean) length of the eigenvector is set to unity.

In this paper we wish to point several results concerning the effect of the perturbation of hermitian matrices on their eigenvectors in the following directions: 1) If the length of a component or of a group of components of the eigenvector is held fixed, how does the length of the vector formed from the remaining entries of the eigenvector behave as a function of the perturbation parameter. 2) If all the components of the eigenvector are allowed to vary in some controlled fashion, how does the length of the eigenvector behave as a function of the perturbation parameter. Roughly speaking, it will be shown that under some moderate restrictions on the perturbation matrices and subject to an appropriate normalization, the length of the eigenvector corresponding to $\lambda_{\max}$ reduces as the hermitian matrix is perturbed in the positive semidefinite direction, while the length of the eigenvector corresponding to $\lambda_{\min}$ increases when the hermitian matrix is subjected to a similar type of perturbation and when the eigenvector is similarly normalized.

In Theorems 2, 3 and 4 we shall assume that the eigenvalues $\lambda_{\min}$ and $\lambda_{\max}$ of each member of a family $A(t)$, $t \in J$, of hermitian matrices are simple. Under these assumptions, Wilkinson [6, pp. 66-67] shows that, subject to certain arbitrary, but fixed, normalization strategies, the components of the corresponding eigenvectors are analytic functions of the matrix entries in some open set in $C^{n,n}$ containing $A(t)$, $t \in J$.

In Theorem 2 we exhibit the results indicated in the opening paragraph for the perturbation in a single diagonal entry. In Theorems 3 and 4 we consider much more general perturbations. Theorems 1 through 4 are given in § 2 while numerical examples illustrating our findings are given in § 3.

* Received by the editors June 7, 1983, and in revised form December 8, 1983.

† Department of Mathematics, University of California, Riverside, California 92521.

‡ Department of Mathematics, University of California, Riverside, California 92521 and Department of Mathematics and Statistics, University of South Carolina, Columbia, South Carolina 29208. The work of this author was supported in part by University of South Carolina Research and Productive Scholarship Grant 13060 E 132.

Theorems 2 through 4 describe the effect of the perturbation on the eigenvectors locally. In Theorem 1 we show that if $\lambda(t)$ is an eigenvalue of an $n \times n$ matrix $A(t)$ such that $|\lambda(t)|$ increases with t, then eventually the corresponding eigenvector subject to an appropriate normalization decreases in length.

For brevity in the presentation, we have stated results here for perturbations by positive semidefinite matrices. Parallel results can be stated for perturbations by negative semidefinite matrices. In addition, our results here can be generalized to infinite-dimensional settings.

2. The main results. As mentioned in the introduction, our first result is concerned with the eventual effect of the perturbation on certain of its eigenvectors of a general matrix.

THEOREM 1. *Suppose that*

$$A(t) = \begin{pmatrix} B(t) & C \\ D & E \end{pmatrix} \tag{2.1}$$

are $n \times n$ matrices, where $B(t)$ are $k \times k$, $k < n$, matrices whose elements are continuous functions in t on the interval $[a, \infty)$. Let $\lambda(t)$ be an eigenvalue of $A(t)$ and let

$$x(t) = \begin{pmatrix} y(t) \\ z(t) \end{pmatrix}$$

be a corresponding eigenvector partitioned in conformity with (2.1) *and normalized such that if $y(t) \neq 0$, then $\|y(t)\| \leqq M$ for some fixed constant $M > 0$. If $|\lambda(t)| \to \infty$ as $t \to \infty$, then*

$$\lim_{t \to \infty} \|z(t)\| = 0. \tag{2.2}$$

Proof. For $t \in [a, \infty)$ let $r_t := \|z(t)\|$. If for some $t_0 \in [a, \infty)$, $r_t = 0$ for all $t \geqq t_0$ there is nothing to prove. Assume therefore that for any $t \in [a, \infty)$ such that $r_t = 0$, there exists a $t_0 > t$ such that $r_{t_0} \neq 0$. Let

$$S = \{t | t \in [a, \infty) \text{ and } r_t \neq 0\}.$$

S is clearly an unbounded set. For $t \in S$ define the $(n-k)$-vector

$$\tilde{z}(t) := z(t)/r_t,$$

so that $\|\tilde{z}(t)\| = 1$ and $z(t) = r_t \tilde{z}(t)$. Then from the eigenvalue-eigenvector relation for $t \in S$, namely,

$$\begin{pmatrix} B(t) & C \\ D & E \end{pmatrix} \begin{pmatrix} y(t) \\ r_t \tilde{z}(t) \end{pmatrix} = \lambda(t) \begin{pmatrix} y(t) \\ r_t \tilde{z}(t) \end{pmatrix}$$

we have that

$$\|Dy(t) + r_t E\tilde{z}(t)\| = r_t |\lambda(t)| \, \|\tilde{z}(t)\| = r_t |\lambda(t)|, \tag{2.3}$$

in which case on dividing both sides of (2.3) by r_t we obtain that

$$\left\| \frac{1}{r_t} Dy(t) + E\tilde{z}(t) \right\| = |\lambda(t)|.$$

Now let $t \to \infty$ through values in S. As $|\lambda(t)| \to \infty$ and as $\|Dy(t)\|$ and $\|E\tilde{z}(t)\|$ are bounded for all $t \in S$, we must have, in fact, that $Dy(t) \neq 0$ for $t \in S$ sufficiently large and that $r_t \to 0$ as $t \in S$ tends to ∞. Thus because $r_t = 0$ outside S, the limit (2.2) is valid. □

In contrast to the above results, all our subsequent statements are concerned with the local effect of the perturbation on the eigenvector. In the next theorem we shall assume that *anytime an eigenvector has its first component nonzero, then that eigenvector has been normalized so that its first component is 1.* We mention that both the theorem and its proof are in the spirit of the results in Elsner, Johnson and Neumann [4].

THEOREM 2. *Let*

$$A(t)=\begin{pmatrix} f(t) & C \\ C^* & D \end{pmatrix} \tag{2.4}$$

be a family of $n\times n$ hermitian matrices, where $f(t)$ is a strictly increasing differentiable function in some interval J. Suppose that $\lambda(t):=\lambda_{\max}(A(t))$ and $\mu(t):=\lambda_{\min}(A(t))$ are simple eigenvalues of $A(t)$ and suppose that their corresponding eigenvectors $x(t)$ and $y(t)$, respectively, have a nonzero first component throughout J. Then:

(i) $\|x(t)\|_2$ *is decreasing in t over J, and*

(ii) $\|y(t)\|_2$ *is increasing in t over J.*

Proof. We shall let $\bar{x}(t)$ and $\bar{y}(t)$ denote the vectors formed from the 2nd through nth components of $x(t)$ and $y(t)$, respectively.

(i) For $t\in J$ consider the eigenvector relationship

$$(A(t)-\lambda(t)I)x(t)=0. \tag{2.5}$$

Differentiating both sides of (2.5) with respect to t and rearranging yields that

$$(A(t)-\lambda(t)I)x'(t)=\lambda'(t)x(t)-A'(t)x(t). \tag{2.6}$$

We next argue that $\lambda'(t)>0$ throughout J, an observation which will be needed further along. On premultiplying both sides of (2.6) by $x^*(t)$ we obtain that

$$\lambda'(t)\|x(t)\|_2^2-f'(t)=\lambda'(t)\|x(t)\|_2^2-f'(t)(x_1(t))^2=0,$$

where $x_1(t)$ denotes the first component of $x(t)$. Then

$$\lambda'(t)=\frac{f'(t)}{\|x(t)\|_2^2}>0.$$

Next, as $x_1'(t)=0$ for $t\in J$, from (2.4) and (2.6) we observe that

$$(D-\lambda(t)I)\bar{x}'(t)=\lambda'(t)\bar{x}(t). \tag{2.7}$$

Because of the interlacing properties of the eigenvalues of hermitian matrices, $D-\lambda(t)I$ is negative semidefinite and so, on premultiplying both sides of (2.6) by $(\bar{x}'(t))^*$ we have that

$$\lambda'(t)(\bar{x}(t))^*\bar{x}(t)=(\bar{x}'(t))^*(D-\lambda(t)I)\bar{x}'(t)\leqq 0.$$

Thus $(\bar{x}'(t))^*\bar{x}(t)\leqq 0$ since $\lambda'(t)>0$. Hence, recalling the $x_1(t)=1$, we conclude that $\|x(t)\|_2$ is decreasing in t over J.

(ii) Just as in part (i) we argue that $\mu'(t)>0$ throughout J. Next, in place of $\lambda(t)$ and $x(t)$ in (2.5) through (2.7), substitute $\lambda(t)$ and $y(t)$, respectively. Then, similarly to (2.7), we obtain that

$$(D-\mu(t)I)\bar{y}'=\mu'(t)\bar{y}(t). \tag{2.8}$$

This time $D-\mu(t)I$ is positive semidefinite and so premultiplying (2.8) by $(\bar{y}(t))^*$ yields that

$$\mu'(t)(\bar{y}'(t))^*\bar{y}(t)=(y'(t))^*[D-\mu(t)I]\bar{y}'(t)\geqq 0.$$

Thus, because $\mu'(t)>0$, $(\bar{y}'(t))^*\bar{y}(t)\geqq 0$ and so $\|y(t)\|_2$ is increasing in t over J. □

For convenience we have stated Theorem 2 with the perturbation occurring in the (1, 1) entry. It is clear that a perturbation in any *single* diagonal entry will yield similar effect on $x(t)$ and $y(t)$ provided that a similar normalization strategy is applied. Consider then the following conjecture.

Conjecture. Suppose that

$$A(t)=\begin{pmatrix} B(t) & C \\ C & D \end{pmatrix} \tag{2.9}$$

is a family of $n\times n$ hermitian matrices over an interval J with the following properties:

(i) $\lambda_{\max}(A(t))$ is a simple eigenvalue of $A(t)$ for each $t\in J$.

(ii) $B(t)$ are $k\times k$, $1<k<n$, positive semidefinite and $B(t_1)<B(t_2)$ for $t_1, t_2\in J$ with $t_1<t_2$.

(iii) $B(t)$ is differentiable throughout J.

Suppose that the first k entries of an eigenvector of $A(t)$ corresponding to $\lambda_{\max}(A(t))$ contain at least one nonzero entry for every $t\in J$ and let

$$z(t)=\begin{pmatrix} u(t) \\ w(t) \end{pmatrix}$$

be a corresponding eigenvector of $A(t)$ normalized so that $\|u(t)\|_2=1$. Then $\|z(t)\|_2$ is a decreasing function in t over J.

The conjecture, which if true would have provided a particular extension of Theorem 2, appears to be false as Example 1 of the next section demonstrates. However, as we shall show next, provided $A'(t)$ satisfies a mild nonnegativity condition on J (see condition (iv) of the following theorem) and provided $A''(t)$ exists and is positive semidefinite, $A(t)$ does possess an eigenvector whose length decreases with t subject to a modified strategy of normalization.

THEOREM 3. *Suppose that $A(t)$, $t\in[a, b]$, is a family of $n\times n$ hermitian matrices with the following properties:*

(i) *$\lambda(t):=\lambda_{\max}(A(t))$ is a simple eigenvalue of $A(t)$ throughout $J:=(a, b)$.*

(ii) *$A(t_1)\leqq A(t_2)$ for $t_1\leqq t_2$ with $t_1, t_2\in J$.*

(iii) *$A''(t)$ exists and is positive semidefinite throughout J.*

(iv) *If $x(t)$ is an eigenvector of $A(t)$ corresponding to $\lambda(t)$, then*

$$(x(t))^*A'(t)x(t)>0 \quad \text{for } t\in J. \tag{2.10}$$

Let $\phi(t)$ be a positive decreasing (nonincreasing) differentiable function on J and let $x(t)$ be an eigenvector of $A(t)$ corresponding to $\lambda(t)$ normalized so that

$$(x(t))^*A'(t)x(t)=\phi(t), \quad \{\phi'(t)<0 \text{ or } \phi'(t)\leqq 0 \text{ as the case may be}\}. \tag{2.11}$$

Then the length $\|x(t)\|_2$ is a decreasing (nonincreasing) function in t over $[a, b]$.

Proof. To facilitate the proof we shall assume that the real part of the components of $x(t)$ are forbidden to reverse their sign (e.g. through multiplication of $x(t)$ by -1) except when passing through the value 0. The assumption (i) together with the normalization (2.11) ensures then that $x(t)$ is differentiable with respect to t in J.

Throughout the proof we shall assume that $\phi'(t)<0$ in J. Consider then the eigenvalue-eigenvector relation on J,

$$(A(t)-\lambda(t)I)x(t)=0. \tag{2.12}$$

Differentiating both sides of (2.12) we obtain, after some rearranging, that

$$(A(t)-\lambda(t)I)x'(t)=\lambda'(t)x(t)-A'(t)x(t), \tag{2.13}$$

and premultiplying both sides of (2.13) by $x^*(t)$ yields

$$\lambda'(t)\|x(t)\|_2^2 = (x(t))^*A'(t)x(t).$$

Thus, because of assumption (iv), $\lambda'(t) > 0$. Continuing, as $A(t) - \lambda(t)I$ is negative semidefinite, it follows from (2.13) that

$$\lambda'(t)(x'(t))^*x(t) - (x'(t))^*A'(t)x(t) = (x'(t))^*(A(t) - \lambda(t)I)x'(t) \leqq 0. \tag{2.14}$$

But then, because by (2.11) and (iii),

$$2(x'(t))^*A(t)x(t) \leqq 2(x'(t))^*A'(t)x(t) + (x(t))^*A''(t)x(t) = \phi'(t) < 0, \tag{2.15}$$

we see that (2.14) and (2.15) yield

$$2(x'(t))^*x(t) < 0.$$

Hence $\|x(t)\|_2$ is decreasing throughout $[a, b]$. □

It may happen that the first derivative of $A(t)$ in the above theorem has $A'(t) \to 0$ as, say, $t \to a$. Then because $\phi(t) \neq 0$ is nondecreasing and so does not tend to 0 as $t \to a$, the eigenvector $x(t)$ satisfying the equality (2.11) will require scaling close to $t = a$ by ever increasing factors to maintain the equality. For stability purposes, this situation can be overcome by considering the eigenvectors corresponding to the maximal eigenvalues of the shifts $A(t) \to A(t) + tI$. Such shifts have no effect on the eigenvectors under consideration, but merely stabilize the computation of $x(t)$ as $t \to a$.

Theorem 3 contains an important special case to which we devote the following corollary.

COROLLARY. *Let A be an $n \times n$ hermitian matrix such that $\lambda_{\max}(A)$ is simple. Suppose that D is a positive semidefinite matrix such that $x^*Dx \neq 0$, where x is an eigenvector of A corresponding to $\lambda_{\max}(A)$. Set*

$$A(t) = A + tD$$

and let $\phi(t)$ be a positive decreasing (nonincreasing) differentiable function in some neighborhood J of $t = 0$. Then there exists a neighborhood $J' \subseteq J$ of $t = 0$ such that $\|x(t)\|_2$ decreases (does not increase) in t over J', where $x(t)$ is an eigenvector corresponding to $\lambda(t) = \lambda_{\max}(A(t))$ normalized so that

$$(x(t))^*Dx(t) = \phi(t). \tag{2.16}$$

Comments. a) An example illustrating the results of Theorem 3 and the corollary will be provided in § 3. We remark that under the assumption of Theorem 3, if $z(t)$ is an eigenvector of $A(t)$ such that

$$z(t) = \alpha(t)x(t)$$

for some differentiable function $\alpha(t)$, then $\|z(t)\|_2$ is decreasing in J if and only if

$$(z'(t))^*z(t) = \alpha'(t)\alpha(t)\|x(t)\|_2^2 + (\alpha(t))^2(x'(t))^*x(t) \leqq 0. \tag{2.17}$$

Suppose now that the family of hermitian matrices of Theorem 3 has the form

$$A(t) = \begin{pmatrix} B(t) & C \\ C^* & D \end{pmatrix}, \qquad t \in J, \tag{2.18}$$

where the $B(t)$'s are $k \times k$, $k < n$, matrices. Partition the eigenvector $x(t)$ of the theorem in conformity with (2.18) into

$$x(t) = \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix}.$$

Because of (2.13), $x_1(t) \neq 0$ throughout J. Define

$$z(t) = \frac{1}{\|x_1(t)\|_2} x(t) := \alpha(t)x(t). \tag{2.19}$$

Then, in reference to the Conjecture and (2.17), Example 3 shows that in general

$$(z'(t))^* z(t) = -(x_1'(t))^* x_1(t) \|x_2(t)\|_2^2 + \|x_1(t)\|_2^2 (x_2'(t))^* x_2(t) \not\leqq 0.$$

We therefore raise here the question of characterizing the situations for which hermitian matrices $A(t)$ of the form (2.18) and which satisfy the requirements of Theorem 3 have a family of eigenvectors normalized as in (2.18) and which satisfy $(z'(t))^* z(t) \leqq 0$ in J.

b) We observe that the function $\phi(t)$ in (2.16) can be chosen to be a positive constant, say k. In this case Corollary 1 determines that a reduction in the length of $x(t)$ is achieved as $x(t)$ traces a certain path on the quadratic surface $\zeta^* D\zeta = k$.

We close this section with a theorem parallel to Theorem 3 which gives conditions for the growth locally of the length of the eigenvector of $A(t)$ corresponding to $\lambda_{\min}(A(t))$. Because of the similarity of the proof of this theorem to that of Theorem 3, it will not be given here.

THEOREM 4. *Suppose that $A(t)$, $t \in [a, b]$, is a family of $n \times n$ hermitian matrices with the following properties*:

(i) *$\mu(t) := \lambda_{\min}(A(t))$ is a simple eigenvalue of $A(t)$ throughout $J := (a, b)$.*

(ii) *$A(t_1) \leqq A(t_2)$ for $t_1 \leqq t_2$ with $t_1, t_2 \in J$.*

(iii) *$A''(t)$ exists and is positive semidefinite throughout J.*

(iv) *If $y(t)$ is an eigenvector of $A(t)$ corresponding to $\mu(t)$, then $(y(t))^* A'(t) y(t) > 0$ for all $t \in J$.*

Let $\psi(t)$ be a positive increasing (nondecreasing) differentiable function on J and let $y(t)$ be eigenvector of $A(t)$ corresponding to $\mu(t)$ normalized so that

$$(y(t))^* A'(t) y(t) = \psi(t), \qquad [\psi'(t) > 0 \text{ or } \psi'(t) \geqq 0 \text{ as the case may be}].$$

Then the length $\|y(t)\|_2$ is an increasing (nondecreasing) function in t over J.

3. Numerical examples. To obtain numerical evidence for the results of the previous section we have examined a varied sample of examples using the MATLAB Package (written by the Department of Computer Science of the University of New Mexico at Albuquerque) on the VAX 11/750 with the UNIX Operating Syetem. Typically our examples were constructed by generating a random matrix C and then forming a hermitian matrix A by taking some matrix function of C^*C.

For the examples displayed in this section, we consider a perturbation of the form

$$A(t) = A + tD, \tag{3.1}$$

where A and D are the 6×6 positive definite and the 6×6 positive semidefinite matrices, respectively, given by

$$A = \begin{bmatrix} 0.8892 & 0.2833 & -0.0489 & 0.1850 & 0.1240 & -0.1716 \\ 0.2833 & 1.2114 & 0.1915 & 0.0762 & 0.0017 & 0.0797 \\ -0.0489 & 0.1915 & 0.6443 & 0.0589 & 0.0703 & 0.0146 \\ 0.1850 & 0.0762 & 0.0589 & 0.8243 & -0.3024 & -0.1150 \\ 0.1240 & 0.0017 & 0.0703 & -0.3024 & 0.7578 & -0.0691 \\ -0.1767 & 0.0797 & 0.0146 & -0.1150 & -0.0691 & 0.8234 \end{bmatrix}$$

and by

$$D = \begin{bmatrix} 1.2700 & 0.6611 & -0.2348 & -0.0397 & 0.0000 & 0.0000 \\ 0.0611 & 0.5712 & -0.3650 & 0.0794 & 0.0000 & 0.0000 \\ -0.2348 & -0.3650 & 0.6127 & -0.3000 & 0.0000 & 0.0000 \\ -0.0397 & 0.0794 & -0.3000 & 0.6823 & 0.0000 & 0.0000 \\ 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \\ 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \end{bmatrix}.$$

Example 1. This example provides a counterexample to the Conjecture of § 2. Let

$$x(t) = \begin{pmatrix} u(t) \\ w(t) \end{pmatrix}$$

be the eigenvector of (3.1) corresponding to $\lambda_{\max}(A(t))$, where $u(t)$ is a 4-vector and where $x(t)$ has been normalized so that its first entry is nonnegative and so that $\|u(t)\|_2 = 1$. Table 1 illustrates that for $A(t)$ of (3.1) and for

$$A(t)\begin{pmatrix} u(t) \\ w(t) \end{pmatrix} = \lambda_{\max}(A(t))\begin{pmatrix} u(t) \\ w(t) \end{pmatrix},$$

$\|w(t)\|_2$ is *increasing* in length.

TABLE 1

t	$\|w(t)\|_2$	t	$\|w(t)\|_2$
.0001	8.7870d−02	.01	8.9143d−02
.0002	8.7883d−02	.02	9.0405d−02
.0003	8.7896d−02	.03	9.1641d−02
.0004	8.7908d−02	.04	9.2849d−02
.0005	8.7921d−02	.05	9.4026d−02
.0006	8.7934d−02	.06	9.5172d−02
.0007	8.7947d−02	.07	9.6284d−02
.0008	8.7960d−02	.08	9.7360d−02
.0009	8.7973d−02	.09	9.8400d−02
.0010	8.7986d−02	.10	9.9401d−02
.0011	8.7999d−02	.11	1.0036d−01
.0012	8.8012d−02	.12	1.0128d−01
.0013	8.8025d−02	.13	1.0216d−01
.0014	8.8038d−02	.14	1.0299d−01
.0015	8.8051d−02	.15	1.0378d−01
.0016	8.8064d−02	.16	1.0453d−01
.0017	8.8077d−02	.17	1.0523d−01
.0018	8.8090d−02	.18	1.0588d−01
.0019	8.8103d−02	.19	1.0649d−01
.0020	8.8116d−02	.20	1.0705d−01

Example 2. In this example the normalization strategy applied to $x(t)$ in the previous example was altered to the normalization stipulated in Theorem 3 and the corollary. Specifically, here $x(t)$ is the eigenvector of the matrix $A(t)$ given in (3.1) corresponding to $\lambda_{\max}(A(t))$:

$$Ax(t) = \lambda_{\max}(A(t))x(t),$$

normalized so that

$$(x(t))^T A'(t)x(t) = (x(t))^T Dx(t) = \phi(t) = 1.$$

Table 2 illustrates the decreasingness in length of this eigenvector.

TABLE 2

t	$\|x(t)\|_2$	t	$\|x(t)\|_2$
.0001	1.2372d+00	.01	1.2273d+00
.0002	1.2371d+00	.02	1.2174d+00
.0003	1.2370d+00	.03	1.2077d+00
.0004	1.2369d+00	.04	1.1980d+00
.0005	1.2368d+00	.05	1.1885d+00
.0006	1.2367d+00	.06	1.1792d+00
.0007	1.2366d+00	.07	1.1700d+00
.0008	1.2365d+00	.08	1.1610d+00
.0009	1.2364d+00	.09	1.1521d+00
.0010	1.2363d+00	.10	1.1435d+00
.0011	1.2362d+00	.11	1.1349d+00
.0012	1.2361d+00	.12	1.1266d+00
.0013	1.2360d+00	.13	1.1184d+00
.0014	1.2359d+00	.14	1.1105d+00
.0015	1.2358d+00	.15	1.1027d+00
.0016	1.2357d+00	.16	1.0950d+00
.0017	1.2356d+00	.17	1.0876d+00
.0018	1.2355d+00	.18	1.0804d+00
.0019	1.2354d+00	.19	1.0733d+00
.0020	1.2353d+00	.20	1.0665d+00

Example 3. Our final example illustrates Theorem 4. Here the normalization

$$(y(t))^T A'(t) y(t) = y(t)^T D y(t) = \psi(t) = 1$$

was applied to the eigenvector $y(t)$ corresponding to $\lambda_{\min}(A(t))$:

$$A(t)y(t) = \lambda_{\min}(A(t))y(t),$$

TABLE 3

t	$\|y(t)\|_2$	t	$\|y(t)\|_2$
.0001	1.2252d+00	.01	1.2351d+00
.0002	1.2253d+00	.02	1.2455d+00
.0003	1.2254d+00	.03	1.2565d+00
.0004	1.2255d+00	.04	1.2679d+00
.0005	1.2256d+00	.05	1.2799d+00
.0006	1.2257d+00	.06	1.2924d+00
.0007	1.2258d+00	.07	1.3055d+00
.0008	1.2259d+00	.08	1.3192d+00
.0009	1.2260d+00	.09	1.3334d+00
.0010	1.2261d+00	.10	1.3482d+00
.0011	1.2262d+00	.11	1.3636d+00
.0012	1.2263d+00	.12	1.3797d+00
.0013	1.2264d+00	.13	1.3963d+00
.0014	1.2265d+00	.14	1.4134d+00
.0015	1.2266d+00	.15	1.4312d+00
.0016	1.2267d+00	.16	1.4496d+00
.0017	1.2268d+00	.17	1.4685d+00
.0018	1.2269d+00	.18	1.4879d+00
.0019	1.2270d+00	.19	1.5079d+00
.0020	1.2271d+00	.20	1.5284d+00

where, once again, $A(t)$ is given in (3.1). Table 3 illustrates the increasingness in length of this eigenvector.

Acknowledgment. The authors are very grateful to Professor Dr. Ludwig Elsner for his constructive criticism on the original draft of this paper, and in particular on a previous version of Theorem 3.

REFERENCES

[1] C. DAVIS, *The rotation of eigenvectors by a perturbation*, J. Math. Anal. Appl., 6 (1963), pp. 159–173.
[2] ———, *The rotation of eigenvectors by a perturbation* II, J. Math. Anal. Appl., 11 (1965), pp. 20–27.
[3] C. DAVIS AND W. KAHAN, *The rotation of eigenvectors by a perturbation* III, SIAM J. Numer. Anal., 7 (1970), pp. 1–45.
[4] L. ELSNER, C. R. JOHNSON AND M. NEUMANN, *The effect of the perturbation of a nonnegative matrix on its Perron eigenvector*, Czech. Math. J., 32 (1982), pp. 99–109.
[5] B. N. PARLETT, *The Symmetric Eigenvalue Problem*, Prentice-Hall, Englewood Cliffs, NJ, 1980.
[6] J. H. WILKINSON, *The Algebraic Eigenvalue Problem*, Clarendon Press, Oxford, 1965.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 2, April 1985

004

INCREMENTAL PROCESSING APPLIED TO MUNKRES' ALGORITHM AND ITS APPLICATION IN STEINBERG'S PLACEMENT PROCEDURE*

H. W. CARTER†, M. A. BREUER‡ AND Z. A. SYED§

Abstract. In this paper we indicate how the concept of incremental processing can be applied to Steinberg's procedure for the placement of modules. In this procedure Munkres' algorithm is repeatedly used to solve linear assignment problems. We consider each assignment problem (matrix) to represent an incremental change with respect to the previous one, and present new techniques for solving a new assignment problem given the results of the previous one. We refer to this new algorithm as the incremental Steinberg algorithm. Experimental results indicate that this new algorithm produces results equally as good as the classical technique but at a substantial reduction in CPU time.

Key words. incremental processing, layout, Munkres' algorithm, PC cards, placement, Steinberg's algorithm

Introduction. The concept of incremental processing was introduced by Breuer [2] as a means of reducing computation time. The basic concept is that when executing an algorithm for similar pieces of data, the intermediate or final results from processing one set of data may be useful when processing the next set. In this paper we apply this concept to the classical problem of placing components either on a circuit board or in a gate array LSI circuit. The algorithm which we chose to study for solving this problem is due to Steinberg [6], which in turn employs Munkres' [5] assignment algorithm. It is this latter algorithm for which we have developed an incremental processing version.

1. Review of Munkres' and Steinberg's algorithms.

1.1. Munkres' algorithm.

The *linear assignment* (LA) problem deals with assigning n objects to n locations, where a_{ij} is the cost of assigning the ith object to the jth location, such that the total cost of the assignment is minimal. The matrix $A=[a_{ij}]$ is called the *assignment matrix*. The problem can be reduced to selecting n elements of A such that

1) the sum of these n elements is minimal, and

2) the selected elements are *independent*, i.e., only one element is selected from each row and column.

By subtracting appropriate constants from the rows and columns of A, it can be shown that the LA problem can be solved by selecting n independent zeros from the resulting matrix [4].

Munkres' algorithm is an efficient way of selecting the n independent zeros from the modified assignment matrix. The complexity of this algorithm is $O(n^3)$. We assume the reader is familiar with this algorithm.

1.2. Steinberg's algorithm.

The placement problem can be defined as follows [3]. Given a set of modules with signal sets defined over subsets of these modules, and a set of locations (slots) with distance d_{ij} defined over all pairs of locations, assign the modules to the locations so as to minimize some objective function. Let $C=[c_{ij}]$ be a

* Received by the editors October 16, 1979, and in final revised form July 24, 1981. This work was supported in part by the National Science Foundation under grants ENG 74-18647 and ECS-8005957.

† Department of Electrical Engineering, Air Force Institute of Technology, Wright-Patterson Air Force Base, Dayton, Ohio 45433.

‡ Department of Electrical Engineering and Systems, University of Southern California, Los Angeles, California 90089-0272.

§ P.E.C.H.S., Karachi, Pakistan.

cost or connection matrix, where c_{ij} is the weighted sum of the signals common to modules i and j. Then we will take as our objective function the expression $\sum_{j>i} c_{ij}d_{p(i)p(j)}$, where p is a permutation, $p(i)$ is the location of module i, and d_{xy} is the distance between locations x and y. Hence, for this choice of C, the placement problem can be formulated as a quadratic assignment problem. Steinberg's algorithm is a heuristic procedure for solving this problem. Steinberg observed that for most practical placement problems there are a large number of sets of modules which are unconnected. The placement cost of a module i in any set of unconnected modules is independent of where the remaining modules in this set are placed. This observation led to the idea that starting from an initial placement, the placement cost could be monotonically decreased if a sequence of better placements for sets of unconnected modules could be found. The placement of sets of unconnected modules is called a *subplacement problem.*

Due to the independence of the placement cost of modules in an unconnected set, each subplacement problem can be formulated as an LA problem.

In most implementations of Steinberg's algorithm, Munkres' algorithm is used. In this paper we will show that for this application, a modified version of Munkres' algorithm can be developed which is significantly more efficient than the classical procedure. We gain this efficiency because Steinberg's algorithm repeatedly executes the Munkres' algorithm, hence creating the environment for incremental processing. Our modified Munkres' method is called the Multiple Value Change (MVC) algorithm. Incorporating this modified version of Munkres' procedure into Steinberg's algorithm produces a new algorithm which we call the incremental Steinberg algorithm.

2.0. Linear assignment updating algorithms. In this section we will be concerned with solving the *linear assignment updating problem* which is defined as follows: Given an assignment matrix A and an optimal LA solution for A, find the optimal LA solution to A', where A' is obtained from A by modifying a few of the entries in A. (These are called incremental changes.) The conventional method of solution to this problem would be to take A' as a new assignment matrix and apply to it standard LA techniques, e.g. Munkres' algorithm. We will show that often a more efficient procedure exists. Two algorithms for the LA updating problem will be presented in this section. The complexity of these algorithms for small number of changes in A is $O(n^2)$.

The general form for our updating algorithm has inputs δ and $X = (S, A, T, U, V)$ and output $X' = (S', A', T', U', V')$. In Munkres' algorithm one starts with a matrix A, and via a process of row and column subtractions, produces a final matrix T from which a solution S is obtained. U and V represent vectors whose elements provide the information on how T was constructed from A, namely $t_{ij} = a_{ij} - u_i - v_j$. S consists of a set of n pairs of the form (i, j) which indicates that row i is assigned to column j. δ represents the change in A which produces A'. If $X = (S, A, T, U, V)$ and $X' = (S', A', T', U', V')$, then the LA updating algorithm calculates X' given X and δ.

Before describing our algorithm for the general case, we will first present results for the special case when only one element in A is changed. Though this case is not used in our final version of the incremental Steinberg algorithm, we present it here for several reasons, namely: 1) it represents the most elementary incremental type of change possible, and can be processed in a special way significantly faster than that required for the general case; 2) the proof for this case can be easily extended to cover the general case; and 3) it most clearly illustrates the elegance of incremental processing.

2.1. The Single Value Change (SVC) algorithm. The SVC algorithm updates X to produce X', where $\delta = (\alpha, p, q)$ represents a single change to the element a_{pq} of A by

an amount α. That is, $a'_{pq} = a_{pq} + \alpha$. The SVC algorithm can be executed repeatedly in order to process a sequence of changes.

The procedure first determines the type of change based upon the sign and magnitude of α, and whether or not (p, q) is in the solution S. If the type of change is a nontrivial update of matrix T to T', then one iteration of the main loop of Munkres' algorithm is performed. Otherwise the procedure exits immediately without performing any part of Munkres' algorithm. For example, if a_{pq} is decreased (increased) in value, and (p, q) was (was not) in the solution, then it remains in (out of) the solution. Figure 1 indicates the possible cases which may occur in the execution of the SVC algorithm.

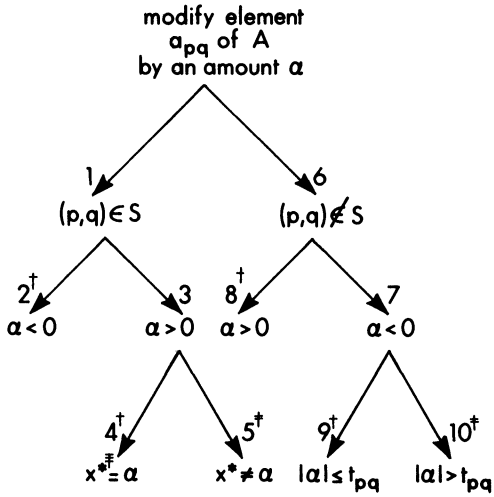


FIG. 1. *Conditions for, and types of modifications required to produce X' from X and δ using the* SVC *algorithm.* † *Solution S unaltered, algorithm exists after making a few changes in A, T and sometimes in U and V.* ‡ *Solution may be altered, algorithm executes one iteration of the main loop of Munkres' procedure.* ‡ *x^* is the minimal element in row p and/or column q of T after replacing t_{pq} by α.*

In the procedure a *covering line* in a matrix is an imaginary line indicating that at least one zero exists in the row or column defined by this line. As was done in [5], some zero elements will be distinguished by means of asterisks (stars) and primes.

The SVC *procedure.* Given an initial state vector $X = (S, A, T, U, V)$ and a change $\delta = (\alpha, p, q)$ to A, where $\alpha \neq 0$, the SVC updating algorithm produces the new state vector $X' = (S', A', T', U', V')$. We assume A is $n \times n$.

Step 1. Obtain A' and T' from A and T by replacing a_{pq} and t_{pq} by $a_{pq} + \alpha$ and $t_{pq} + \alpha$ respectively.

Step 2. There are three cases that have to be considered (refer to Fig. 1).

(i) nodes 8, 9: solution is unaltered and no further processing is needed.

(ii) nodes 2, 4: again the solution is unaltered. If t'_{pq} is the minimum element in row p, then subtract it from each element in row p and add it to u_p, otherwise subtract it from each element in column q and add it to v_q.

(iii) nodes 5, 10: the solution may be altered. Subtract from row p its minimal entry and add this entry to u_p; similarly subtract from column q its minimal entry and add this entry to v_q. Now cover all columns except q and apply Steps 3–5.

Step 3. Select a noncovered zero in T' and prime it. (There must be at least one the first time through this step.) Consider the row containing the selected zero. If there

is no 0* in this row go to Step 5, otherwise cover the row and uncover the column containing the 0* in this row. Repeat Step 3 until all zeros are covered.

Step 4. Let h be the smallest uncovered element in T'. Add h to each element in each covered row i and subtract h from the corresponding element u_i in U. Also subtract h from each element of each uncovered column j, and add h to the corresponding element v_j in V. Go to Step 3.

Step 5. Construct a sequence of alternating starred and primed zeros as follows. Let Z_0 denote the starting 0′ found in Step 3. (This 0′ has no 0* in its row.) Let Z_1 denote the 0* in Z_0's column (if any). Let Z_2 denote the 0′ in Z_1's row. Continue until the sequence stops at a 0′ which has no 0* in its column. Unstar each starred zero and star each primed zero in the sequence. Remove all primes. If n starred zeros are obtained then exit. (These represent the solution S'.) Otherwise go to Step 3.

Note that Steps 3, 4 and 5 are derived from Munkres' procedure.

2.1.1. Establishing the validity of the SVC algorithm. Two assignment matrices A and T are considered to be *equivalent* if an optimal assignment for one is also an optimal assignment for the other. The validity and optimality of the SVC algorithm rests on the following observations.

Given A, T, U, V where $t_{ij} = a_{ij} - u_i - v_j$, then T and A are equivalent [1]. Clearly for the following cases, $S' = S$:

(1) $(p, q) \in S$, $\alpha < 0$,

(2) $(p, q) \notin S$, $\alpha > 0$,

(3) $(p, q) \in S$, $\alpha > 0$ and $x^* = \alpha$, where x^* is defined in Fig. 1 of the SVC algorithm,

(4) $(p, q) \notin S$, $\alpha < 0$ and $t_{pq} > |\alpha|$.

Finally note that the execution of Step 2 of the SVC algorithm preserves equivalence of T and T'. Hence when Munkres' algorithm is entered it converges to an optimal solution to A'.

2.1.2. Complexity analysis for the SVC algorithm. The speed advantage to the SVC algorithm is given by the following results.

THEOREM 1. *n 0*'s are obtained the first time Step 5 is executed.*

Proof. This result follows from the fact that when Step 5 is entered $(n-1)$ 0* elements already exist in T', hence only one new 0* element is required. Since each iteration of Step 5 adds at least one new 0* element, the theorem follows. □

Note that the locations of the initial $(n-1)$ 0*'s do not necessarily correspond to the locations of the final 0*'s.

Consider unit operations to be the following: scan a row or column; cover or uncover a row or column; star, prime, unstar, or unprime a zero; add to or subtract from a row or column.

The worst case path for the SVC algorithm requires $(n-1)$ iterations of Steps 3 and 4, and the total number of operations is $4n^2+4n-5$ and hence this procedure is $O(n^2)$. When Steps 3, 4 and 5 are not required, then the number of operations varies from constant time to $O(n)$.

2.2. The Multiple Value Change (MVC) algorithm. The MVC algorithm updates an optimal LA solution when r rows and c columns of the original assignment matrix A are modified producing the matrix A'. In fact, the dimensions of A and A' need not be the same.

The MVC *procedure.* Let A' be obtained from A by changing rows $i_1, i_2, \cdots, i_r$ and columns $j_1, j_2, \cdots, j_c$. Let $\Delta = A' - A$; then Δ may have nonzero entries only in these rows and columns.

Form the matrix $T'' = T + \Delta$; note that some entries of T'' may be negative. For each $(p, q) \in S$ for which $p \neq i_1, i_2, \cdots, i_r$ and $q \neq j_1, j_2, \cdots, j_c$, star the zero at position (p, q) in T''. Now subtract from each row of T'' that does not contain a 0* its minimal entry (which may be negative) and add it to the corresponding element of U. Now subtract from each column that does not contain a 0* its minimal entry and add it to the corresponding element of V. The result is T'. Now cover each column containing a 0* and then apply Steps 3-5 of the SVC algorithm.

Note that if A' is obtained by changing k rows and/or columns of A, where $k = r + c$, then one begins the Munkres' algorithm with at least $(n - k)$ 0* entries.

Complexity analysis of the MVC *algorithm.* The total number of operations for the MVC algorithm is $T(n, k) = t(n, k) + n + 3k$, where $t(n, k)$ is the number of operations required for Steps 3-5 of Munkres' algorithm, and $k = r + c$ is the number of row and column changes. From Munkres [5], the maximum number of operations required to obtain $m + 1$ zeros, where m independent zeros have already been determined, is given by the expression $m^2 + 10m + 3nm - n + 4$. If this expression is summed from $(n - k)$ to $(n - 1)$, then we will have the expression for $t(n, k)$; hence

$$T(n, k) = n + 3k + \sum_{m=n-k}^{n-1} (m^2 + 10m + 3nm - n + 4).$$

In Fig. 2 we indicate the value of $T(n, k)$ for $0.1 \leqq k/n \leqq 1.0$, $n = 10$ and 100, where the value of T has been normalized by dividing by $T_{\max} = T(n, n)$. ($T(n, n)$ is essentially the time required to solve an $n \times n$ assignment problem.) We see that there is very little variation in $T(n, k)/T_{\max}$ as a function of n, while for $k < 0.5$ the growth in $T(n, k)$ as a function of k is almost linear.

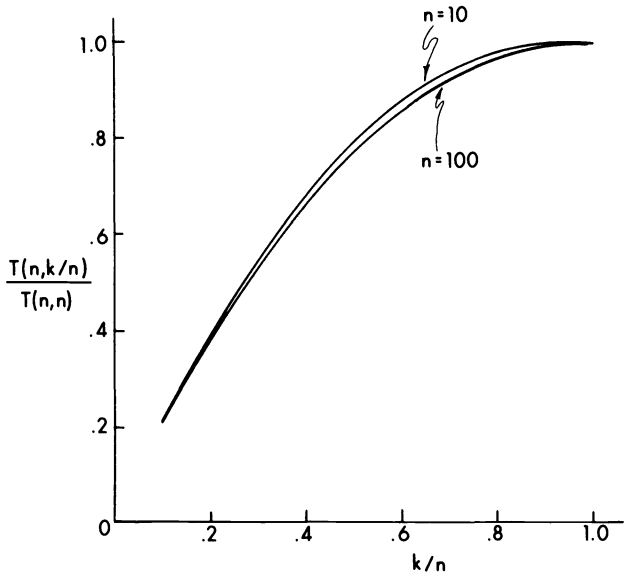


FIG. 2. *Number of operations (normalized) vs. number of rows and columns changed expressed as percent of total number of rows (or columns) in the matrix.*

As k approaches n, $T(n, k)$ is of the order $O(n^3)$, and there is no advantage in using the MVC algorithm. We normally do not select to use the MVC algorithm to replace Munkres' algorithm unless $0 < k \leqq K < n$. We have not yet found the optimal value for K for deciding which of the two algorithms to select.

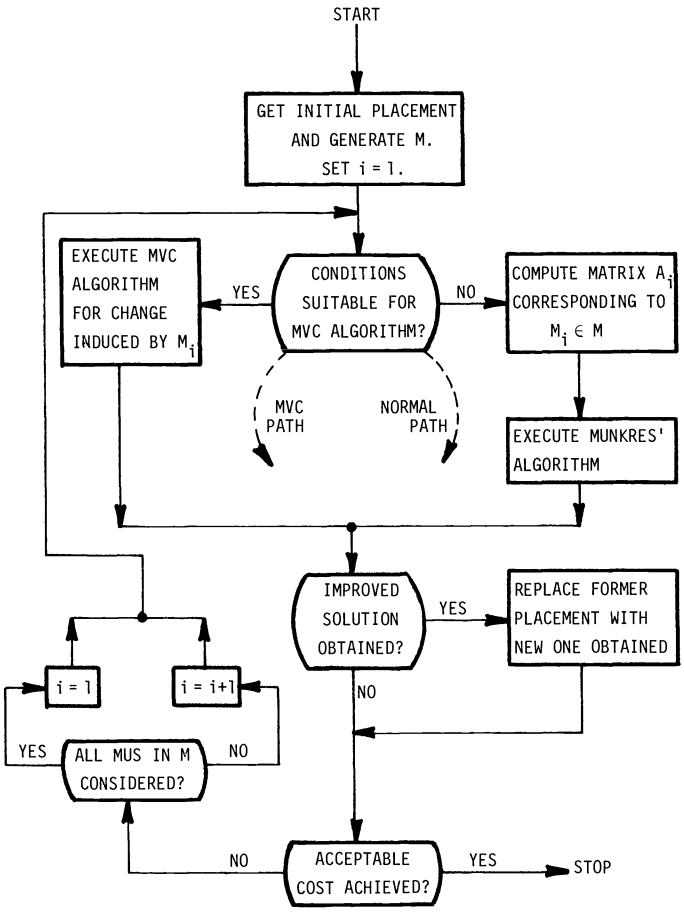


FIG. 3. *Flowchart of the incremental Steinberg algorithm.*

3. The incremental Steinberg algorithm. A flow chart of the modified version of the Steinberg algorithm using the MVC algorithm is shown in Fig. 3. Here M is a subset of the set M^* of all maximal unconnected sets (MUS). There are two main conditions which enhance the suitability of the MVC algorithm over Munkres' algorithm, namely:

(1) The assignment matrices for consecutive iterations should have a large number of identical rows and columns;

(2) The location of the new solution elements (row and column assignments) for A' should be similar to those for A.

Condition 1 can be met in part by careful selection and ordering of the maximal unconnected sets. Condition 2 cannot be directly dealt with, but hopefully it is partially satisfied by Condition 1.

Only those parts of the modified Steinberg algorithm which are different from the classical Steinberg method will now be described.

As in the classical method, the set M^* of all maximal unconnected sets is generated. For most practical problems the number of sets in this set is very large so a subset $M \subseteq M^*$ is selected and used. As is normally done, the subset is selected using the following selection criteria:

(1) sets containing large number of modules;

(2) sets such that the union of all selected sets equals the module set.

In addition, sets are selected so that we maximize the number of elements in common between consecutive sets. This latter condition is used to satisfy Condition 1 stated previously. We satisfy these conditions by first selecting sets which have a large number of modules in common, and then *ordering* these sets such that the number of common modules between consecutive sets is maximized. It will be shown later that this enhancement procedure does indeed reduce CPU time, but unfortunately, in some cases results in a slightly larger final assignment cost. For the sake of brevity, we have not included the heuristic procedure used to order the sets in M.

Example. Consider a network which consists of three signal sets:

$$S_1 = \{A, B, C, E, H\}, \quad S_2 = \{B, D, E, G\}, \quad S_3 = \{B, D, F\},$$

and the eight modules $A, B, \cdots, H$ which are to be placed on a 3×3 slotted board. The ordered maximal unconnected sets are: $M_1 = \{E, F\}$, $M_2 = \{A, F, G\}$, $M_3 = \{F, G, H\}$, $M_4 = \{C, F, G\}$, $M_5 = \{C, D\}$, $M_6 = \{A, D\}$, $M_7 = \{D, H\}$. (Some MUSs can be eliminated if desired.) Because the board has one empty slot, we add the pseudo element Z to all MUSs.

The initial placement is shown in Fig. 4 where nets are connected via chains and rectilinear distances are used.

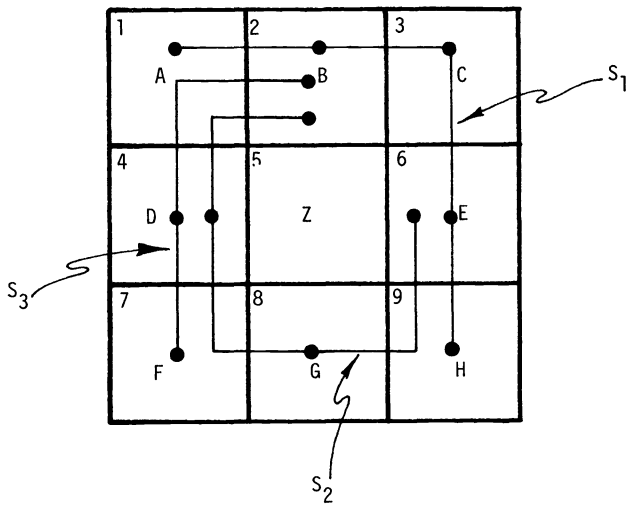


FIG. 4. *Initial placement.*

For $i = 1$, $M_i = \{E, F, Z\}$ and we have

$$A_1 = \begin{matrix} \\ E \\ F \\ Z \end{matrix} \begin{matrix} \begin{matrix} 6 & 7 & 5 \end{matrix} \\ \begin{bmatrix} 10 & 10 & 11 \\ 4 & 3 & 3 \\ 0 & 0 & 0 \end{bmatrix} \end{matrix}.$$

4. Experimental results. The algorithms described in this paper have been implemented in PASCAL and executed on a PDP/10. In this section we will present some of the results obtained. Our main questions to be answered are: how does the incremental Steinberg algorithm compare in both final assignment cost and CPU time to the classical Steinberg algorithm?

The final assignment cost and CPU time required to achieve a solution is a function of the *path* taken by the algorithm. The solution path depends on such factors as the initial placement, the selected MUS, the ordering of the selected MUS, and the optimal solution (more than one may exist) selected for each application of the LA process.

In order to make an accurate timing comparison, both our new and the classical methods should follow the same path. Hence we used the same initial conditions, selection, ordering and placement updating algorithms for both cases. In most cases these conditions forced the two algorithms to follow the same path, but in some cases, where a choice of optimal LA solutions exist at some stage, different solutions were selected and the paths did diverge.

Four boards were evaluated. Their properties are summarized in Table 1.

TABLE 1
Board characteristics.

BOARD NO.	TYPE	NO.OF MODULES	NO.OF SLOTS	NO.OF SIGNAL NETS
1	STEINBERG	34	36	——
2	REAL*	31	32	50
3	REAL*	30	32	51
4	REAL*	24	32	52

*THESE ARE ACTUAL BOARDS.

Experiment 1. This experiment uses the following selection and ordering algorithms for generating M.

ALGORITHM 1. *Generation of M.*

Step 1. $i = 1$.

Step 2. If $i >$ number of modules, then exit.

Step 3. Find the largest MUS which contains the module i.

(a) If this set is already selected then set $i = i + 1$ and go to Step 2.

(b) If there is no tie then select this set.

(c) If there is a tie then select that set which has the largest number of common entries with the already selected sets. If a tie still exists then use the "lower index rule." Set $i = i + 1$ and go to Step 2.

ALGORITHM 2. *Ordering elements in M.*

Step 1. Starting with each set in M, find the best ordering in terms of maximizing the intersection between consecutive MUSs. (Details of how this is done are not given here.)

Step 2. Select the best of all the orderings generated in Step 1.

Finally the updating criterion used is to update the placement each time a placement having a reduced cost is found.

The results from this first experiment are shown in Table 2. The results indicate an average reduction in CPU time of almost 50%, with almost no change in the value of the cost function. Note, however, that these results are biased in that the elements in M are both selected and ordered to benefit the incremental Steinberg algorithm. We rectify this situation in the next experiment.

TABLE 2
Results from Experiment 1.

BOARD NO.	NO. OF MUS IN M	PREPROCESSING TIME[1] (% OF TOTAL)		COST			TOTAL CPU TIME (SEC.)		
		INCREMENTAL ALGORITHM	CLASSICAL ALGORITHM	INCREMENTAL ALGORITHM	CLASSICAL ALGORITHM	"%"*	INCREMENTAL ALGORITHM	CLASSICAL ALGORITHM	"%"*
1	23	59	28	6131	6131	0	44	88	50
2	29	89	66	160	161	0.62	62	81	23
3	18	41	17	2586	2586	0	19	43	55
4	16	18	3	1020	1020	0	7	30	77
					AVERAGE	~0			50

[1] INCLUDES GENERATING M*, M AND ORDERING M

*"%" MEANS % IMPROVEMENT

Experiment 2. For this experiment the following changes were made, namely:

1) a new selection rule for generating M is used; and

2) *no* ordering of the sets in M is used for the classical algorithm, but the elements are still ordered for the incremental version. Note that the same path will no longer be followed. The new selection rule is given next.

ALGORITHM 3. *Generation of M.*

Only Step 3(c) of this algorithm is presented since the rest of the algorithm is the same as Algorithm 1.

Step 3(c). If there is a tie, then select that set which has the smallest number of common modules with the already selected set of MUSs. If a tie still exists, then use the "lower index rule." Let $i = i + 1$ and go to Step 2.

Though this rule is the opposite of that found in Algorithm 1 and gives a worst case condition for applying the MVC algorithm, we still get very good results (see Table 3).

TABLE 3
Results from Experiment 2.

BOARD NO.	NO. OF MUS IN M	PREPROCESSING TIME[1] (% OF TOTAL)		COST			TOTAL CPU TIME (SEC.)		
		INCREMENTAL ALGORITHM	CLASSICAL ALGORITHM	INCREMENTAL ALGORITHM	CLASSICAL ALGORITHM	"%"*	INCREMENTAL ALGORITHM	CLASSICAL ALGORITHM	"%"*
1	21	55	27	6234	5575	-11.8	45	91	50.7
2	17	87	65	139	149	6.7	61	81	24.6
3	17	52	35	2772	2934	7.1	14	21	30.6
4	12	12	4	1039	1039	0.0	6	17	64.9
					AVERAGE	0.5			42.7

[1] INCLUDES GENERATING M*, M AND ORDERING M

*"%" MEANS % IMPROVEMENT

Our incremental Steinberg algorithm was executed using Algorithm 3 for generating M with and without ordering. The results indicated a 2% average increase in cost and a 14% average decrease in run time when ordering is used. The reason for the reduction in run time is due to the fact that ordering makes the MVC algorithm more efficient. However, it appears that by making consecutive MUSs similar, we may be increasing the chances that the placement gets into a local optimal condition and hence leads to suboptimal results. However, these results are not conclusive since for two cases there was actually a small decrease in the final cost.

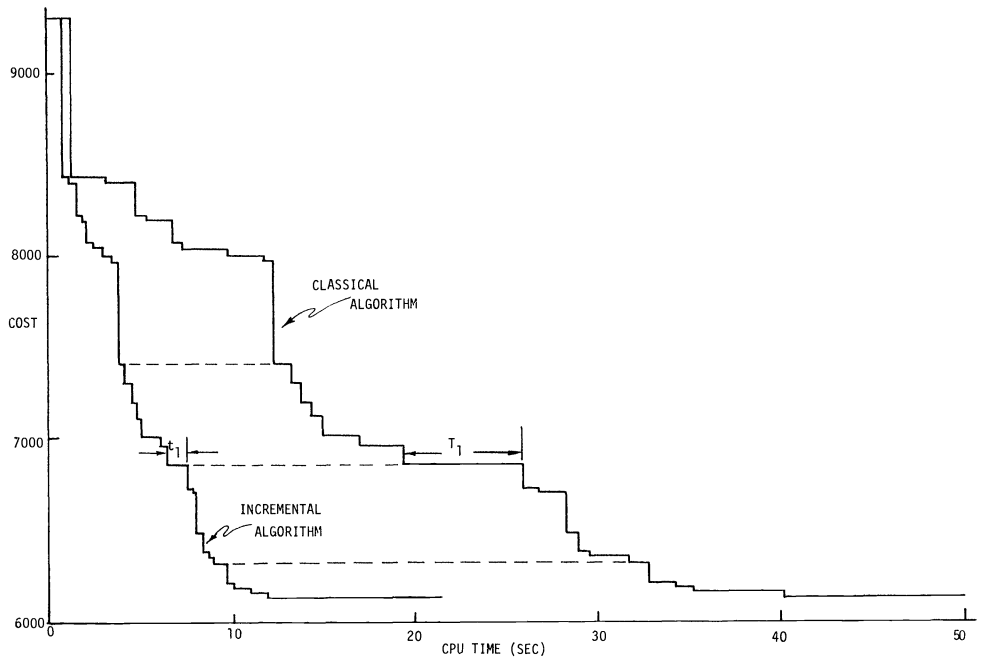


FIG. 5. *Cost vs.* CPU *time for both algorithms.* (*No preprocessing time indicated.*)

Finally, in Fig. 5 we indicate a plot of the current solution cost vs. CPU time for both the classical and incremental Steinberg algorithms. Board No. 1 is used along with Algorithm 1. Here we see that the path taken by the two algorithms is identical; each cost for the incremental algorithm occurs also for the classical algorithm, except that for the latter it is shifted to the right.

Conclusion. In this paper we have presented a new version of Steinberg's placement algorithm which employs the concept of incremental processing. This new algorithm deals primarily with reducing the time required to solve the linear assignment problem.

Our experimental results indicate that this new algorithm leads to substantial savings in computation time. We have also indicated that the method used in selecting the MUSs to be used as well as their order can influence the performance of the algorithm.

Acknowledgment. The authors would like to thank the first reviewer, who showed us how to simplify the presentation of our algorithms.

REFERENCES

[1] F. BOURGEOIS AND J. LASSALLE, *An extension of the Munkres' algorithm for the assignment problem to rectangular matrices*, Comm. ACM, 14 (1971), pp. 802-804.

[2] M. A. BREUER, *Incremental processing in design automation*, SIGDA Newsletter, 6 (1976), pp. 2-9.

[3] M. HANAN, P. K. WOLFF, SR. AND B. J. AGULE, *A study of placement techniques*, J. Design Automation and Fault Tolerant Computing, 1 (1976), pp. 28-61.

[4] H. W. KUHN, *The Hungarian method for the assignment problem*, Naval Res. Logistics Quart., 2 (1955), pp. 83-97.

[5] J. MUNKRES, *Algorithms for the assignment and transportation problems*, J. Soc. Ind. Appl. Math., 5 (1957), pp. 32-38.

[6] L. STEINBERG, *The backboard wiring problem: a placement algorithm*, SIAM Rev., 3 (1961), pp. 37-50.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 2, April 1985

NUMERICAL SOLUTION OF NAVIER–STOKES PROBLEMS BY THE DUAL VARIABLE METHOD*

CHARLES A. HALL†

Abstract. Computational fluid dynamics as a research area has attracted mathematicians not only because of its importance to the engineering community, but also because of the pitfalls that are encountered in solving various discretizations of the governing Navier-Stokes equations. Such pitfalls are highlighted in this paper along with methods to circumvent them.

Discretizations of the Navier-Stokes equations often can be viewed as systems defining flows on an associated network. This observation provides a means of economizing on their numerical solution.

Key words. Navier-Stokes, convection-diffusion, networks, stability, dual variable

AMS(MOS) subject classifications. 05C38, 65M10, 76D05

1. Introduction. For centuries man has been intrigued by the mysteries of fluid flow. Early in the eighteenth century, Daniel Bernoulli initiated the science of hydrodynamics which deals with the motion of fluids, d'Alembert introduced the principle of conservation of mass in a liquid, and a mathematical theory of fluid flow emerged under the guidance of Euler and Lagrange. It was Navier (1785–1836) and later Stokes (1819–1903) who derived the basic nonlinear linear differential equations describing the motion of a viscous fluid. These equations are central to all modern investigations of fluid dynamics. From these early beginnings there has been a slow and arduous development of rather complex theories to explain the phenomena of fluid flow. To be sure, this development, even though spurred on by our 20th century interests in aerodynamics, is far from complete.

Although analytical methods were first used to solve specialized fluid flow problems, the advent of high speed digital computers coupled with robust numerical algorithms has made it possible today to solve complex, large-scale fluid flow problems arising in diverse engineering applications. Specific applications include:

- flow of air around airplanes and automobiles,
- flow of water through soil,
- flow of steam-water mixtures in heat exchangers,
- flow of fuel-air mixtures in combustion engines,
- flow of oil through porous rock,
- flow of tidewater in lakes and estuaries,
- flow of lubricants around bearings,

to name but a few.

Computational fluid dynamics is a research area which has attracted many mathematicians, especially numerical analysts, not only because of its importance to the engineering community, but also because of the many pitfalls that are encountered in solving various discretizations of the governing equations. The nonlinearities involved in such convection-diffusion models give rise to nontrivial questions of the existence and uniqueness of a solution. Spatial as well as temporal instabilities may occur.

* Received by the editors July 26, 1983, and in revised form December 10, 1983. This research was supported by the Air Force Office of Scientific Research under grant AFOSR-80-0176 and was presented as an invited talk at the 62nd Summer Meeting of MAA, August 23–26, 1982, Toronto, Canada.

† Institute for Computational Mathematics and Applications, Department of Mathematics and Statistics, University of Pittsburgh, Pittsburgh, Pennsylvania 15261.

In addition, the equations involved are typically not well structured and the task of constructing efficient, robust solution algorithms is formidable. Many two-dimensional and certainly all practical three-dimensional analyses of fluid flow problems tax current computer capabilities at all but a very few installations.

This paper deals primarily with the *dual variable method* [1], [8] which uses network theory to construct matrix transformations of the discrete Navier-Stokes equations that nominally reduce by a factor of 27 the computational cost in solving two-dimensional transient fluid flow problems. This method is atypical since the very construction of the algorithm as well as its analysis embraces rather nontrivial mathematical concepts. The dual variable method has been successfully applied in large scale production computer codes which have been used to model for example two phase flow through steam generators of nuclear power plants, flow of combustion gases through automotive catalytic convertors and flow of binary gas mixtures in gas turbine engines.

2. The Navier–Stokes equations. The mathematical equations governing the flow of a viscous incompressible fluid are the well-known [14], [17] Navier-Stokes equations, which in dimensionless form can be written:

$$\begin{aligned}\frac{\partial u}{\partial t}+u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}&=-\frac{\partial p}{\partial x}+\frac{1}{R}\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}\right)+F_1,\\ \frac{\partial v}{\partial t}+u\frac{\partial v}{\partial x}+v\frac{\partial v}{\partial y}&=-\frac{\partial p}{\partial y}+\frac{1}{R}\left(\frac{\partial^2 v}{\partial x^2}+\frac{\partial^2 v}{\partial y^2}\right)+F_2,\end{aligned} \tag{1}$$

where (u, v) is the dimensionless velocity vector, p is dimensionless pressure, R is the Reynolds number and F_i are source terms. The Reynolds number is defined to be $R \equiv u_0 d\rho/\mu$ where μ is the fluid viscosity, ρ is the fluid density, d is a characteristic length and u_0 is a characteristic velocity. Generally speaking, the higher the Reynolds number, the more difficult it is to compute solutions to (1). These *nonlinear* convection-diffusion partial differential equations can be derived from first principles as representing a conservation of momentum.

Similarly, the physical law of the conservation of mass gives rise to the constraint that

$$\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}=0. \tag{2}$$

Mathematically, our problem is to find u, v, p satisfying (1)-(2) for (x, y) in some flow region $\Omega \subset R^2$, and for time $t \geqq 0$, subject to a combination of flow specified, pressure specified and line of symmetry (free slip) boundary conditions. Since (1) is time dependent we must also specifiy an initial state $u(x, y, 0)$, $v(x, y, 0)$ and $p(x, y, 0)$.

Equations (1)-(2) represent an extremely simplified model, but are sufficient for our main purpose of illustrating the dual variable reduction technique. More complex models, such as those solved numerically to obtain results to be shown later, nominally require that:

- A thermal energy or temperature transport equation be added to system (1)-(2) which predicts the enthalpy h or the temperature T of the fluid.
- A state equation be added in which the density ρ depends locally on the fluid temperature and/or fluid pressure.
- Resistance terms (possibly nonlinear) and gravitational terms be added as sources F_1 and F_2.

For example, the system investigated in [5] is of the form

$$
\begin{aligned}
&\frac{\partial(\rho u)}{\partial t}+\frac{\partial}{\partial x}(\rho u^2)+\frac{\partial}{\partial y}(\rho uv)=-\frac{\partial p}{\partial x}+\mu\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}\right)+(a|u|+b)u,\\
&\frac{\partial(\rho v)}{\partial t}+\frac{\partial}{\partial x}(\rho uv)+\frac{\partial}{\partial y}(\rho v^2)=-\frac{\partial p}{\partial y}+\mu\left(\frac{\partial^2 v}{\partial x^2}+\frac{\partial^2 v}{\partial y^2}\right)+(c|v|+d)v,\\
&\frac{\partial \rho}{\partial t}+\frac{\partial(\rho u)}{\partial x}+\frac{\partial(\rho v)}{\partial y}=0, \qquad (3)\\
&\frac{\partial(\rho h)}{\partial t}+\frac{\partial}{\partial x}(\rho uh)+\frac{\partial}{\partial y}(\rho vh)=F_3,\\
&\rho=\rho(h).
\end{aligned}
$$

Density now varies with enthalpy (so-called *thermally-expandable* flow) and hence with position. As such it can not be taken outside the differentiation. For steam-water mixtures [5], [9] the density may very radically in adjoining flow cells along a steam-water interface making (3) a much more formidable numerical problem than the incompressible problem (1)-(2).

3. Discretizations of Navier–Stokes equations. Early analytical approaches for solving the Navier-Stokes equations gave way to numerical methods early in the 20th century. Currently, there is a scholarly controversy over which means of discretizing the Navier-Stokes equations is the "best" approach; finite difference or finite element, and within each of these methods, heated discussion continues as to various choices between difference operators and element types respectively [4], [7], [15]. Our purpose here is in no way to resolve these issues. However, we will illustrate in the next section that for most reasonable choices, the dual variable method provides a means of economizing on the cost of solving whatever discretization is chosen.

First though, we illustrate two of the computational pitfalls encountered in the discretization of convection-diffusion equations such as (1), and the ways in which one can circumvent these problems. For ease of exposition, consider the one-dimensional boundary value problem:

$$
\begin{aligned}
&au_x-\frac{1}{R}u_{xx}=0, \qquad 0<x<1,\\
&u(0)=0, \qquad u(1)=1,
\end{aligned} \qquad (4)
$$

where a and R are positive constants. One can verify that the solution to (4) is

$$u(x)=(1-e^{aRx})/(1-e^{aR}). \qquad (5)$$

Suppose we seek a finite difference solution to (4) in a uniform grid $0=x_0<x_1<\cdots<x_N=1$ of gauge $\Delta x=1/N$ and use the "standard" central difference operators to obtain the second order consistent finite difference equations

$$a\left(\frac{u_{i+1}-u_{i-1}}{2\Delta x}\right)-\frac{1}{R}\left(\frac{u_{i-1}-2u_i+u_{i+1}}{\Delta x^2}\right)=0, \qquad 1\leqq i\leqq N-1 \qquad (6)$$

where $u_i \doteq u(x_i)$. This set of difference equations has the closed form solution

$$u_i=(1-Z^i)/(1-Z^N), \qquad 0\leqq i\leqq N \qquad (7)$$

where $Z=(2+aR\Delta x)/(2-aR\Delta x)$. Note that Z is the $(1,1)$-Padé second order approximant to $e^{aR\Delta x}$.

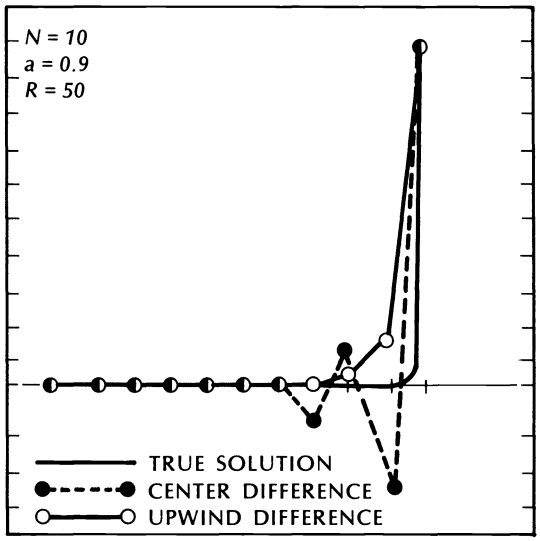


FIG. 1. *One-dimensional convection diffusion.*

But, unfortunately the centered difference approximation (7) oscillates unless $\Delta x < 2/aR$. For example, if $R = 50$ and $a = 0.9$ then for $N \leqq 22$ the solution oscillates in space (cf. Fig. 1). Such oscillations, or bounded spatial instabilities also occur in finite element solutions [4], [7], [12]. In fact, (6) is precisely the system of equations obtained from the finite element Galerkin method based on linear elements.

An accepted means of avoiding the aforementioned instability is through the use of *upwind differencing* [14], in which (4) is discretized as

$$a\left(\frac{u_i^E - u_i^W}{\Delta x}\right) - \frac{1}{R}\left(\frac{u_{i-1} - 2u_i + u_{i+1}}{\Delta x^2}\right) = 0, \qquad 1 \leqq i \leqq N-1, \tag{8}$$

where

$$u_i^E = \begin{cases} u_i & \text{if } a > 0, \\ u_{i+1} & \text{if } a < 0, \end{cases} \qquad u_i^W = \begin{cases} u_{i-1} & \text{if } a > 0, \\ u_i & \text{if } a < 0. \end{cases}$$

The derivative u_x is thus approximated by the "gradient" of the fluid entering node i from "upwind". In our case ($a > 0$), the system (8) has solution

$$u_i = (1 - Y^i)/(1 - Y^N), \qquad 0 \leqq i \leqq N \tag{9}$$

where $Y = (1 + aR\Delta x)$ is the (1, 0)-Padé first order approximant to $e^{aR\Delta x}$. The upwind difference solution (9) has the highly desirable property that it does *not* oscillate for any value of Δx (cf. Fig. 1). Note that if $a < 0$, then

$$u_i = (1 + Y^i)/(1 + Y^N), \qquad 0 \leqq i \leqq N \tag{10}$$

where $Y = (1 - aR\Delta x)^{-1}$ is the (0, 1)-Padé approximation to $e^{aR\Delta x}$. Again the solution does not oscillate.

Upwind differencing has also been used successfully in conjunction with finite elements by Christie et al. [4], Heinrich et al. [12] and Heinrich and Zienkiewicz [13].

Our purpose here is *not* to denigrate the second order consistent, oscillating scheme in (6), but only to emphasize that spatial instabilities or oscillations do enter into many such discretizations of Navier-Stokes equations and these must be dealt

with accordingly. If one cannot afford to choose Δx small enough so as to stabilize (6), then the first order consistent *nonoscillating* scheme (9) has much to offer, and in fact has been used with great success by the author and his colleagues [5], [8], [9] for two-dimensional Navier-Stokes problems. However, I want to point out that there are other investigators who advocate not "suppressing the wiggles"; see for example Gresho and Lee [7].

Next, let us recall that there is also a temporal instability when one discretizes a time dependent problem such as the Navier-Stokes problem (1). Again for simplicity consider, the one-dimensional problem ($a>0$)

$$u_t + au_x - \frac{1}{R}u_{xx} = 0, \qquad 0<x<1, \quad t>0 \tag{11}$$

subject to initial conditions

$$u(x,0) = \begin{cases} \varepsilon > 0, & x = .5, \\ 0, & x \neq .5 \end{cases} \tag{12}$$

and boundary conditions $u(0,t) = u(1,t) = 0$ for $t \geqq 0$. The *explicit* (forward time difference) discretization of (11) is

$$\left(\frac{u_i^{m+1} - u_i^m}{\Delta t}\right) + a\left(\frac{u_i^m - u_{i-1}^m}{\Delta x}\right) - \frac{1}{R}\left(\frac{u_{i-1}^m - 2u_i^m + u_{i+1}^m}{\Delta x^2}\right) = 0, \qquad 1 \leqq i \leqq N-1 \tag{13}$$

where Δt is the time step and $u_i^m \doteq u(i\Delta x, m\Delta t)$. By standard arguments we now show that this finite difference scheme is *stable* if and only if

$$\Delta t < \frac{R\Delta x^2}{2 + aR\Delta x}. \tag{14}$$

This is called the *von Neumann stability criterian* [6]. If (14) is satisfied then from (13) we have

$$u_i^{m+1} = \left[\left(\frac{1 + aR\Delta x}{R\Delta x^2}\right)\Delta t\right] u_{i-1}^m + \left[1 - \left(\frac{2 + aR\Delta x}{R\Delta x^2}\right)\Delta t\right] u_i^m + \left[\frac{\Delta t}{R\Delta x^2}\right] u_{i+1}^m \tag{15}$$

as a convex combination of values of u at the mth time step. As such u_i^{m+1} is bounded above by the maximum and below by the minimum values of u at the previous time step and the boundary data. Replacing m by $m-1$, etc. we deduce that u_i^{m+1} is bounded above and below by the boundary and initial data.

But if (14) is *not* satisfied, then by an argument similar to [6, p. 93] one can show that for some j

$$|u_j^m| \geqq (2m+1)^{-1}\left[2\left(\frac{2 + aR\Delta x}{R\Delta x^2}\right)\Delta t - 1\right]^m \varepsilon. \tag{16}$$

The right-hand side grows exponentially as $\Delta t \to 0$, and hence the difference scheme (13) is *unstable* if (14) is violated.

For large R, (14) requires $\Delta t < \Delta x/a$, but often a will be quite large forcing Δt to be very small.

If such small time steps are computationally prohibitive (and for many applications they are) then one is led naturally to replacing (13) by an implicit (backward time

differencing) discretization of (11) as

$$(17)\quad \left(\frac{u_i^{m+1}-u_i^m}{\Delta t}\right)+a\left(\frac{u_i^{m+1}-u_{i-1}^{m+1}}{\Delta x}\right)-\frac{1}{R}\left(\frac{u_{i-1}^{m+1}-2u_i^{m+1}+u_{i+1}^{m+1}}{\Delta x^2}\right)=0,\qquad 1\leqq i\leqq N-1.$$

That this discretization is stable for *all* choices of Δt also follows from standard arguments [6, p. 102]. To wit, (17) yields

$$(18)\quad u_i^{m+1}=u_i^m-\Delta t\left[\left(\frac{a}{\Delta x}+\frac{2}{R\Delta x^2}\right)u_i^{m+1}-\left(\frac{a}{\Delta x}+\frac{1}{R\Delta x^2}\right)u_{i-1}^{m+1}-\left(\frac{1}{R\Delta x^2}\right)u_{i+1}^{m+1}\right].$$

Now suppose j is such that $u_j^{m+1}\geqq u_k^{m+1}$ for all k. Then with $i=j$ in (18) we have, since the quantity in brackets is nonnegative, that

$$(19)\qquad u_j^{m+1}\leqq u_j^m.$$

Again, replacing m by $m-1$, etc., we deduce that the finite difference approximants u_j^{m+1} are bounded by the boundary and initial data.

The point of the above discussion is that, for Navier-Stokes type equations, *the choice of implicit (backward) time differencing and upwind spatial differencing leads to a very robust numerical algorithm.*

For the two-dimensional Navier-Stokes problem, (1)-(2), we follow this course and use the discretization that was used successfully in [9], [10]. The flow region Ω is subdivided into a union of rectangular mesh boxes or control volumes whose sides are parallel to the coordinate axes. A MAC placement of variables [11] is used in which a pressure is associated with the center of a control volume and the component of velocity normal to a control volume side is associated with the center of that side (cf. Fig. 2).

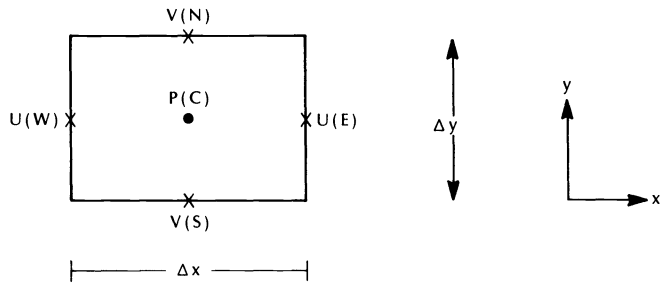


FIG. 2. *A control volume with* MAC *placement of variables and compass designations* (MAC ≡ Marker And Cell).

These flows are all oriented in the positive coordinate direction. The centered difference approximation, at the control volume center, to (2) is

$$(20)\qquad \frac{[U(E)-U(W)]}{\Delta x}+\frac{[V(N)-V(S)]}{\Delta y}=0.$$

After multiplying by $\Delta x\,\Delta y$, (20) simply states that the net flow across the boundary of the control volume is zero; conservation of mass.

The first (second) component of the Navier-Stokes equation (1) is approximated at the center of the vertical (horizontal) sides of each control volume. Let us assume that the time derivatives are approximated by implicit first order backward differences, the convection terms (e.g., $u\partial u/\partial x$) are approximated using upwind differences and that the diffusion terms (e.g., $\partial^2 u/\partial x^2$) are approximated using centered differences.

Further, these equations are linearized by lagging the convective coefficients one time step ($u^{m+1}(\partial u/\partial x)^{m+1}$ is approximated by $u^m(\partial u/\partial x)^{m+1}$, etc.). The details of this discretization are given in [1], [8] and are not included here. It suffices to state that the linearized finite difference approximation to (1)-(2) obtained is of the form

$$\begin{bmatrix} Q_m & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} \mathbf{V}^{m+1} \\ \mathbf{P}^{m+1} \end{bmatrix} = \begin{bmatrix} \mathbf{K}_m \\ \mathbf{S}_m \end{bmatrix}. \tag{21}$$

Suppose there are N control volumes with L sides across which a normal component of velocity is to be determined (note that the boundary conditions determine some components of velocity and some pressures). Then, $\mathbf{V}^{m+1}$ is the $L\times 1$ vector of velocity component approximations at time step $(m+1)$ and $\mathbf{P}^{m+1}$ is the $N\times 1$ vector of pressure approximations at time step $(m+1)$. A is the $N\times L$ discrete divergence operator approximating the divergence in (2), A^T is the $L\times N$ discrete gradient operator approximating the gradient in the term grad p in (1), Q_m is the $L\times L$ discretization of the convective and diffusion terms of (1) and $\mathbf{K}_m, \mathbf{S}_m$ are source terms containing known information such as boundary flows.

It must be emphasized that most other finite difference *or* finite element discretizations of (1)-(2) result in systems of equations of the same generic form as given in (21) and most of what follows is also applicable to those discretizations. In the next section the *dual variable method* is presented which replaces the $(L+N)\times(L+N)$ system (21) by an equivalent $(L-N)\times(L-N)$ system. Note that $L\doteq 2N$ and that this reduction in dimension is a factor of 3.

4. The dual variable method. Many discretizations of the Navier-Stokes equations (1)-(2), and in particular that used in [1], [8], lead to a system of the generic form (21). The second block row is termed the *discrete continuity equation*

$$A\mathbf{V}^{m+1} = \mathbf{S}_m \tag{22}$$

and the first row the *discrete momentum* equation

$$Q_m\mathbf{V}^{m+1} = -A^T\mathbf{P}^{m+1} + \mathbf{K}_m. \tag{23}$$

For the choices made in § 3 it can be shown [8] that Q_m is diagonally dominant and hence of rank L. Further, if there is one segment of $\partial\Omega$ on which the pressure is specified (normal velocity to be determined) then A is of rank N, ([8], see also § 5 below). Hence, the matrix in (20) is of rank $L+N$ and the system has a unique solution.

Implicit finite difference equations, such as those developed in [8], while relatively simple to formulate, are a set of $(L+N)$ equations which are so complicated that the cost of solution may offset any savings realized by allowing the use of a large time step. The dual variable method [1], [5], introduces a set of auxiliary variables in the implicit equations which reduces the computational problem to one of solving a system of $L-N$ equations.

Assuming that (21) or equivalently (22)-(23) has a unique solution and ignoring for a moment all motivation for what follows, the dual variable method consists of the following observations and purely algebraic steps.

Step 1. A solution $\mathbf{V}^{m+1}$ to the $N\times L$ system (22) must be of the form

$$\mathbf{V}^{m+1} = \mathbf{V}_P^{m+1} + \mathbf{V}_H^{m+1} \tag{24}$$

where $\mathbf{V}_P^{m+1}$ is a particular solution of (22) and $\mathbf{V}_H^{m+1}$ is a solution to the homogeneous system $A\mathbf{V}_H^{m+1} = \mathbf{0}$. This is sophomoric linear algebra. Hence, the first step is the determination of a particular solution $\mathbf{V}_P^{m+1}$ to (22).

Step 2. Find a basis $\{\mathbf{C}_1, \mathbf{C}_2, \cdots, \mathbf{C}_R\}$ for the null space of A and form the $L \times R$ matrix C with $\mathbf{C}_i$ as its ith column. Then

$$AC = 0 \tag{25}$$

and

$$\mathbf{V}_H^{m+1} = C\mathbf{X}^{m+1} \tag{26}$$

for *some* $R \times 1$ vector $\mathbf{X}^{m+1}$.

Step 3. Substitute $\mathbf{V}^{m+1}$ from (24) into (23) to obtain the system

$$Q_m C\mathbf{X}^{m+1} = -A^T\mathbf{P}^{m+1} + (\mathbf{K}_m - Q_m\mathbf{V}_P^{m+1}). \tag{27}$$

Step 4. Multiply (27) by C^T and use (23) to obtain the $R \times R$ system

$$(C^T Q_m C)\mathbf{X}^{m+1} = \mathbf{B}_m \tag{28}$$

where $\mathbf{B}_m = C^T(\mathbf{K}_m - Q_m\mathbf{V}_P^{m+1})$. The matrix transformation in (28) is termed the *dual variable transformation* and (28) itself is termed the *dual variable system.*

Step 5. Solve (28) for $\mathbf{X}^{m+1}$, recover the velocities $\mathbf{V}^{m+1}$ from (24) and the pressures $\mathbf{P}^{m+1}$ from (25).

The following questions should occur to the astute reader's mind and are addressed below and in subsequent sections:

- How can a particular solution $\mathbf{V}_P^{m+1}$ to (22) be found efficiently?
- What is the dimension R of the null space of A?
- How can a basis $\mathbf{C}_1, \cdots, \mathbf{C}_R$ for the null space of A be found efficiently?
- Can such a basis be found so that $C^T Q_m C$ is sparse and such that $C^T Q_m C$ can be formed efficiently?
- Is $C^T Q_m C$ nonsingular and can (26) be solved efficiently?
- How can (25) be solved efficiently for $\mathbf{P}^{m+1}$?

The inherent advantage of the dual variable transformation is the considerable reduction in the size ($L+N$ to R) of the system to be solved and hence a nontrivial reduction in the computational cost per time step. If this advantage is to be of any real consequence then the cost of the subsidiary calculations to obtain $\mathbf{V}_P^{m+1}$ and C must be minimal. That this latter cost is modest is demonstrated in the next section. See also [1], [8].

Next consider the solvability of the dual variable system (26). Recalling that C is $L \times R$ and of rank R and Q_m is $L \times L$ and nonsingular, we seek conditions on Q_m such that the $R \times R$ matrix $C^T Q_m C$ is nonsingular. The example

$$[1, 0]\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}\begin{bmatrix} 1 \\ 0 \end{bmatrix} = 0$$

shows that the nonsingularity of Q is not sufficient. On the other hand, it is well-known that *$C^T Q_m C$ is nonsingular if Q_m is positive definite.* Unfortunately, Q_m derived as in § 3 is *not* positive definite nor even symmetric. The following results from [2] do establish sufficient conditions for the nonsingularity of $C^T Q_m C$.

THEOREM 1 [2]. *If $\mathbf{Y}^T Q_m \mathbf{Y} \neq 0$ for all nonzero $\mathbf{Y}$ in the range $\mathcal{R}(C)$ of the matrix C, then $C^T Q_m C$ is nonsingular.*

Proof by contradiction. Let $\boldsymbol{\gamma} \neq \mathbf{0}$ be an $R \times 1$ vector such that $C^T Q_m C\boldsymbol{\gamma} = \mathbf{0}$. Then $Q_m C\boldsymbol{\gamma}$ belongs to the orthogonal complement of $\mathcal{R}(C)$. If $\mathbf{Y} \equiv C\boldsymbol{\gamma}$, then $\mathbf{Y} \neq \mathbf{0}$, $\mathbf{Y}$ belongs to $\mathcal{R}(C)$ and

$$\mathbf{Y}^T Q_m \mathbf{Y} = \mathbf{Y}^T Q_m C\boldsymbol{\gamma} = 0,$$

which contradicts the hypothesis. Q.E.D.

The hypothesis of the above theorem may be difficult to verify. However, the following theorem which is a corollary of Theorem 1 is directly applicable to the dual variable system.

THEOREM 2 [2]. *If Q_m has positive diagonal and is row and column diagonally dominant with strict diagonal dominance in either the rows or columns, then C^TQ_mC is nonsingular.*

Proof. Let $Q_m=[q_{ij}]$ and assume for definiteness that it is rowwise strictly diagonally dominant. Then for any $L\times 1$ vector $\mathbf{Y}$,

$$\mathbf{Y}^TQ_m\mathbf{Y}=\tfrac{1}{2}\mathbf{Y}^T(Q_m+Q_m^T)\mathbf{Y}. \tag{29}$$

But for $k=1,\cdots,L$ we have $|q_{kk}|>\sum_{j\neq k}|q_{kj}|$ and $|q_{kk}|\geqq\sum_{j\neq k}|q_{jk}|$. Thus

$$2|q_{kk}|>\sum_{j\neq k}|q_{kj}|+\sum_{j\neq k}|q_{jk}|\geqq\sum_{j\neq k}|q_{kj}+q_{jk}|.$$

Thus $\frac{1}{2}(Q_m+Q_m^T)$ is strictly diagonally dominant and consequently positive definite. Therefore, by (29) $\mathbf{Y}^TQ_m\mathbf{Y}>0$ for all $\mathbf{Y}\neq 0$ and by Theorem 1, C^TQ_mC is nonsingular. Q.E.D.

In [8, pp. 6–18] it is shown that the matrix Q_m in (23) satisfies the conditions of Theorem 2 for all Δt. It is also shown in [8] that C^TQ_mC is a very sparse matrix when C is chosen as indicated in the next section.

5. Networks and the dual variable transformation. The essential ingredient of the dual variable method is the transformation in (26) which recasts the implicit finite difference equations (22)–(23) into an equation in terms of the vector $\mathbf{X}^{m+1}$, the entries of which are termed *dual variables.* T. Porsching [16] first observed that the matrix A is an incidence matrix of a directed network associated with the finite difference grid. Various results from network theory then provide for efficient construction of the transformations involved, hence preserving the cost savings realized in reducing the size of system to be solved at each time step.

The directed planar network $\mathcal{N}$ of interest has a geometric realization $G(\mathcal{N})$ in which the nodes are the mesh box centers, and the *interior links* connect the nodes of contiguous mesh boxes and are directed in the positive sense of the x or y axis. The *boundary links* of $\mathcal{N}$ are those links of $G(\mathcal{N})$ that are normal to segments of $\partial\Omega$ where a pressure is specified.

For example, consider a channel with no slip walls containing an obstacle as illustrated in Fig. 3.

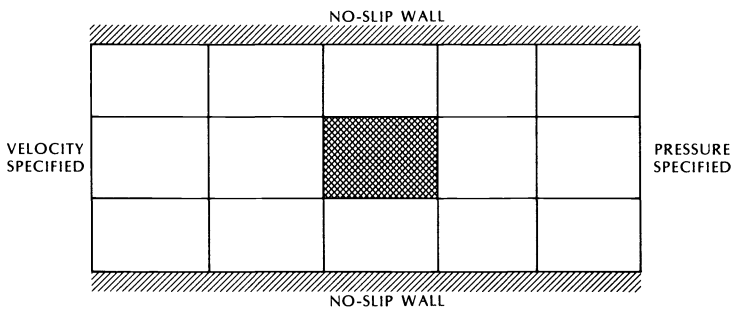


FIG. 3. *Channel with obstacle.*

This flow region is decomposed into 14 control volumes. Assume the boundary conditions are as specified, then the associated network $\mathcal{N}$ is as illustrated in Fig. 4.

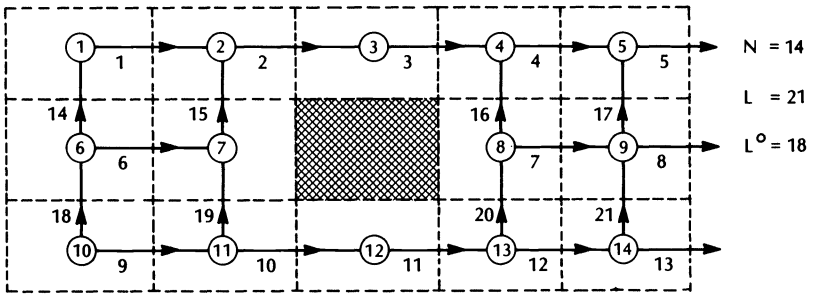


FIG. 4. *Flow region* Ω *and associated network* $G(\mathcal{N})$.

The number N of nodes is precisely the number of unknown pressures and the number L of links is precisely the number of unknown velocities.

Recall that $A=[a_{ij}]$ is an *incidence matrix* of the directed network $\mathcal{N}$ if

$$a_{ij}=\begin{cases}+1 & \text{if link } j \text{ is incident from node } i,\\ -1 & \text{if link } j \text{ is incident to node } i,\\ 0 & \text{otherwise.}\end{cases} \tag{30}$$

The incidence matrix for the network $\mathcal{N}$ in Fig. 4 is the 14×21 matrix

$$A=\begin{array}{c}\begin{array}{cccccccccccccccccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & 16 & 17 & 18 & 19 & 20 & 21 & \end{array}\\ \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}\begin{array}{c}1\\2\\3\\4\\5\\6\\7\\8\\9\\10\\11\\12\\13\\14\end{array}\end{array} \tag{31}$$

Notice that the columns of A corresponding to interior links have exactly two nonzero entries, $+1$ and -1, while the columns corresponding to boundary links have exactly one nonzero entry.

The study of flows on a network is greatly facilitated by the use of the incidence matrix. It provides a compact, concise way of mathematically stating that the flowing quantity is conserved at the nodes of the network. The matrix A in the continuity equation (22) for the mesh in Fig. 4 is, as observed in [16], precisely the incidence matrix A in (31) for $G(\mathcal{N})$ and the ith equation in (22) states that the net flow into the ith control volume is the ith entry of $\mathbf{S}_m$; cf. (20). For example, the net flow into control volume 6 (row 6 of A), $(\mathbf{V}_6+\mathbf{V}_{14}-\mathbf{V}_{18})$, is balanced by the specified flow into this control volume from outside of Ω. The net flow into control volume 9 (row 9 of A), $(-\mathbf{V}_7+\mathbf{V}_8+\mathbf{V}_{17}-\mathbf{V}_{21})$, is zero and the corresponding entry of $\mathbf{S}_m$ is zero.

That the rank of A in (31) is 14 follows almost by inspection, however, it is also a consequence of the following result.

THEOREM 3. *If $G(\mathcal{N})$ has at least one boundary link then the incidence matrix A is of rank N.*

Proof. Suppose $\boldsymbol{\alpha}^T A = \mathbf{0}$. We show $\boldsymbol{\alpha} = \mathbf{0}$ and hence the rows are linearly independent. By hypothesis, there is a node k which is the extremity of a boundary link g (e.g. node 5 and link 5 in Fig. 4). Since a_{kg} is the only nonzero entry in column g of A we must have $\alpha_k = 0$.

Discard node k, as well as any boundary links of which it is an extremity. Assume for ease of exposition that there is one such boundary link. We create a sub-network whose incidence matrix is an $(N-1)\times(L-1)$ submatrix of A. The subnetwork also has a boundary link which can be used to deduce another entry of α is zero.

Repeating this process a finite number of times, we deduce that $\boldsymbol{\alpha} = \mathbf{0}$. Q.E.D.

From Theorem 3 we conclude that the matrix C used in the dual variable transformation (27) is $L\times(L-N)$ and has rank $R = L-N$. Further, the dual variable system is $(L-N)\times(L-N)$ as promised.

Next consider the task of actually constructing the matrix C, or equivalently, a basis for the null space of A. Again network theory plays a key role.

A *cycle* of $\mathcal{N}$ is a chain of interior links whose extremities coincide and is such that any other node is encountered at most once during a traverse of the chain. We can associate a *cycle vector* $\mathbf{c}_k = (c_{1k}, \cdots, c_{LK})^T$ with cycle k by the definition

$$c_{jk} = \begin{cases} +1 & \text{if link } j \text{ has a positive orientation during a counterclockwise traverse of the chain,} \\ -1 & \text{if link } j \text{ has a negative orientation during a counterclockwise traverse of the chain,} \\ 0 & \text{otherwise.} \end{cases} \tag{32}$$

From Fig. 4, the network $\mathcal{N}$ has a cycle containing four links with cycle vector:

$$(-1\ 0\ 0\ 0\ 0\ +1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ -1\ +1\ 0\ 0\ 0\ 0\ 0\ 0)^T; \tag{33}$$

and a cycle containing eight links with cycle vector:

$$(0\ -1\ -1\ 0\ 0\ 0\ 0\ 0\ 0\ +1\ +1\ 0\ 0\ 0\ -1\ +1\ 0\ 0\ -1\ +1\ 0)^T. \tag{34}$$

THEOREM 4 [3]. *If there are L^0 interior links in $\mathcal{N}$ then there are $L^0 - N + 1$ linearly independent cycle vectors.*

Proof. See [3, p. 124].

For a planar network $\mathcal{N}$ define a *country* (or face) of $\mathcal{N}$ as a finite region of the plane bounded by links which contains neither nodes nor links in its interior. The *boundary* of a country is the cycle formed by the links which surround it. The two cycle vectors given in (33) and (34) are boundaries of countries, however the following is not,

$$(-1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ +1\ 0\ 0\ 0\ 0\ -1\ +1\ 0\ 0\ -1\ +1\ 0\ 0)^T. \tag{35}$$

The region bounded by the links in this last cycle contains the link numbered 6 in its interior.

THEOREM 5 [3]. *$L^0 - N + 1$ linearly independent cycle vectors are obtainable from cycles each of which is the boundary of some country.*

Proof. See [3, p. 136].

With regard to boundary links, there are $L - L^0 - 1$ chains such that the first and last links of the kth chain are respectively the kth and $(k+1)$st boundary links. (In Fig. 4, the boundary links are links 5, 8 and 13.) Such chains are called *pseudo-cycles*

and the kth one defines the *pseudo-cycle vector* $\mathbf{d}_k = (d_{1k}, \cdots, d_{Lk})^T$ where the d_{ik} are defined similar to the c_{jk} above. For example, the network $\mathcal{N}$ in Fig. 4 has a pseudo-cycle with pseudo-cycle vector:

$$(36) \qquad (0\ 0\ 0\ 0\ -1\ 0\ 0\ +1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ -1\ 0\ 0\ 0\ 0)^T.$$

Each $\mathbf{d}_k$ is independent of the other pseudo-cycle vectors *and* all cycle vectors since it contains a nonzero component where they do not. Hence:

THEOREM 6 [8]. *Let C be the $L\times(L-N)$ matrix whose columns are (L^0-N+1) linearly independent cycle vectors generated from boundaries of countries and $(L-L^0-1)$ pseudo-cycle vectors. Then, the rank of C is $L-N$ and the columns of C form a basis for the null space of A.*

Proof. That $A^TC=0$ is proven in [8].

The matrix is termed a *fundamental matrix of* $\mathcal{N}$. For the network in Fig. 4 we can choose C to be the 21×7 matrix:

$$(37) \qquad C = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \\ 10 \\ 11 \\ 12 \\ 13 \\ 14 \\ 15 \\ 16 \\ 17 \\ 18 \\ 19 \\ 20 \\ 21 \end{array} \begin{bmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 \\ +1 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & +1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & +1 & -1 \\ 0 & 0 & 0 & +1 & 0 & 0 & 0 \\ 0 & +1 & 0 & 0 & 0 & 0 & 0 \\ 0 & +1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & +1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & +1 \\ -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ +1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & +1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & +1 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & +1 & 0 & 0 & 0 \\ 0 & +1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & +1 & 0 & -1 \end{bmatrix}.$$

One can verify directly that $AC=0$ for A in (31) and C in (37). For the flow region in Fig. 3, the system (21) is 35×35 while the dual variable system (28) is 7×7.

It is most fortuitous that in practice the fundamental matrix C need not be formed explicitly as an $L\times(L-N)$ matrix. The transformed matrix C^TQ_mC in (28) can be formed directly once the labels of the (at most) two countries which share each one of the L links are known. See [2] for details of this construction as well as a discussion of the structure of C^TQ_mC. Suffice to state here that C^TQ_mC is a sparse border-banded matrix and system (28) can be solved efficiently using for example the frontal method.

The first step of the dual variable method requires the construction of a particular solution to the discrete continuity equation (22). Such a solution is easily determined by using a *spanning tree* of $\mathcal{N}$, and algorithms for the determination of a spanning tree are well known [3]. With the spanning tree available, one sets the velocities on the

links of $\mathcal{N}$ which are not in the tree equal to zero. Also, the velocities are set to zero on boundary links.

Beginning with its outermost extremities, one then proceeds through the nodes of the tree so that as each node is encountered, all but one velocity component associated with links incident on that node has been determined. The continuity equation is used to determine the remaining velocity, taking into account any specified boundary velocity components.

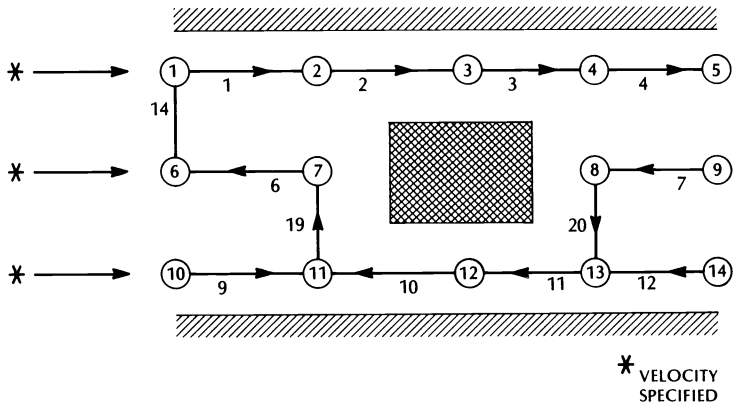


FIG. 5. *Spanning tree for* $\mathcal{N}$.

For the network in Fig. 4, a spanning tree is given in Fig. 5. If the continuity equations in (22) are ordered as the nodes are encountered above in the tree, (14, 9, 8, 13, 12, 10, 11, 7, 6, 1, 2, 3, 4, 5), then the unknown velocities can be ordered (12, 7, 20, 11, 10, 9, 19, 6, 14, 1, 2, 3, 4) so that the system to be solved is triangular.

Hence, the construction of a particular solution of the discrete continuity equation is an easy task once a tree is determined. This same tree can be used to recover the pressures from the pressure drops $A^T\mathbf{P}^{m+1} = -Q_m\mathbf{V}^{m+1} + \mathbf{K}_m$ (cf. equation (23)).

Two final comments on the dual variable method. First, numerical experimentation [5], [9], [10] indicates that there is considerable cost savings when the dual variable method is applied to two-dimensional transient problems. Second, the dual variable method can also be applied to compressible flow, steady flow and three-dimensional flow problems.

6. Sample flow problems. Three examples are now presented in which the dual variable method has been successfully applied on a rather large scale to practical, real world engineering fluid flow problems. The reader is referred to [8], [10], [16] for more details.

A. Flow of exhaust gases in an automative catalytic converter. Fig. 6 illustrates a cross-section of a GM bead-bed catalytic converter. These converters are packed with some 250,000 one-eighth inch porous spheres on which have been deposited small amounts of platinum. Hot engine exhaust gases pass into the converter and then down through the matrix of pellets where catalytic oxidation takes place to reduce the constituent CO and HC emissions.

The accurate prediction of this chemical process requires knowledge of the dynamic behavior of the exhaust gases as functions of converter geometry, pellet diameter, etc. Ignoring the feedback of the chemical reaction, a reasonable flow pattern can be

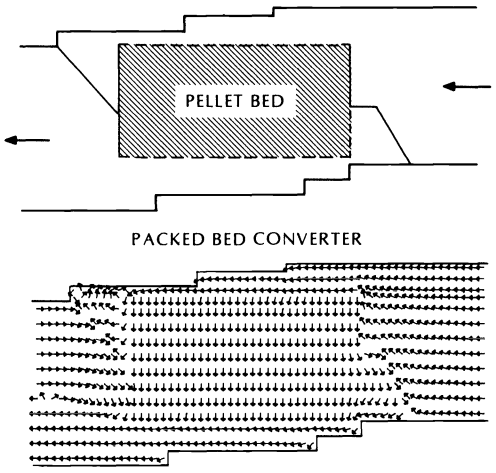


FIG. 6. *Catalytic converter geometry and flow field.*

achieved by an incompressible forced flow model. Fig. 6 contains the velocity profile for such a problem. The number of flow cells is $N = 743$, there are $L^0 = 1{,}412$ interior links and $L = 1{,}423$ total links. Hence, the implicit finite difference system (21) is of order $L + N = 2{,}166$ while the dual variable system (28) is of order $L - N = 680$.

B. Flow of a two-phase mixture in a preheater section of a steam generator. Simulation of two-phase flow is essential for accurate prediction of the transient and steady state performance of the nuclear steam supply system components of nuclear power plants. The system of equations given in (3) model such flows under the assumption that the two phases, steam and water, are a single homogeneous mixture. The mixture is assumed to be thermally expandable; density varies with enthalpy.

Figure 7 is a schematic of the preheater section of a steam generator. Thousands of tubes containing the primary "hot" fluid intersect this flow region in the vertical direction. These tubes are modeled by resistance terms in the momentum equations and they provide sources of heat for the thermal energy equation. We are modeling the secondary fluid which is returning "cold" through the feed water nozzle, circulates through this region from bottom to top, and exits through openings in the tube support plate. The secondary fluid is heated as it passes around the "hot" tubes. The baffles slow the flow, so that sufficient heat is transferred to produce steam near the top of the flow region. These baffles are interesected by the tubes around which there are openings permitting some vertical flow.

The side walls and deflection plate are modeled as free-slip walls, while the baffles and support plate are modeled by means of form loss terms in the momentum equations.

This problem involved $N = 385$ flow cells $L^0 = 715$ interior links and $L = 735$ total links. Hence, equation (21) is of order 1,120 while the dual variable system (28) is of order 350.

The steam-water interface is of great interest in such applications. Fig. 7 contains illustrations of the two phases for different times. The shaded region is steam.

C. Flow of a two gas mixture through an axially symmetric centerbody combustor. Mathematical combustor models of combustion tunnels are being employed to provide information about performance trends of gas turbine engines and to predict velocity, pressure and thermodynamic property profiles in simulated practical combustion environments.

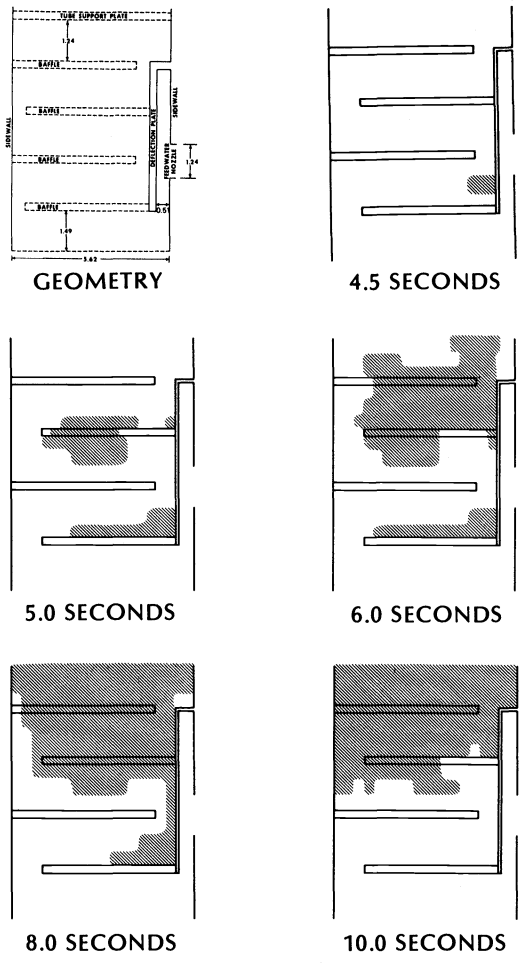


FIG. 7. *Transient development of steam in a preheater section of a steam generator.*

Figure 8 is a schematic of an axi-symmetric centerbody combustor consisting of a 0.4592 ft diameter cylindrical bluff-body placed concentrically in a 0.8332 ft diameter duct. Air is forced through the duct around the centerbody and gaseous fuel (propane) or an inert gas (CO_2) is injected at the center of the centerbody downstream face through a 0.01575 ft diameter tube.

For noncombusting flows predictions of the mixture velocity, pressure, temperature and density as well as the fuel concentration are sought in the region immediately downstream of the centerbody as illustrated in Fig. 8. The system of partial differential equations consists of axially symmetric (r, z) analogues of (3), the ideal gas law and a transport of fuel concentration equation.

Figure 8 shows typical streamlines for flow past the centerbody. There are two torodial vortices; the fuel vortex is trapped in front of the air vortex forcing the two gases to be mixed.

This problem involves $N = 390$ flow cells, $L^0 = 739$, and $L = 769$. Hence, the system (21) is of order 1,159 while the dual variable system (28) is of order 379.

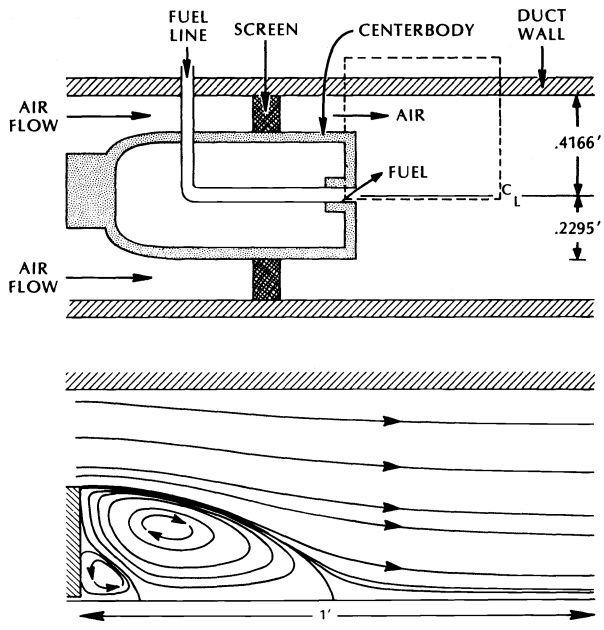


FIG. 8. *Centerbody combustor geometry and sample flow field.*

REFERENCES

[1] R. AMIT, C. A. HALL AND T. A. PORSCHING, *An application of network theory to the solution of implicit Navier-Stokes difference equations*, J. Comp. Phys., 40 (1981), pp. 183-201.

[2] R. AMIT, C. CULLEN, C. HALL, G. MESINA AND T. PORSCHING, *Flow problems, finite differences and the frontal method*, in Proc. of Third International Conference on Finite Elements in Flow Problems, Banff, Alberta, Canada, 1980.

[3] C. BERGE AND A. GHOUILA-HOURI, *Programming, Games and Transportation Networks*, Methuen, London, 1965.

[4] I. CHRISTIE, D. F. GRIFFITHS, A. R. MITCHELL AND O. C. ZIENKIEWICZ, *Finite element methods for second order differential equations with significant first derivatives*, Int. J. Num. Meth. Eng., 10 (1976), pp. 1389-1396.

[5] R. S. DOUGALL, C. A. HALL AND T. A. PORSCHING, DUVAL: *A computer program for the numerical solution of two-dimensional, two-phase flow problems*, Volumes 1-3, Electric Power Research Institute, Report NP-2099, Palo Alto, CA, April, 1982.

[6] G. E. FORSYTHE AND W. R. WASOW, *Finite Difference Methods for Partial Differential Equations*, John Wiley, New York, 1960.

[7] P. M. GRESHO AND R. L. LEE, *Don't suppress the wiggles—they're telling you something*, Comp. and Fluids, 9 (1981), pp. 223-253.

[8] C. A. HALL, T. A. PORSCHING AND R. S. DOUGALL, *Numerical methods for thermally expandable two-phase flow—computational techniques for steam generator modeling*, Electric Power Research Institute, Report NP-1416, Palo Alto, CA, May, 1980.

[9] C. A. HALL AND T. A. PORSCHING, DUVAL: *A computer program for the implicit treatment of two-dimensional two-phase fluid transients*, Institute for Computational Mathematics and Applications, Report 81-25, Univ. Pittsburgh, Pittsburgh, PA, August, 1981.

[10] ———, *Non-isothermal flow through an axially symmetric centerbody combustor via the dual variable method*, Institute for Computational Mathematics and Applications, Report 82-38, Univ. Pittsburgh, Pittsburgh, PA, May, 1982.

[11] F. H. HARLOW AND F. E. WELCH, *Numerical calculations of time dependent viscous incompressible flow of fluid with a free surface*, Phys. Fluids, 8 (1965), p. 2182.

[12] J. HEINRICH, P. HUYAKORN, O. ZIENKIEWICZ AND A. MITCHELL, *An upwind finite element scheme for two-dimensional convective transport equations*, Int. J. Num. Meth. Eng., 11 (1977), pp. 131-143.
[13] J. HEINRICH AND O. ZIENKIEWICZ, *Quadratic finite element schemes for two-dimensional convective-transport problems*, Int. J. Num. Meth. Eng., 11 (1977), pp. 1831-1844.
[14] P. J. ROACHE, *Computational Fluid Dynamics*, Hermosa, Albuquerque, NM, 1976.
[15] ———, *Recent developments and problem areas in computational fluid dynamics*, in Computational Mechanics, Lecture Notes in Mathematics 461, A. Dold and B. Eckmann, eds., Springer-Verlag, New York, 1975.
[16] T. A. PORSCHING, *A finite difference method for thermally expandable fluid transients*, Nucl. Sci. Eng., 64 (1977), pp. 177-186.
[17] R. VAN MISES AND K. O. FRIEDRICHS, *Fluid Dynamics*, Applied Mathematical Sciences, Springer-Verlag, New York, 1971.

Note added in proof. After the presentation of this paper, the author became aware of the following null space papers:

[1] K. GUSTAFSON AND R. HARTMANN, *Divergence-free bases for finite element schemes in hydrodynamics*, SIAM J. Numer. Anal., 20 (1983), pp. 697-721.
[2] M. BERRY, I. KANEKO, M. LAWO AND R. PLEMMONS, *An algorithm to compute a sparse basis of the null space*, submitted for publication, March, 1984.
[3] I. KANEKO, M. LAWO AND G. THIERAUF, *On computational procedures for the force method*, Int. J. Num. Meth. Eng., 19 (1982), pp. 1469-1495.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 2, April 1985

A DECOMPOSITION AND SCALING-INEQUALITY FOR LINE-SUM-SYMMETRIC NONNEGATIVE MATRICES*

GEORGE B. DANTZIG†, B. CURTIS EAVES† AND URIEL G. ROTHBLUM‡

Abstract. A matrix B is called *line-sum-symmetric* if it is square and the sum of elements in each row of B equals the sum of elements in the corresponding column. Results from the theory of network flows are used to obtain a decomposition of nonnegative line-sum-symmetric matrices. The decomposition is employed to prove the following inequality: Assume $D(x)AD(y)$ is line-sum-symmetric where A is a square nonnegative matrix and $D(x)$ and $D(y)$ are diagonal matrices whose diagonal elements are the coordinates of the nonnegative vectors x and y, respectively. Then $y^TAx \geqq x^TAy$.

AMS(MOS) subject classifications. primary 15A39, secondary 15A63

1. Introduction. A (real) matrix B is called *line-sum-symmetric* if it is square and the sum of the elements in each row of B equals the sum of the elements in the corresponding column of B. A *scaling* of an $n \times n$ nonnegative matrix A is a matrix having the form DAE where D and E are $n \times n$ nonnegative diagonal matrices. For a nonnegative vector x in R^n, let $D(x)$ denote the corresponding $n \times n$ diagonal matrix whose diagonal elements are the coordinates of x. The purpose of this paper is to show that if the scaling $D(x)AD(y)$ of the $n \times n$ nonnegative matrix A is line-sum-symmetric, where x and y are nonnegative vectors in R^n, then $y^TAx \geqq x^TAy$. This inequality is used in Eaves (1984) to compute an equilibrium for the linearization of a pure exchange economy with Cobb–Douglas preferences.

Our proof of the above inequality relies on a decomposition of nonnegative line-sum-symmetric matrices. Specifically, we call an $n \times n$ matrix C a *simple circuit matrix* if there exist distinct integers $i_1, \cdots, i_k$, in $\{1, \cdots, n\}$ such that

$$B_{ij} = \begin{cases} 1 & \text{if } (i,j) = (i_t, i_{t+1}) \quad \text{for some } t \in \{1, \cdots, k\}, \\ 0 & \text{otherwise,} \end{cases}$$

where $i_{k+1} \equiv i_1$. We use known results concerning network flows to show that every nonnegative line-sum-symmetric matrix is a nonnegative combination of simple circuit matrices. Of course, this decomposition resembles Birkhoff's celebrated result that every doubly stochastic matrix is a convex combination of permutation matrices. Implicit in this decomposition of nonnegative line-sum-symmetric matrices is a characterization of the extreme rays of the cones of such matrices as the sets of the form $\{\alpha C: \alpha \geqq 0\}$, where C is a circuit matrix (cf. Saunders and Schneider (1979, p. 532)).

Necessary and sufficient conditions for the existence of a doubly stochastic scaling of a square nonnegative matrix, where the corresponding diagonal matrices have positive diagonal elements, were obtained by Brualdi, Parter and Schneider (1966) and, independently, by Sinkhorn and Knopp (1967). In particular, such scalings exist for matrices all of whose components are positive. Of course, every doubly stochastic

* Received by the editors January 10, 1984, and in revised form February 16, 1984. Research and reproduction of this report were partially supported by National Science Foundation grants MCS-8119774, MCS-7926009, ECS-8012974, MCS-8121838, U.S. Department of Energy contract DE-AMO3-76SF00326, PA # DE-ATO3-76ER72018, Office of Naval Research contract N00014-75-C-0267 at Stanford University; and National Science Foundation grant ECS-7825182 at Yale University.

† Department of Operations Research, Stanford University, Stanford, California 94305.

‡ Faculty of Industrial Engineering and Management, Technion–Israel Institute of Technology, Haifa 32000, Israel.

matrix is line-sum-symmetric; thus, the above results establish sufficient conditions for the existence of line-sum-symmetric scalings of square nonnegative matrices. Necessary and sufficient conditions for the existence of a line-sum-symmetric scaling of a square nonnegative matrix are given in Eaves, Hoffman, Rothblum and Schneider (1984).

2. Line-sum-symmetric matrices. Let n be a positive integer, let $N \equiv \{1, \cdots, n\}$ and let G be the *complete graph* with node set N and arc set $N \times N$. A *flow* on G is a function z which assigns a nonnegative number to each arc of G such that for $i \in N$,

$$\sum_{j \in N} z(i,j) = \sum_{j \in N} z(j,i). \tag{1}$$

If z is regarded as an $n \times n$ (nonnegative) matrix, (1) states that this matrix is line-sum-symmetric. Indeed, it follows that there is a one-to-one correspondence between flows on G and $n \times n$ nonnegative line-sum-symmetric matrices.

Let $i_1, \cdots, i_k$, with $k \geqq 0$, be a sequence of distinct integers in N. We call the collection of arcs $\{(i_t, i_{t+1}): t = 1, \cdots, k\}$, where $i_{k+1} \equiv i_1$, a *simple circuit on G*. Given a simple circuit $C \subseteq N \times N$ on G, we define the *simple circuit flow corresponding to C* by

$$z(a) = \begin{cases} 1, & a \in C, \\ 0, & a \in N \times N \backslash C. \end{cases}$$

Evidently, under the one-to-one correspondence between flows on G and $n \times n$ line-sum-symmetric matrices, simple circuit flows correspond to $n \times n$ simple circuit matrices.

The theorem below is concerned with expressing a nonnegative line-sum-symmetric matrix B as a *nonnegative combination of simple circuit matrices*, i.e., it is shown that for such a matrix B there exist simple circuit matrices $C_1, \cdots, C_m$ and nonnegative numbers $\theta_1, \cdots, \theta_m$ such that $B = \sum_{i=1}^{m} \theta_i C_i$. We say that C_i *is used in* such a combination if the coefficient θ_i is positive.

THEOREM 1. *Let B be an $n \times n$ nonnegative matrix. Then B is line-sum-symmetric if and only if B is a nonnegative combination of simple circuit matrices. Moreover, $B_{ij} = 0$ if and only if $C_{ij} = 0$ for each simple circuit matrix C used in such a combination.*

Proof. It is well known that every flow on a graph is a nonnegative combination of simple circuit flows (e.g., Denardo (1982, p. 99)). Thus, the correspondence between $n \times n$ nonnegative line-sum-symmetric matrices and flows on G immediately yields the asserted decomposition of such matrices. The characterization of the above matrices for which given coordinates vanish follows immediately. □

The following are immediate conclusions of Theorem 1. Let $\alpha \subset N \times N$. Then there exists an $n \times n$ nonnegative line-sum-symmetric matrix $B \neq 0$ with $B_{ij} = 0$ for all $(i,j) \in \alpha$ if and only if there exists such a simple circuit matrix. Also, for $\alpha \subseteq N \times N$, there exists an $n \times n$ nonnegative, line-sum-symmetric matrix B with $B_{ij} > 0$ for each $(i,j) \in \alpha$ if and only if for every $(i,j) \in \alpha$ there exists a simple circuit matrix C with $C_{ij} > 0$. These results were obtained by Saunders and Schneider (1979, Thms. 2.3 and 2.5).

3. The inequality concerning line-sum-symmetric scalings of square nonnegative matrices. Before establishing the inequality concerning line-sum-symmetric scalings of square nonnegative matrices, we need a few additional definitions.

Given a vector $x \in R^n$, we denote by $D(x)$ the $n \times n$ diagonal matrix whose diagonal elements are $x_1, \cdots, x_n$, i.e.,

$$D(x)_{ij} = \begin{cases} x_i & \text{if } i = j, \\ 0 & \text{if } i \neq j. \end{cases}$$

Also, we denote by $x^\dagger$ the vector in R^n defined by

$$(x^\dagger)_i = \begin{cases} (x_i)^{-1} & \text{if } x_i \neq 0, \\ 0 & \text{if } x_i = 0. \end{cases}$$

Evidently, for $x \in R^n$,

$$D(x)e = x, \tag{2}$$

where $e = (1, \cdots, 1)^T$. Also, for $A \in R^{n\times n}$ and $x, y \in R^n$, we have that

$$[D(x)AD(y)]_{ij} = x_i A_{ij} y_j, \qquad i, j = 1, \cdots, n. \tag{3}$$

Finally, for $x \in R^n$, we have that $D(x)D(x^\dagger) = D(x^\dagger)D(x)$ is the $n \times n$ diagonal matrix whose iith component is 1 if $x_i \neq 0$ and 0 if $x_i = 0$. In particular,

$$D(x)D(x^\dagger) = D(x^\dagger)D(x) \leqq I. \tag{4}$$

THEOREM 2. *Let A be an $n \times n$ nonnegative matrix and let x and y be two nonnegative vectors in R^n where $D(x)AD(y)$ is line-sum-symmetric. Then $y^TAx \geqq x^TAy$.*

Proof. We first establish the conclusion of the theorem for the case where $D(x)AD(y)$ is a simple circuit matrix. In this case there exist distinct integers, say i_1, $i_2, \cdots, i_k$, such that

$$\{(i,j): A_{ij} \neq 0\} = \{(i,j): [D(x)AD(y)]_{ij} \neq 0\} = \{(i,j): [D(x)AD(y)]_{ij} = 1\}$$
$$= \{(i_1, i_2), (i_2, i_3), \cdots, (i_k, i_{k+1})\},$$

where $i_{k+1} \equiv i_1$. In particular, with $K \equiv \{1, \cdots, k\}$

$$x_{i_t} A_{i_t i_{t+1}} y_{i_{t+1}} = 1, \qquad t \in K. \tag{5}$$

We conclude that

$$x^TAy = \sum_{i,j\in N} x_i A_{ij} y_j = \sum_{t\in K} x_{i_t} A_{i_t i_{t+1}} y_{i_{t+1}} = k \tag{6}$$

and that

$$y^TAx = \sum_{i,j\in N} y_i A_{ij} x_i = \sum_{t\in K} y_{i_t} A_{i_t i_{t+1}} x_{i_{t+1}}. \tag{7}$$

In particular, (7) and the fact that the arithmetic mean is always larger than or equal to the geometric mean imply that

$$\begin{aligned} k^{-1}y^TAx &= k^{-1} \sum_{t\in K} y_{i_t} A_{i_t i_{t+1}} x_{i_{t+1}} \geqq \left(\prod_{t\in K} y_{i_t} A_{i_t i_{t+1}} x_{i_{t+1}} \right)^{1/k} \\ &= \left(\prod_{t\in K} x_{i_t} \right)^{1/k} \left(\prod_{t\in K} A_{i_t i_{t+1}} \right)^{1/k} \left(\prod_{t\in K} y_{i_t} \right)^{1/k} = \left(\prod_{t\in K} x_{i_t} A_{i_t i_{t+1}} y_{i_{t+1}} \right)^{1/k}. \end{aligned} \tag{8}$$

We conclude from (8), (5) and (6) that

$$y^TAx \geqq k \left(\prod_{t\in K}^{k} x_{i_t} A_{i_t i_{t+1}} y_{i_{t+1}} \right)^{1/k} = k = x^TAy,$$

completing our proof in the case where $D(x)AD(y)$ is a simple circuit matrix.

We finally consider the case where $D(x)AD(y)$ is an arbitrary line-sum-symmetric matrix. As $A \geqq 0$, $D(x)AD(y) \geqq 0$, and therefore Theorem 1 implies that $D(x)AD(y)$ is a (possibly vacuous) linear combination with positive coefficients of simple circuit matrices. Thus, there exist positive numbers $\theta_1, \cdots, \theta_m$ and simple circuit matrices

$C(1), \cdots, C(m)$ such that

$$D(x)AD(y) = \sum_{q=1}^{m} \theta_q C(q). \tag{9}$$

Allow $m = 0$ in order to cover the trivial case where $D(x)AD(y) = 0$.

For $q = 1, \cdots, m$, let $A(q) = D(x^\dagger)C(q)D(y^\dagger)$. We next claim that

$$D(x)A(q)D(y) = C(q), \qquad q = 1, \cdots, m. \tag{10}$$

First observe that if $x_i \neq 0$ and $y_j \neq 0$, then for $q = 1, \cdots, m$, $A(q)_{ij} = (x^\dagger)_i C(q)_{ij}(y^\dagger)_j = (x_i)^{-1}C(q)_{ij}(y_j)^{-1}$, implying that $C(q)_{ij} = x_i A(q)_{ij} y_j = [D(x)A(q)D(y)]_{ij}$. Alternatively, if either $x_i = 0$ or $y_j = 0$, then $\sum_{q=1}^{m} \theta_q C(q)_{ij} = [D(x)AD(y)]_{ij} = x_i A_{ij} y_j = 0$, implying that for $q = 1, \cdots, m$, $C(q)_{ij} = 0$ and therefore $[D(x)A(q)D(y)]_{ij} = x_i A(q)_{ij} y_j = 0 = C(q)_{ij}$. This completes the proof of (10). We next conclude from the established conclusion of our theorem for simple circuit matrices, from (10) and from the fact that each $C(q)$ is a simple circuit matrix, that

$$y^T A(q)x \geqq x^T A(q)y, \qquad q = 1, \cdots, m, \tag{11}$$

and therefore

$$y^T \left[\sum_{q=1}^{m} \theta_q A(q)\right] x \geqq x^T \left[\sum_{q=1}^{m} \theta_q A(q)\right] y. \tag{12}$$

We next observe that (2), (10) and (9) imply that

$$\begin{aligned} x^T \left[\sum_{q=1}^{m} \theta_q A(q)\right] y &= e^T D(x) \left[\sum_{q=1}^{m} \theta_q A(q)\right] D(y)e \\ &= e^T \left[\sum_{q=1}^{m} \theta_q D(x)A(q)D(y)\right] e = e^T \left[\sum_{q=1}^{m} \theta_q C(q)\right] e \\ &= e^T D(x)AD(y)e = x^T Ay. \end{aligned} \tag{13}$$

Next, by (4), $D(x^\dagger)D(x) \leqq I$ and $D(y)D(y^\dagger) \leqq I$. These facts combined with (9) and the definition of $A(q)$, $q = 1, \cdots, m$, imply that

$$\begin{aligned} \sum_{q=1}^{m} \theta_q A(q) &= \sum_{q=1}^{m} \theta_q D(x^\dagger)C(q)D(y^\dagger) = D(x^\dagger)\left[\sum_{q=1}^{m} \theta_q C(q)\right] D(y^\dagger) \\ &= D(x^\dagger)D(x)AD(y)D(y^\dagger) \leqq A. \end{aligned} \tag{14}$$

We finally conclude from (14), (12) and (13) that

$$y^T Ax \geqq y^T \left[\sum_{q=1}^{m} \theta_q A(q)\right] x \geqq x^T \left[\sum_{q=1}^{m} \theta_q A(q)\right] y = x^T Ay. \qquad \square$$

Acknowledgment. The authors acknowledge R. N. Kaul for pointing out to them the fact that an earlier proof of Theorem 2 was incorrect.

REFERENCES

R. A. BRUALDI, S. V. PARTER AND H. SCHNEIDER (1966), *The diagonal equivalence of a nonnegative matrix to a stochastic matrix*, J. Math. Anal. Appl., 16, pp. 31-50.

E. V. DENARDO (1982), *Dynamic Programming: Theory and Applications*, Prentice-Hall, Englewood Cliffs, NJ.

B. C. EAVES (1984), *On linear complementary problems obtained by linearizing the economic equilibrium problem*, unpublished manuscript.

B. C. EAVES, A. HOFFMAN, U. G. ROTHBLUM AND H. SCHNEIDER (1984), *Line-sum-symmetric scalings of square nonnegative matrices*, Math. Programming, to appear.

P. KNOPP AND R. SINKHORN (1967), *Concerning nonnegative matrices and doubly stochastic matrices*, Pacific J. Math., 21, pp. 343-348.

B. D. SAUNDERS AND H. SCHNEIDER (1979), *Applications of the Gordan-Stiemke theorem in combinatorial matrix theory*, SIAM Rev., 21, pp. 528-541.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 2, April 1985

007

A NOTE ON THE NUMBER OF INVARIANT CAPITAL STOCKS*

CHIA-SHIN CHUNG†

Abstract. This paper explores a different aspect of the invariant capital stock—determining the number of invariant capital stocks. In a 2×2 model, it can be shown, that, under some reasonable assumptions, that number is odd. More restrictive condition is also given for it to be unique.

Key words. invariant capital stocks-boundary type and interior type, r-productive

1. Introduction. There has been continuing interest in the problem of computing optimal invariant capital stock, which will be referred to as an optimal stationary program (OSP) in this paper. Different solving procedures have been proposed by many under different sets of assumptions; see Hansen and Koopman [3] and Dantzig and Mann [2]. In this note, we will shift our interest to a different aspect of this problem—determining the total number of invariant capital stocks. In a 2×2 model similar to [2] and [3], we can show that, for a general concave utility function, the number of optimal invariant capital stocks is odd. It is in fact unique under more restrictive assumptions. Proofs of some lemmas in this paper can be found in Chung [1], hence will be omitted in this paper.

2. The model. We first formulate our model as follows:

Given an initial stock $X(0)$, find a program $(X(t), C(t))_{t=0}^{\infty}$ that will solve the following optimization problem:

$$\max \sum_{t=0}^{\infty} r^t u(C(t))$$

$$\text{subject to: } AX(t+1)+C(t)\leqq X(t),$$

$$eX(t+1)\leqq 1,$$

$$X(t+1), C(t)\geqq 0, \qquad t=0, 1, 2, \cdots.$$

Here $e=(1, 1)$, $r\in(0, 1)$, $X(t)=(x(t), y(t))\geqq 0$, $C(t)=(c_1(t), c_2(t))\geqq 0$ and $A=\begin{pmatrix} a & b \\ c & d \end{pmatrix}$. We also assume that $b, c>0$ and $\nabla u(C)=(u_1(C), u_2(C))>0$, for all $C\geqq 0$.

We now define the two different types of OSPs as follows:

DEFINITION. A feasible program is an OSP, if it is optimal and $(X(t), C(t))=(X, C)$ for all $t\geqq 0$. An OSP (X, C) is called a boundary OSP if either $c_1=0$ or $c_2=0$, and it is called an interior OSP if both c_1 and c_2 are positive.

We need the following assumption.

Assumption. A is r-productive, which means that $rI-A$ is a Leontief matrix.

The following lemma gives a necessary and sufficient condition for the existence of an OSP; its proof can be found in Jones [4].

LEMMA 1. *(X, C) is an* OSP *if and only if there exists $(p, w)\geqq 0$ such that the following conditions are satisfied*:

$$\text{(a)} \qquad \begin{pmatrix} A-I & I \\ e & 0 \end{pmatrix}\begin{pmatrix} X \\ C \end{pmatrix}\leqq \begin{pmatrix} 0 \\ 1 \end{pmatrix},$$

* Received by the editors July 19, 1983, and in final revised form March 5, 1984. This work contains portions of the author's Ph.D. thesis. The work was supported in part by the National Science Foundation under grant SES-7805196.

† Department of Quantitative Business Analysis, Cleveland State University, Cleveland, Ohio 44115.

$$\text{(b)} \quad (p, w)\begin{pmatrix} A-rI & I \\ e & 0 \end{pmatrix} \geqq \begin{pmatrix} 0 \\ \nabla u(C) \end{pmatrix},$$

$$\text{(c)} \quad (p, w)\left[\begin{pmatrix} 0 \\ 1 \end{pmatrix} - \begin{pmatrix} A-I & I \\ e & 0 \end{pmatrix}\begin{pmatrix} X \\ C \end{pmatrix}\right] = 0,$$

$$\left[(p, w)\begin{pmatrix} A-rI & I \\ e & 0 \end{pmatrix} - \begin{pmatrix} 0 \\ \nabla u(C) \end{pmatrix}\right]\begin{pmatrix} X \\ C \end{pmatrix} = 0.$$

Note 1. By assumption on u and Lemma 1(b), $p \geqq \nabla u(C) > 0$, hence Lemma 1(c) implies $(A-I)X + C = 0$.

LEMMA 2. *Let* (X, C) *be an* OSP; *then* $x+y=1$. *Here* $X=(x, y)$.

Proof. If $x+y<1$, then by Lemma 1(c), $w=0$. Lemma 1(b) implies $p(rI-A) \leqq 0$ and $p \geqq \nabla u(C) > 0$. The assumption implies that $(rI-A)^{-1} \geqq 0$, hence $p = p(rI-A)(rI-A)^{-1} \leqq 0$, which is a contradiction.

3. Main result. We will first find necessary and sufficient conditions for both types of OSPs. From Lemma 2, we see that for a boundary OSP to exist, it has to be one of the solutions to the following two systems of equations:

$$\begin{aligned} &(A-I)X + C = 0, && (A-I)X + C = 0, \\ &x+y=1, && x+y=1, \\ &c_2 = 0, && c_1 = 0. \end{aligned}$$

Their respective solutions can be computed as follows:

$$(X^1, C^1) = ((1-d)/(1+c-d),\ c/(1+c-d),\ \det(A-I)/(1+c-d),\ 0),$$

$$(X^2, C^2) = (b/(1+b-a),\ (1-a)/(1+b-a),\ 0,\ \det(A-I)/(1+b-a)).$$

THEOREM 1. (a) (X^1, C^1) *is an* OSP *iff* $u_1(C^1)/u_2(C^1) \geqq (r+c-d)/(r+b-a)$.
(b) (X^2, C^2) *is an* OSP *iff* $u_1(C^2)/u_2(C^2) \leqq (r+c-d)/(r+b-a)$.

Proof. By assumption, $b, c > 0$. Hence $X^1 > 0$ and $c_1^1 > 0$. By Lemma 1(c), $(p^1, w^1) = u_1(C^1)(1, (r+b-a)/(r+c-d), \det(rI-A)/(r+c-d))$. For (p^1, w^1) to satisfy Lemma 1(a)(b), it is necessary and sufficient that $u_1(C^1)/u_2(C^1) \geqq (r+c-d)/(r+b-a)$. This proves (a). Similarly, (b) can also be shown.

For the interior OSP, we have the following theorem.

THEOREM 2. *A stationary program* (X, C) *is an interior* OSP *iff there exists* $g \in (0, 1)$ *such that* (a) $(X, C) = (1-g)(X^1, C^1) + g(X^2, C^2)$, (b) $u_1(C)/u_2(C) = (r+c-d)/(r+b-a)$.

Proof. Sufficiency can be easily proved, so it will not be shown here. Let (X, C) be any interior OSP. By Lemma 2 and Note 1, c_1 can be expressed as a decreasing function of c_2. Since $c_1 < c_1^1$, it implies that there exists $g \in (0, 1)$ such that $c_1 = (1-g)c_1^1$. By note 1, (a) can now be shown. (b) follows immediately from Lemma 1(c).

Using Theorems 1 and 2, we are now ready to prove our main result.

THEOREM 3. *Except in degenerate cases, the number of* OSPs *is odd.*

Proof. Let $C = (1-g)C^1 + gC^2$; then $u_1(C)/u_2(C)$ is a function of g for $0 \leqq g \leqq 1$. We rewrite the function as $F(g)$. We now divide the problem into the following two cases:

Case 1. Only one of (X^1, C^1) and (X^2, C^2) is an OSP.

By Theorem 1, both endpoints of the graph of $F(g)$ are either above or below the horizontal line $y = (r+c-d)/(r+b-a)$ at the same time. Hence, except in degenerate cases, the graphs of $F(g)$ can cross the lines an even number of times. By Theorem

2, this means there are an even number of interior OSPs. Hence the total number of OSPs is odd.

Case 2. Both (X^1, C^1) and (X^2, C^2) are OSPs or both are not.

This implies one of the endpoints of $F(g)$ is above the line $y = (r+c-d)/(r+b-a)$ and the other endpoint is below the line. Hence, except in degenerate cases, the graph of $F(g)$ would cross the line an odd number of times. This proves the theorem.

For some special cases, the number of OSPs is unique.

COROLLARY. *If* $u_{12}(C) \geqq 0$ *for all* $C \geqq 0$, *then there exists a unique* OSP. *In particular, if u is separable, it has a unique* OSP.

Proof. It is easy to show that u_1/u_2 is an increasing function of g. The corollary follows immediately.

Acknowledgment. I wish to express my deepest appreciation to Professor David Gale for his advice and encouragement of this work.

REFERENCES

[1] C. CHUNG, *Stability analysis in a Leontief optimal growth model*, OR Center report 81-13, Univ. California, Berkeley, 1981.

[2] G. DANTZIG AND A. MANNE, *A complementarity algorithm for an optimal capital path with invariant proportions*, J. Econ. Theory, 9 (1974), pp. 312-323.

[3] T. HANSEN AND T. KOOPMANS, *On the definition and computation of a capital stock invariant under optimization*, J. Econ. Theory, 5 (1974), pp. 487-523.

[4] P. JONES, *Computing an invariant capital stock*, this Journal, 3 (1982), pp. 145-150.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 2, April 1985

008

ERROR BOUNDS FOR THE SSOR SEMI-ITERATIVE METHOD*

LALA B. KRISHNA†

Abstract. The SSOR semi-iterative method is applied to the system of linear equations $A\mathbf{x}=\mathbf{b}$, where A is an $n\times n$ Hermitian positive definite matrix. We find the bounds for the A-norm of the error vector in terms of the spectral radius of the SSOR iteration matrix.

AMS(MOS) subject classifications. 65F10, 65F15
CR categories. 3.15, 5.14

1. Introduction.

To solve the system of n linear equations

$$A\mathbf{x}=\mathbf{b}, \tag{1.1}$$

where $A\in\mathbb{C}^{n,n}$ is a nonsingular complex matrix, we consider the splitting of A,

$$A=D-L-U, \tag{1.2}$$

where D, $-L$ and $-U$ denote the diagonal, strictly lower and strictly upper triangular parts of A.

The Symmetric Successive Overrelaxation (SSOR) iterative method [6, p. 461] is defined by

$$\mathbf{X}^{(k+1)}=\mathcal{S}_\omega\mathbf{x}^{(k)}+\omega(2-\omega)(D-\omega U)^{-1}D(D-\omega L)^{-1}\mathbf{b}, \qquad k=0,1,\cdots, \tag{1.3}$$

where

$$\mathcal{S}_\omega := (D-\omega U)^{-1}\{(1-\omega)D+\omega L\}(D-\omega L)^{-1}\{(1-\omega)D+\omega U\}$$

is the SSOR iteration matrix associated with the matrix A.

If G is an iteration matrix, then it is known that the associated iterative method converges for any choice of initial vector if and only if the spectral radius, $\rho(G)$, of G is less than unity.

The semi-iterative method was first introduced by Varga [4] in 1957.

Consider the iterative procedure

$$\mathbf{x}^{(k+1)}=G\mathbf{x}^{(k)}+\mathbf{g}, \qquad k=0,1,2,\cdots, \tag{1.4}$$

where G is a fixed $n\times n$ iteration matrix corresponding to the system (1.1). The error vector of the kth iterate is

$$\mathbf{E}^{(k)}:=\mathbf{x}^{(k)}-\mathbf{x}, \qquad k=0,1,2,\cdots, \tag{1.5}$$

where $\mathbf{x}$ is the unique solution of (1.1). Given the sequence $\alpha_{k,i}$ satisfying

$$\sum_{i=0}^{k}\alpha_{k,i}=1, \qquad k=0,1,2,\cdots, \tag{1.6}$$

we define a sequence of vectors

$$\mathbf{v}^{(k)}=\sum_{i=0}^{k}\alpha_{k,i}\mathbf{x}^{(i)}.$$

If

$$\boldsymbol{\eta}^{(k)}=\mathbf{v}^{(k)}-\mathbf{x}, \qquad k=0,1,2,\cdots, \tag{1.7}$$

* Received by the editors June 30, 1983, and in revised form February 10, 1984.
† Department of Mathematical Sciences, University of Akron, Akron, Ohio 44325.

then

$$\eta^{(k)} = \sum_{i=0}^{k} \alpha_{k,i}\mathbf{E}^{(i)} = \left(\sum_{i=0}^{k} \alpha_{k,i}G^i\right)\mathbf{E}^{(0)}, \tag{1.8}$$

and in particular $\eta^{(0)} = \mathbf{E}^{(0)}$.

Define

$$p_k(x) = \sum_{i=0}^{k} \alpha_{k,i}x^i. \tag{1.9}$$

Then from (1.8),

$$\eta^{(k)} = p_k(G)\mathbf{E}^{(0)}. \tag{1.10}$$

Suppose G has all real eigenvalues λ and lies in the range

$$\alpha \leqq \lambda \leqq \beta < 1, \quad \text{where} \quad \beta > \alpha. \tag{1.11}$$

If

$$\gamma(\lambda) = \frac{2\lambda - (\alpha+\beta)}{\beta - \alpha}, \tag{1.12}$$

then $\gamma(\alpha) = -1$ and $\gamma(\beta) = 1$.

Moreover,

$$z := \gamma(1) = \frac{2-(\alpha+\beta)}{\beta-\alpha} > 1. \tag{1.13}$$

Let

$$Q_k(\gamma) = p_k\left(\frac{(\beta-\alpha)\gamma + \beta + \alpha}{2}\right). \tag{1.14}$$

Then

$$p_k(\lambda) = Q_k\left(\frac{2\lambda - (\alpha+\beta)}{\beta-\alpha}\right) = Q_k(\gamma), \tag{1.15}$$

and

$$\max_{\alpha \leqq \lambda \leqq \beta} |p_k(\lambda)| = \max_{-1 \leqq \gamma \leqq 1} |Q_k(\gamma)|. \tag{1.16}$$

By Young [6, p. 302, Thm. 3.1] and Varga [5], since $Q_k(z) = p_k(1) = 1$ and $z > 1$, the polynomial $Q_k(\gamma)$ which minimizes the right-hand side of (1.16) is given by

$$Q_k(\gamma) = \frac{T_k(\gamma)}{T_k(z)},$$

where $T_k(z)$ is the Chebyshev polynomial of degree k in z. An easy calculation [6] shows that

$$\eta^{(k+1)} = \rho_{k+1}(G)\mathbf{E}^{(0)} = \left[\frac{T_{k+1}(2G - (\beta+\alpha)I)/(\beta-\alpha)}{T_{k+1}(z)}\right]\mathbf{E}^{(0)}. \tag{1.17}$$

Using a three-term recurrence formula for Chebyshev polynomials, we have for $k \geqq 1$,

$$\eta^{(k+1)} = 2\left[\frac{2G - (\beta+\alpha)I}{\beta-\alpha}\right]\frac{T_k(z)}{T_{k+1}(z)}\eta^{(k)} - \frac{T_{k-1}(z)}{T_{k+1}(z)}\eta^{(k-1)}. \tag{1.18}$$

Therefore, from (1.7),

$$\mathbf{v}^{(k+1)} = 2\left[\frac{2}{\beta-\alpha}G - \left(\frac{\beta+\alpha}{\beta-\alpha}\right)I\right]\frac{T_k(z)}{T_{k+1}(z)}\mathbf{v}^{(k)} - \frac{T_{k-1}(z)}{T_{k+1}(z)}\mathbf{v}^{(k-1)} + \frac{4}{\beta-\alpha}\frac{T_k(z)}{T_{k+1}(z)}\mathbf{g}. \tag{1.19}$$

On simplification, we get

$$\mathbf{v}^{(k+1)} = \frac{l_{k+1}}{2-(\alpha+\beta)}\{(2G-(\beta+\alpha)I)\mathbf{v}^{(k)} + 2\mathbf{g}\} + (1-l_{k+1})\mathbf{v}^{(k-1)}, \tag{1.20}$$

where

$$l_1 = 1, \quad l_2 = \frac{2z^2}{2z^2-1}, \quad l_{k+1} = \left(1 - \frac{1}{4z^2}l_k\right)^{-1}, \qquad k = 2, 3, \cdots. \tag{1.21}$$

So the SSOR semi-iterative method applied to (1.1) will be given by

$$\mathbf{v}^{(k+1)} = \frac{l_{k+1}}{2-(\alpha+\beta)}\{(2\mathcal{S}_\omega - (\beta+\alpha)I)\mathbf{v}^{(k)} + 2\mathbf{g}\} + (1-l_{k+1})\mathbf{v}^{(k-1)}, \tag{1.22}$$

$$k = 0, 1, 2, \cdots,$$

where $\mathbf{g} = \omega(2-\omega)(D-\omega U)^{-1}D(D-\omega L)^{-1}\mathbf{b}$ and $\mathcal{S}_\omega$ is the SSOR iteration matrix associated with matrix A.

DEFINITION 1.1. If A is a Hermitian positive definite matrix, then the A-norm of a column vector $\mathbf{x}$ is defined by

$$\|\mathbf{x}\|_A := (\mathbf{x}^*A\mathbf{x})^{1/2}.$$

DEFINITION 1.2. For any matrix $B = [b_{i,j}] \in R^{n,n}$ with $b_{i,j} \leqq 0$, $i \neq j$, $1 \leqq i, j \leqq n$, we define a matrix $C = [c_{i,j}] \in R^{n,n}$ such that $B = \tau I - C$, where $\tau = \max_{1\leqq i\leqq n}\{b_{i,i}\}$ and $c_{i,i} = \tau - b_{i,i} \geqq 0$, $1 \leqq i \leqq n$, $c_{i,j} = -b_{i,j} \geqq 0$, $i \neq j$, $1 \leqq i, j \leqq n$.

The matrix B defined above is called a nonsingular M-matrix if $\tau > \rho(C)$, where $\rho(C)$ is the spectral radius of the matrix C. This definition was given by Ostrowski [3]. We remark that a nonsingular symmetric M-matrix is also positive definite.

2. Main results. In the following theorem, we give the error bounds for the A-norm of the error vector at the kth iteration of the SSOR semi-iterative method in terms of the spectral radius of the SSOR iteration matrix $\mathcal{S}_\omega$.

THEOREM 1. *For the system* (1.1), *let*:

(i) *A be an $n \times n$ Hermitian positive definite matrix and ω be any real number in* $(0, 2)$;

(ii) *$\boldsymbol{\eta}^{(k)}$ be the error vector at the kth iteration of the* SSOR *semi-iterative method defined by* (1.8);

(iii) *λ be the spectral radius of the $\mathcal{S}_\omega$, i.e., $\lambda = \rho(\mathcal{S}_\omega)$.*

Then

$$\frac{\|\boldsymbol{\eta}^{(k)}\|_A}{\|\boldsymbol{\eta}^{(0)}\|_A} \leqq \frac{2\lambda^k}{(1+\sqrt{1-\lambda})^{2k} + (1-\sqrt{1-\lambda})^{2k}}, \qquad \boldsymbol{\eta}^{(0)} \neq \mathbf{0}. \tag{2.1}$$

Proof. Define $\tilde{L} := D^{-1}L$, $\tilde{U} := D^{-1}L^*$, where $U = L^*$. Then from (1.3), it follows that

$$\mathcal{S}_\omega = (I-\omega\tilde{U})^{-1}((1-\omega)I + \omega\tilde{L})(I-\omega\tilde{L})^{-1}((1-\omega)I + \omega\tilde{U}). \tag{2.2}$$

Since $((1-\omega)I + \omega\tilde{L})(I-\omega\tilde{L})^{-1} = (I-\omega\tilde{L})^{-1}((1-\omega)I + \omega\tilde{L})$, then

$$\mathcal{S}_\omega = (I-\omega\tilde{U})^{-1}(I-\omega\tilde{L})^{-1}((1-\omega)I + \omega\tilde{L})((1-\omega)I + \omega\tilde{U}). \tag{2.3}$$

Set

$$M_\omega := \frac{D}{\omega(2-\omega)}(I-\omega\tilde{L})(I-\omega\tilde{U}),$$
$$N_\omega := \frac{D}{\omega(2-\omega)}((1-\omega)I+\omega\tilde{L})((1-\omega)I+\omega\tilde{U}). \tag{2.4}$$

So,

$$A = M_\omega - N_\omega \quad \text{and} \quad \mathcal{S}_\omega = M_\omega^{-1}N_\omega. \tag{2.5}$$

Let

$$P := \frac{1}{\sqrt{\omega(2-\omega)}}(D-\omega L)D^{-1/2}. \tag{2.6}$$

Then

$$M_\omega = PP^*. \tag{2.7}$$

Moreover, $\mathcal{S}_\omega = I - M_\omega^{-1}A$ and $(2/\lambda)\mathcal{S}_\omega - I = (2/\lambda - 1)I - (2/\lambda)M_\omega^{-1}A$. Set $t := T_k(z)$, the Chebyshev polynomial of degree k in z. For $0<\omega<2$, all eigenvalues λ_i of $\mathcal{S}_\omega$ lie in the range $0 \leqq \lambda_i \leqq \lambda = \rho(\mathcal{S}_\omega)$, $1 \leqq i \leqq n$. From (1.17),

$$\boldsymbol{\eta}^{(k)} = \frac{T_k((2\mathcal{S}_\omega - \lambda I)/\lambda)}{T_k(z)}\boldsymbol{\eta}^{(0)} = \frac{T_k(((2/\lambda)-1)I-(2/\lambda)M_\omega^{-1}A)}{t}\boldsymbol{\eta}^{(0)}, \tag{2.8}$$

where

$$z = \frac{2}{\lambda} - 1. \tag{2.9}$$

Multiplying both sides of (2.8) by P^*, it follows that

$$P^*\boldsymbol{\eta}^{(k)} = \frac{1}{t}\left[\left(\frac{2}{\lambda}-1\right)I - \frac{2}{\lambda}P^{-1}AP^{-*}\right]P^*\boldsymbol{\eta}^{(0)}. \tag{2.10}$$

Set $\tilde{A} := P^{-1}AP^{-*}$. Since $\tilde{A}$ is Hermitian, we can write $\tilde{A} = \mathcal{V}\Lambda\mathcal{V}^*$, where $\mathcal{V}^*\mathcal{V} = \mathcal{V}\mathcal{V}^* = I$ and

$$\Lambda = \begin{bmatrix} \mu_1 & & 0 \\ & \ddots & \\ 0 & & \mu_n \end{bmatrix},$$

with $\mu_1, \mu_1, \cdots, \mu_n$ eigenvalues of $\tilde{A}$. Note that eigenvalues of $\tilde{A}$ are the same as those of $M_\omega^{-1}A$. Moreover,

$$\|V^*\tilde{A}^{1/2}P^*\boldsymbol{\eta}^{(k)}\|_2 = \|\tilde{A}^{1/2}P^*\boldsymbol{\eta}^{(k)}\|_A = \|\boldsymbol{\eta}^{(k)}\|_A, \tag{2.11}$$

and multiplying both side of (2.10) by $\mathcal{V}^*\tilde{A}^{1/2}$ gives

$$\mathcal{V}^*\tilde{A}^{1/2}P^*\boldsymbol{\eta}^{(k)} = \mathcal{V}^*\tilde{A}^{1/2}\frac{1}{t}T_k\left[\left(\frac{2}{\lambda}-1\right)I - \frac{2}{\lambda}\tilde{A}\right]P^*\boldsymbol{\eta}^{(0)}$$
$$= \frac{1}{t}T_k\left[\left(\frac{2}{\lambda}-1\right)I - \frac{2}{\lambda}\Lambda\right]V^*\tilde{A}^{1/2}P^*\boldsymbol{\eta}^{(0)}.$$

If we set

$$\tilde{\boldsymbol{\eta}}^{(k)} := \mathcal{V}^* \tilde{A}^{1/2} P^* \boldsymbol{\eta}^{(k)} = \Lambda^{1/2} \mathcal{V}^* P^* \boldsymbol{\eta}^{(k)}, \qquad k = 0, 1, 2, \cdots, \tag{2.12}$$

then

$$\tilde{\boldsymbol{\eta}}^{(k)} = \frac{1}{t} T_k \left[\left(\frac{2}{\lambda} - 1 \right) I - \frac{2}{\lambda} \Lambda \right] \tilde{\boldsymbol{\eta}}^{(0)}, \tag{2.13}$$

and

$$\|\boldsymbol{\eta}^{(k)}\|_A^2 = \|\mathcal{V}^* \tilde{A}^{1/2} P^* \boldsymbol{\eta}^{(k)}\|_2^2 = \|\tilde{\boldsymbol{\eta}}^{(k)}\|_2^2 = \frac{1}{t^2} \sum_{i=1}^{n} \left[T_k \left(\left(\frac{2}{\lambda} - 1 \right) - \frac{2}{\lambda} \mu_i \right) \right]^2 \tilde{\eta}_i^{(0)^2}, \tag{2.14}$$

where $\tilde{\eta}_i^{(0)}$ is the ith component of the vector $\tilde{\boldsymbol{\eta}}^{(0)}$ defined in (2.12). The eigenvalues λ_i of $\mathcal{S}_\omega$ and μ_i are related by

$$\lambda_i = 1 - \mu_i, \qquad i = 1, 2, \cdots, n. \tag{2.15}$$

If $y_i = (2\lambda_i/\lambda) - 1$, then $-1 \leqq y_i \leqq 1$, $i = 1, 2, \cdots, n$. So from (2.14),

$$\|\boldsymbol{\eta}^{(k)}\|_A^2 = \frac{1}{t^2} \sum_{i=1}^{n} [T_k(y_i)]^2 \tilde{\eta}_i^{(0)^2} \leqq \frac{1}{t^2} \max_{i=1,2,\cdots,n} [T_k(y_i)]^2 \sum_{i=1}^{n} \tilde{\eta}_i^{(0)^2}.$$

Or,

$$\frac{\|\boldsymbol{\eta}^{(k)}\|_A^2}{\|\boldsymbol{\eta}^{(0)}\|_A^2} \leqq \frac{1}{t^2} \max_{i=1,2,\cdots,n} [T_k(y_i)]^2, \qquad \boldsymbol{\eta}^{(0)} \neq \mathbf{0}.$$

Since $\max_{-1 \leqq y \leqq 1} |T_k(y)| = 1$,

$$\frac{\|\boldsymbol{\eta}^{(k)}\|_A}{\|\boldsymbol{\eta}^{(0)}\|_A} \leqq \frac{1}{t}, \qquad \boldsymbol{\eta}^{(0)} \neq \mathbf{0}. \tag{2.16}$$

So from (2.8),

$$\frac{\|\boldsymbol{\eta}^{(k)}\|_A}{\|\boldsymbol{\eta}^{(0)}\|_A} \leqq \frac{1}{T_k(2/\lambda - 1)}, \qquad \boldsymbol{\eta}^{(0)} \neq 0.$$

But it follows [6, p. 302] that

$$T_k\left(\frac{2}{\lambda} - 1 \right) = \frac{1}{2} \left[\left(\frac{2}{\lambda} - 1 + \frac{2}{\lambda}(1-\lambda)^{1/2} \right)^k + \left(\frac{2}{\lambda} - 1 - \frac{2}{\lambda}(1-\lambda)^{1/2} \right)^k \right].$$

Hence, from (2.16),

$$\begin{aligned} \frac{\|\boldsymbol{\eta}^{(k)}\|_A}{\|\boldsymbol{\eta}^{(0)}\|_A} &\leqq \frac{2}{(2/\lambda - 1 + (2/\lambda)\sqrt{1-\lambda})^k + (2/\lambda - 1 - (1/\lambda)\sqrt{1-\lambda})^k} \\ &= \frac{2\lambda^k}{(1+\sqrt{1-\lambda})^{2k} + (1-\sqrt{1-\lambda})^{2k}}, \qquad \boldsymbol{\eta}^{(0)} \neq \mathbf{0}. \end{aligned}$$

□

Now we state the following theorem when A is an $n \times n$ nonsingular symmetric M-matrix; the proof is given in [2]. We remark that a nonsingular symmetric M-matrix is also positive definite.

THEOREM 2. *Let A be an $n \times n$ nonsingular symmetric M-matrix and $\omega_1 = 2/(1+\sqrt{1-\mu^2})$, where $\mu = \rho(B)$ and $B = I - (\operatorname{diag} A)^{-1} A$. Then $(\omega_1 - 1)^2 \leqq \rho(\mathcal{S}_{\omega_1}) \leqq (\omega_1 - 1)$.*

As a consequence of Theorem 1 and Theorem 2, we can state the following corollary.

COROLLARY 3. *Let A be an $n \times n$ nonsingular symmetric M-matrix and $\omega_1 = 2/(1+\sqrt{1-\mu^2})$, where $\mu = \rho(B)$ and $B = I - (\operatorname{diag} A)^{-1}A$. Then the A-norm of the error vector $\eta^{(k)}$ after the kth iteration of the* SSOR *semi-iterative method is bounded by*

$$\frac{\|\eta^{(k)}\|_A}{\|\eta^{(0)}\|_A} \leqq \frac{2(\omega_1 - 1)^k}{(1+\sqrt{2-\omega_1})^{2k} + (1-\sqrt{2-\omega_1})^{2k}}, \qquad \eta^{(0)} \neq \mathbf{0}.$$

3. Conclusions. Let $\lambda = \rho(\mathcal{S}_\omega)$ be the spectral radius of the SSOR iteration matrix $\mathcal{S}_\omega$ associated with a Hermitian positive definite matrix A for a given ω in $(0, 2)$. If we apply the SSOR semi-iterative method to solve (1.1), then the process can be terminated after K iterations, where K satisfies

$$\frac{2\lambda^K}{(1+\sqrt{1-\lambda})^{2K} + (1-\sqrt{1-\lambda})^{2K}} \leqq 10^{-s}$$

for some preassigned positive integer s.

Young [7] has shown that the SSOR semi-iterative method offers a substantial reduction in the number of iterations required, as compared with the SOR iterative method, for many problems, in particular for the general class of elliptic boundary value problems.

However, Alefeld [1] has shown that if A is an M-matrix of the form

$$A = \begin{bmatrix} D_1 & H \\ K & D_2 \end{bmatrix}, \tag{3.1}$$

where D_1 and D_2 are diagonal matrices, then

$$\min_{0<\omega<2} \rho(\mathcal{S}_\omega) = \rho(\mathcal{S}_1).$$

In the case (3.1), the Gauss-Seidel semi-iterative method [6] should be used to solve (1.1). The same is true if the matrix A is an Hermitian positive definite matrix and has the form (3.1) [6, p. 464, Thm. 2.2].

REFERENCES

[1] G. ALEFELD, *On the convergence of the symmetric* SOR *method for matrices with red-black ordering*, Numer. Math., 39 (1982), pp. 113-117.

[2] L. B. KRISHNA, *On the convergence of the symmetric successive overrelaxation method*, Linear Algebra Appl., 56 (1984), pp. 185-194.

[3] A. M. OSTROWSKI, *Uber die determinanten mit uberwiegender hauptdiagonale*, Comment. Math. Helv., 10 (1937), pp. 69-96.

[4] R. S. VARGA, *A comparison of the successive overrelaxation method and semi-iterative methods using Chebyshev polynomials*, J. Soc. Indus. Appl. Math., 5 (1957), pp. 39-46.

[5] ———, *Matrix Iterative Analysis*, Prentice-Hall, Englewood Cliffs, NJ, 1962.

[6] D. YOUNG, *Iterative Solution of Large Linear Systems*, Academic Press, New York, 1971.

[7] ———, *On the accelerated* SSOR *method for solving large linear systems*, Adv. Math., 23 (1977), pp. 215-271.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 2, April 1985

RANDOMLY NEAR-TRACEABLE GRAPHS*

JOHN FREDERICK FINK†

Abstract. A walk generated by a (not necessarily completed) depth-first search of a graph is called a DFS *walk*. A connected graph is *randomly near-traceable* if it admits no DFS walk $W: w_1, w_2, \cdots, w_n$ having consecutive vertices w_k and w_{k+1} that both appear on the subwalk $w_1, w_2, \cdots, w_{k-1}$; thus, in a depth-first search of a randomly near-traceable graph, whenever we backtrack to a previously visited vertex, that vertex is adjacent to at least one unvisited vertex. We characterize the bipartite randomly near-traceable graphs and show that for every randomly near-traceable graph G that is not a cycle, the radius of G is at most 2. Other results are also presented.

AMS(MOS) subject classification. 05C

1. Introduction.

In [2] Chartrand and Kronk defined a graph G to be *randomly traceable* if for every vertex v of G every path beginning at v can be extended to a Hamiltonian path beginning at v. They characterized randomly traceable graphs as follows:

THEOREM A. *A graph G of order p is randomly traceable if and only if G is the cycle C_p, the complete graph K_p, or the regular complete bipartite graph $K(p/2, p/2)$, the last being possible if and only if p is even.*

In this paper we will define and investigate a related class of graphs that we refer to as randomly near-traceable graphs. The development of this topic will be based on the concept of a depth-first search of a connected graph. We will see that the definition of a randomly traceable graph can also be stated in terms of the depth-first search procedure and, as a consequence, that every randomly traceable graph is randomly near-traceable. (All terms not defined herein are as defined in [1].)

Since the depth-first search procedure for a connected graph can be formulated in several ways, it is convenient for us to describe what we mean by a depth-first search of a connected graph. A depth-first search of a connected graph G is a step-by-step method for generating a walk that visits (i.e., includes) each vertex of G. At a given step in a depth-first search of G, the vertex which is currently being visited is designated the *active vertex*.

To begin a depth-first search of a connected graph G, we randomly select a first vertex to visit—this is the first active vertex and the first vertex of our walk. Next we select, at random, a vertex adjacent to our first active vertex and visit it; this becomes the new active vertex and the second vertex in our walk. In general, if v_a denotes the current active vertex in our search, and if the walk generated so far is not a spanning walk, we proceed as follows. If there are unvisited vertices adjacent to v_a, select one at random, visit it, designate if the new active vertex, and append it to our walk. If each vertex adjacent to v_a has been visited, we backtrack to (i.e., revisit) the vertex that was the active vertex immediately before v_a was first visited, designate this the current active vertex and add it to our walk. We repeat the foregoing general procedure (using the new active vertex) until each vertex of G has been visited. As soon as each vertex has been visited, the depth-first search terminates.

A walk generated by a (not necessarily completed) depth-first search of a graph is called a *depth-first search walk*, or, more briefly, a *DFS walk*. We see now that a graph G is randomly traceable if and only if G is connected and every DFS walk in

* Received by the editors September 1, 1983, and in revised form January 16, 1984.
† Department of Mathematics, University of Louisville, Louisville, Kentucky 40292.

G is a path. Thus, every completed depth-first search of a randomly traceable graph yields a Hamiltonian path in G. Reformulated, this means that a connected graph G is randomly traceable if and only if every depth-first search of G is completed without backtracking (i.e., revisiting a vertex).

If no depth-first search walk $W: w_1, w_2, \cdots, w_n$ in a connected graph G contains consecutive vertices w_k and w_{k+1} both of which appear on the subwalk $w_1, w_2, \cdots, w_{k-1}$ of W, then G is said to be *randomly near-traceable.* Thus, in a depth-first search of a randomly near-traceable graph, whenever we backtrack to a previously visited vertex, that vertex is adjacent to at least one unvisited vertex. To illustrate this concept, we will demonstrate that, of the two graphs in Fig. 1, only G_1 is randomly near-traceable.

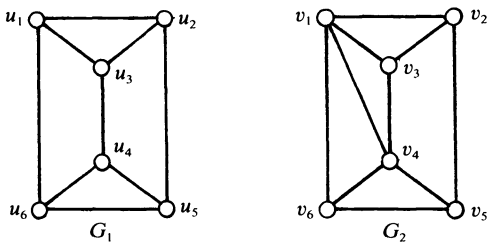


FIG. 1.

To show that G_1 is randomly near-traceable it suffices, by symmetry, to examine only those DFS walks which begin at u_1 and proceed next to either u_2 or u_6. It is easily seen that every DFS walk which begins as u_1, u_6 is a Hamiltonian path. Hence, we now consider only those DFS walks that begin as u_1, u_2. The two DFS walks which begin as u_1, u_2, u_3 are Hamiltonian paths. Thus, we now consider those DFS walks which begin as u_1, u_2, u_5; these are

$$W_1: u_1, u_2, u_5, u_6, u_4, u_3,$$

$$W_2: u_1, u_2, u_5, u_4, u_3, u_4, u_6,$$

$$W_3: u_1, u_2, u_5, u_4, u_6, u_4, u_3.$$

Among these walks, backtracking occurs only in W_2 and W_3. In each of these two walks, we backtrack to u_4 and find an unvisited vertex (u_6 or u_3 respectively) adjacent to it and continue our depth-first search by visiting that vertex. We conclude that G_1 is randomly near-traceable.

To see that G_2 is not randomly near-traceable, we consider the nonspanning DFS walk

$$v_1, v_4, v_5, v_2, v_3$$

in G_2. Since each vertex adjacent to v_3 is on this walk, it is necessary now to backtrack to v_2. However, it is now the case that every vertex adjacent to the previously visited vertex v_2 has already been visited. Thus, G_2 is not randomly near-traceable.

By definition, whether or not a given graph is randomly near-traceable depends on the structural characteristics of each depth-first search of the graph. As we shall see from the next lemma, a depth-first search walk in a randomly near-traceable graph has a very explicit structure.

LEMMA 1. *A graph G is randomly near-traceable if and only if every spanning depth-first search walk $W: w_1, w_2, \cdots, w_n$ in G is either a Hamiltonian path or satisfies the condition that for some integer k, with $1 \leqq k \leqq n-3$, the subwalk $w_1, w_2, \cdots, w_{k+1}$ is*

a path, $w_k = w_{k+2} = w_{k+4} = \cdots = w_{n-1}$, *and* $\{w_{k+1}, w_{k+3}, w_{k+5}, \cdots, w_n\}$ *is an independent set of vertices.*

Proof. Suppose that each depth-first search walk in G is either a Hamiltonian path or satisfies the condition stated above. Then, on no DFS walk W: $w_1, w_2, \cdots, w_n$ do there appear consecutive vertices w_l and w_{l+1} that are both visited on the subwalk $w_1, w_2, \cdots, w_{l-1}$. Hence, by definition, G is randomly near-traceable.

For the converse, suppose that W: $w_1, w_2, \cdots, w_n$ is a DFS walk in a randomly near-traceable graph G and that W is not a Hamiltonian path. Let k be the largest integer for which P: $w_1, w_2, \cdots, w_{k+1}$ is a path. since P cannot be extended to a longer path beginning at w_1, each vertex of G that is adjacent to w_{k+1} must be on P. Also, since G is randomly near-traceable, it follows that $w_k = w_{k+2}$ and that the vertex w_{k+3} is not on P.

Note that if $i < k$, then w_i is visited only once on W. To see this, assume to the contrary that for some $i < k$, vertex w_i is revisited on W. Let l be chosen so that w_l represents the second occurrence of w_i on W. Since P is a path, and since $w_{k+2} = w_k$ and w_{k+3} is not on P, it follows that $l > k+3$. Since G is randomly near-traceable, it follows from the definition that the vertex labelled w_{l-1} does not occur on $w_1, w_2, \cdots, w_{l-2}$. Thus, on the initial visit to w_{l-1} in the depth-first search corresponding to W, it was necessary to backtrack to the vertex proceeding w_{l-1} on W, namely w_{l-2}. Hence $w_{l-2} = w_l = w_i$. This, however, is a contradiction since $i < l-2 < l$ implies that w_l does not represent the second occurrence of w_i on W.

Now, since

$$w_1, w_2, \cdots, w_k, w_{k+3}, w_{k+4}, \cdots, w_{n-1}, w_n$$

is a DFS walk containing every vertex of G except w_{k+1} and since w_{k+1} is not adjacent to w_n, it follows that w_{k+1} must be adjacent to w_{n-1}. Since w_{k+1} is adjacent only to vertices on P and since none of the vertices $w_1, w_2, \cdots, w_{k-1}$ is revisited on W, we see that $w_{n-1} = w_k$.

Since G is randomly near-traceable and since w_{n-1} does not represent the first occurrence of w_k on W, it follows that w_{n-2} does not appear on the subwalk $w_1, w_2, \cdots, w_{n-3}$. Thus, $w_{n-3} = w_{n-1} = w_k$. By continuing to argue in the above manner, we conclude that $(n-1)-k$ is an even number and that $w_{n-1} = w_{n-3} = w_{n-5} = \cdots = w_{k+2} = w_k$.

Since $w_k = w_{k+2} = w_{k+4} = \cdots = w_{n-1}$, we see that w_{k+2j-1} is not adjacent to w_{k+2l-1} for any integers j and l where $1 \leqq j < l \leqq (n-k+1)/2$. Hence $\{w_{k+1}, w_{k+3}, w_{k+5}, \cdots, w_n\}$ is an independent set of vertices. □

If G, W and k are as in the statement of Lemma 1, then we will usually denote the DFS walk W by

$$W: w_1, w_2, \cdots, w_k \rightarrow (w_{k+1}, w_{k+3}, \cdots, w_n).$$

Usually we will label the ith newly visited vertex in W as u_i. Hence, if G has order p, then

$$W: u_1, u_2, \cdots, u_k \rightarrow (u_{k+1}, u_{k+2}, \cdots, u_p)$$

denotes the DFS walk

$$W: u_1, u_2, \cdots, u_{k-1}, u_k, u_{k+1}, u_k, u_{k+2}, u_k, \cdots, u_{p-1}, u_k, u_p.$$

The tree induced by a spanning DFS walk W in a randomly near-traceable graph G will, by Lemma 1, necessarily have one of the forms illustrated in Fig. 2.

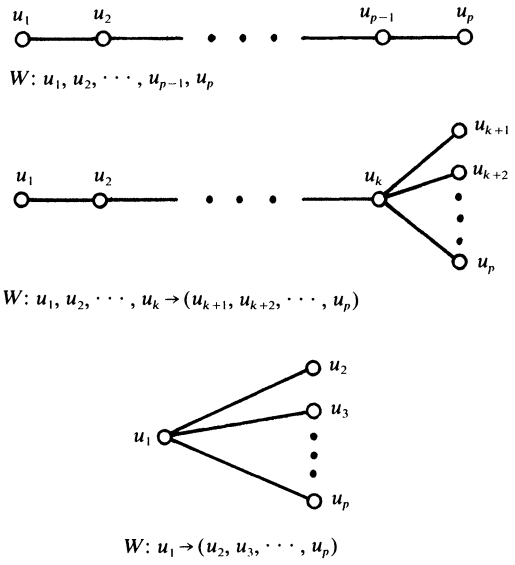


FIG. 2.

Also from Lemma 1, it follows that a DFS walk in a randomly near-traceable graph that results from an incomplete depth-first search in which at least one backtrack step has occurred has the form

$$u_1, u_2, \cdots, u_k, u_{k+1}, u_k, u_{k+2}, \cdots, u_k, u_{k+m};$$

this will be denoted by

$$u_1, u_2, \cdots, u_k \rightarrow (u_{k+1}, \cdots, u_{k+m}).$$

2. *n*-partite randomly near-traceable graphs. Since there is no backtracking in a depth-first search of a randomly traceable graph, it follows that every randomly traceable graph is randomly near-traceable. Thus, by Theorem A every cycle, complete graph and regular complete bipartite graph is randomly near-traceable. The complete graphs and regular complete bipartite graphs are therefore examples of randomly near-traceable complete n-partite graphs (for appropriate values of n). The following theorem asserts that in fact every complete n-partite graph is randomly near-traceable.

THEOREM 1. *Every complete n-partite graph is randomly near-traceable* ($n \geqq 2$).

Proof. Let $W: w_1, w_2, \cdots, w_t$ be a spanning depth-first search walk in a complete n-partite graph G, where $n \geqq 2$. Suppose that W is not a Hamiltonian path, and let k be the least integer such that the vertex w_k is visted more than once on W. Then, since W is a DFS walk, the subwalk

$$W_0: w_1, w_2, \cdots, w_k, w_{k+1}$$

is a path which cannot be extended to a longer path beginning at w_1. Thus $w_k = w_{k+2}$ and each vertex of G that is not on W_0 is not adjacent to w_{k+1}; hence each vertex not on W_0 belongs to the partite set of G that contains w_{k+1}. Since G is a complete n-partite graph, this implies that each vertex not on W_0 is adjacent to w_k and that $w_k = w_{k+2} = w_{k+4} = \cdots = w_{t-1}$, while $w_{k+1}, w_{k+3}, w_{k+5}, \cdots, w_t$ are distinct. Since W was an arbitrary DFS walk in G, we conclude that G is randomly near-traceable. □

Thus, randomly traceable graphs and complete n-partite graphs are randomly near-traceable. These are however not the *only* such graphs. For example, the graph

G_1 of Fig. 1 and the graph obtained by joining one pair of nonadjacent vertices in $K(m, m)$, where $m \geqq 3$, are examples of tripartite randomly near-traceable graphs. Thus Theorems A and 1 do not provide *all* examples of randomly near-traceable tripartite graphs. As the next theorem shows, however, they *do* include all bipartite randomly near-traceable graphs.

THEOREM 2. *A bipartite graph G is randomly near-traceable if and only if G is a cycle or a complete bipartite graph.*

Proof. Let G be a randomly near-traceable bipartite graph. Let U and V be the partite sets of G and suppose that $|U| \leqq |V|$. Furthermore, suppose that G is neither a cycle nor a regular complete bipartite graph; hence, by Theorem A, the graph G is not randomly traceable.

Since G is not randomly traceable, there is a spanning DFS walk W of G having the form

$$W: x_1, x_2, \cdots, x_k \to (x_{k+1}, x_{k+2}, \cdots, x_p),$$

where p is of the order of G. (Note that if $k = 1$, then $G \cong K(1, p-1)$; Hence we shall assume that $k \geqq 2$.) since $p - k \geqq 2$, and since G is bipartite with $|U| \leqq |V|$, we conclude that $|U| < |V|$ and that $x_{k+1}, x_{k+2}, \cdots, x_p \in V$. Moreover, it follows that $x_{k-j} \in V$ if and only if $k-j$ is odd. It is this condition that leads us to consider two cases.

Case 1. Suppose $k = 2n$. We relabel the vertices as follows: for $1 \leqq i \leqq n$ we set $v_i = x_{2i-1}$ and $u_i = x_{2i}$; and for $j = 1, 2, \cdots, p-k$ we set $v_{n+j} = x_{k+j}$. Thus W has the form

$$W: v_1, u_1, v_2, u_2, \cdots, v_n, u_n \to (v_{n+1}, v_{n+2}, \cdots, v_{n+p-k}).$$

The tree induced by W is illustrated in Fig. 3.

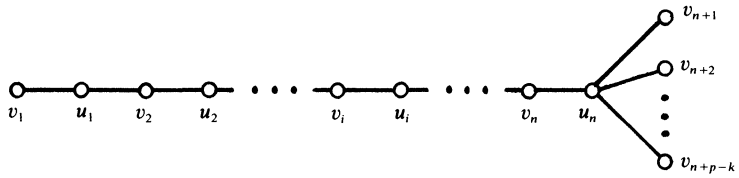


FIG. 3.

Observe now that $U = \{u_1, u_2, \cdots, u_n\}$ and $V = \{v_1, v_2, \cdots, v_{n+p-k}\}$. With this fact and the nonspanning DFS walk

$$u_1, v_2, u_2, v_3, \cdots, v_n, u_n$$

we see that $v_1 u_n$ is an edge of G. Using this edge to construct the nonspanning DFS walk

$$v_1, u_n, v_n, u_{n-1}, \cdots, v_2, u_1,$$

we see that $u_1 v_{n+j}$ is an edge of G for $1 \leqq j \leqq p-k$.

We now proceed inductively to show that for $i = 1, 2, \cdots, n-1$ the edges $u_i v_1$ and $u_i v_{n+j}$ are in G for $1 \leqq j \leqq p-k$. Assuming that $u_{i-1} v_1$ and all edges $u_{i-1} v_{n+j}$ are in G, we consider the nonspanning DFS walks

$$v_i, u_{i-1}, v_{i-1}, u_{i-2}, \cdots, v_2, u_1, v_{n+l}, u_n, v_n, u_{n-1}, v_{n-1}, \cdots, v_{i+1}, u_i, \quad \text{for } l = 1, 2$$

and see that each of the edges $u_i v_1$ and $u_i v_{n+j}$ must be in G for $1 \leqq j \leqq p-k$. Thus, for $i = 1, 2, \cdots, n-1$ the edges $u_i v_1$ and $u_i v_{n+j}$ are in G, for all j, where $1 \leqq j \leqq p-k$. In particular, v_1, the first vertex of W, is adjacent to each vertex in U. By repeating the

foregoing arguments for each spanning DFS walk of the form

$$v_j, u_j, v_{j+1}, u_{j+1}, \cdots, v_n, u_n, v_1, u_1, v_2, u_2, \cdots, v_{j-1}, u_{j-1} \to (v_{n+1}, v_{n+2}, \cdots, v_{n+p-k}),$$

where $j = 2, 3, \cdots, n$, we see that each vertex v_j is adjacent to every vertex in U. Thus G is isomorphic to the complete bipartite graph $K(n, n+p-k)$.

Case 2. Suppose $k = 2n+1$. Relabel the vertices of W as follows: for $1 \leqq i \leqq n$, set $u_i = x_{2i-1}$ and $v_i = x_{2i}$; set $u_{n+1} = x_k$ and for $j = 1, 2, \cdots, p-k$ set $v_{n+j} = x_{k+j}$. Then $U = \{u_1, u_2, \cdots, u_{n+1}\}$, $V = \{v_1, v_2, \cdots, v_{n+p-k}\}$ and W has the form

$$W: u_1, v_1, u_2, v_2, \cdots, u_n, v_n, u_{n+1} \to (v_{n+1}, v_{n+2}, \cdots, v_{n+p-k}).$$

The tree induced by W is shown in Fig. 4.

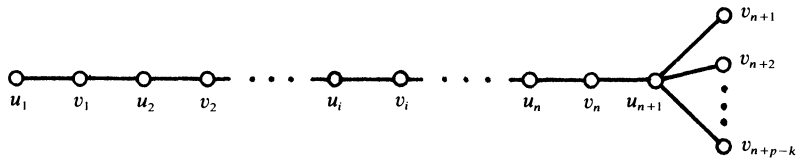


FIG. 4.

Since u_1 is not adjacent to u_{n+1} the two DFS walks

$$v_1, u_2, v_2, u_3, \cdots, v_n, u_{n+1}, v_{n+1} \quad \text{and} \quad v_1, u_2, v_2, u_3, \cdots, v_n, u_{n+1}, v_{n+2}$$

imply that $u_1 v_{n+j}$ is an edge of G for each $j = 1, 2, \cdots, p-k$. Similarly, for each $i = 2, 3, \cdots, n$, the nonspanning DFS walks

$$v_{i-1}, u_{i-1}, v_{i-2}, u_{i-2}, \cdots, v_1, u_1, v_{n+1}, u_{n+1}, v_n, u_n, \cdots, v_i, u_i$$

and

$$v_{i-1}, u_{i-1}, v_{i-2}, u_{i-2}, \cdots, v_1, u_1, v_{n+2}, u_{n+1}, v_n, u_n, \cdots, v_i, u_i$$

imply that $u_i v_{n+j}$ is an edge of G for each $j = 1, 2, \cdots, p-k$. In particular, we have, for each fixed $j = 1, 2, \cdots, p-k$, that v_{n+j} is adjacent to each vertex in U.

The nonspanning DFS walk

$$u_1, v_{n+1}, u_2, v_{n+2}, u_{n+1}, v_n, u_n, v_{n-1}, u_{n-1}, \cdots, u_3, v_2$$

implies that $v_1 u_3$ is an edge of G. In general, for $4 \leqq j \leqq n$, the nonspanning DFS walk

$$u_1, v_{n+1}, u_2, v_2, u_3, v_3, \cdots, v_{j-2}, u_{j-1}, v_{n+2}, u_{n+1}, v_n, u_n, v_{n-1}, u_{n-1}, \cdots, u_j, v_{j-1}$$

implies that $v_1 u_j$ is an edge of G. Thus, v_1 is adjacent to each vertex in U.

By applying arguments similar to the foregoing to the spanning DFS walk

$$W_i: u_i, v_i, u_{i+1}, v_{i+1}, \cdots, u_{n+1}, v_{n+1}, u_1, v_1, u_2, v_2, \cdots, u_{i-1}$$
$$\to (v_{i-1}, v_{n+2}, v_{n+3}, \cdots, v_{n+p-k}),$$

we see that v_i is adjacent to each vertex of U for each $i = 1, 2, \cdots, n$. Thus, G is isomorphic to the complete bipartite graph $K(n+1, n+p-k)$.

For the converse, we see from Theorem A that any cycle is randomly near-traceable, and from Theorem 1 that any complete bipartite graph is randomly near-traceable. □

3. Radius and diameter of randomly near-traceable graphs. If u and v are any two vertices of a connected graph G, then the *distance* from u to v, denoted $d(u, v)$, is the length of a shortest $u-v$ path in G. The *diameter* of G, denoted diam G, is the maximum distance between any two vertices in G. The *radius* of G is denoted rad G

and defined by

$$\text{rad}\,(G) = \min_{u \in V(G)} \{ \max_{v \in V(G)} d(u, v) \}.$$

Thus, if rad $G = r$, then there is at least one vertex u in G such that $d(u, v) \leqq r$ for every $v \in V(G)$.

Each randomly near-traceable graph that we have discussed so far has radius at most 2. As the next theorem shows, this is not coincidental.

THEOREM 3. *If G is a randomly near-traceable graph that is not a cycle, then* rad $G \leqq 2$.

Proof. If G is also randomly traceable, then, by Theorem A, either G is a complete graph or a regular complete bipartite graph. In either instance rad $G \leqq 2$. Thus, we henceforth assume that G is not randomly traceable.

Suppose also that G has order p and that rad $G \neq 1$; that is $\Delta(G) \leqq p-2$. Then, by Lemma 1, there is a depth-first search of G which yields a spanning walk of the form

$$W: u_1, u_2, \cdots, u_k \to (u_{k+1}, u_{k+2}, \cdots, u_p).$$

The spanning tree induced by W is indicated in Fig. 5.

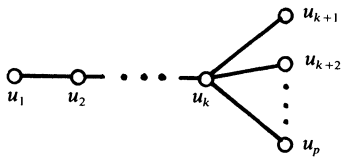


FIG. 5.

Clearly, if $2 \leqq k \leqq 4$ then $d(u_{k-1}, u_i) \leqq 2$ for $i = 1, 2, \cdots, p$, so that rad $G \leqq 2$. Suppose then that $k \leqq 5$. We shall show that $d(u_{k+i}, u_j) \leqq 2$ for all $i = 1, 2, \cdots, p-k$ and $j = 1, 2, \cdots, p$. We consider two cases.

Case 1. Assume that $u_1 u_k$ is an edge of G. Then $d(u_{k+i}, u_1) \leqq 2$ and $d(u_{k+i}, u_{k-1}) \leqq 2$ for $i = 1, 2, \cdots, p-k$. Also, if for each $j = 1, 2, \cdots, k-2$, the vertex u_{k+i} is adjacent to either u_j or u_{j+1} then $d(u_{k+i}, u_j) \leqq 2$. Assume then, to the contrary, that for some j in the range $1 \leqq j \leqq k-2$, and some $i = 1, 2, \cdots p-k$ neither $u_j u_{k+i}$ nor $u_{j+1} u_{k+i}$ is an edge of G. Now, however, the DFS walk

$$u_{j+2}, u_{j+3}, \cdots, u_k, u_1, u_2, \cdots, u_j, u_{j+1}$$

cannot be continued to include every vertex in the independent set $\{u_{k+1}, u_{k+2}, \cdots, u_p\}$ without backtracking to a vertex preceding u_j. This is a contradiction. Hence, for every $j = 1, 2, \cdots, p$ and every $i = 1, 2, \cdots, p-k$, we have $d(u_j, u_{k+i}) \leqq 2$.

Case 2. Suppose that $u_1 u_k$ is not an edge of G. Then the DFS walk

$$u_2, u_3, \cdots, u_{k-1}, u_k, u_{k+i}$$

for any i in the range $1 \leqq i \leqq p-k$, together with the fact that $\{u_{k+1}, u_{k+2}, \cdots, u_p\}$ is an independent set, implies that u_1 is adjacent to each vertex in $\{u_{k+1}, u_{k+2}, \cdots, u_p\}$. (See Fig. 6.)

Now, if $2 \leqq j \leqq k-2$, then the DFS walks

$$W_1: u_{j+2}, u_{j+3}, \cdots, u_k, u_{k+1}, u_1, u_2, \cdots, u_j, u_{j+1}$$

and

$$W_2: u_{j+2}, u_{j+3}, \cdots, u_k, u_{k+2}, u_1, u_2, \cdots, u_j, u_{j+1}$$

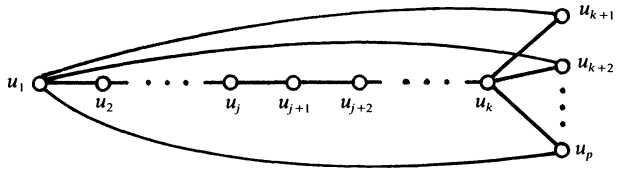


FIG. 6.

together with the fact that $\{u_{k+1}, u_{k+2}, \cdots, u_p\}$ is an independent set, imply that either u_j or u_{j+1} is adjacent to each of the vertices $u_{k+1}, u_{k+2}, \cdots$, and u_p. Thus, for each $j = 1, 2, \cdots, p$ and each $i = 1, 2, \cdots, p-k$, we have $d(u_j, u_{k+i}) \leqq 2$.

From Cases 1 and 2 we conclude that rad $G \leqq 2$. □

Since it is always the case that diam $G \leqq 2$ rad G, we have the following corollary to Theorem 3.

COROLLARY 1. *If G is a randomly near-traceable graph which is not a cycle, then* diam $G \leqq 4$.

We see that if G is a randomly near-traceable graph which is not a cycle, the distance between any pair of vertices in G is at most 4. Each example of a randomly near-traceable graph (which is not a cycle) that we have investigated herein has diameter 1 or 2. In light of this observation, we close with the following:

Conjecture. If G is a randomly near-traceable graph that is not a cycle, then diam $G \leqq 2$.

REFERENCES

[1] M. BEHZAD, G. CHARTRAND AND L. LESNIAK-FOSTER, *Graphs and Digraphs*, Wadsworth International, Belmont, CA, 1979.

[2] G. CHARTRAND AND H. V. KRONK, *Randomly traceable graphs*, SIAM J. Appl. Math., 16 (1968), pp. 696-700.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 2, April 1985

010

OPTIMAL WEIGHING DESIGNS*

CHING-SHUI CHENG†, JOSEPH C. MASARO‡ AND CHI SONG WONG‡

Abstract. A technique is developed for finding optimum designs for weighing n objects in N weighings ($N \geqq n$) on a chemical balance. Certain designs are shown to be optimal with respect to a large class of criteria (including the A- and D-criteria) for sufficiently large $N \equiv 2$ or 3 (mod 4). For small N, the result allows the elimination of a large number of competitors, and those that remain can be checked by a computer.

Key words. optimum designs, A-optimality, D-optimality, Φ_p-optimality, Hadamard maximum determinant problem

AMS(MOS) subject classifications. primary 62K05, 05B20

1. Introduction. Let N and n be positive integers with $n \leqq N$ and let $\mathscr{D}(N, n)$ denote the set of all $N \times n$ matrices $\mathbf{X} = \{x_{ij}\}$ with $x_{ij} = 1, -1$ or 0; such a matrix will be called a *weighing design matrix.* If $\mathbf{X}^*$ minimizes $\Phi(\mathbf{X}'\mathbf{X})$ over $\mathscr{D}(N, n)$ for some real-valued function Φ, then $\mathbf{X}^*$ is said to be Φ-*optimum.* The problem of characterizing such matrices $\mathbf{X}^*$ arises in the study of weighing designs and 2^n fractional factorial designs; for details see Cheng (1980) or Galil and Kiefer (1980). Another important application is to Hadamard transform optics in spectroscopy; we refer readers to the book of Harwit and Sloane (1979).

The well-known D-, A- and E-optimality criteria are obtained by taking $\Phi(\mathbf{X}'\mathbf{X})$ to be $\det(\mathbf{X}'\mathbf{X})^{-1}$, $\operatorname{tr}(\mathbf{X}'\mathbf{X})^{-1}$ or the maximum eigenvalue of $(\mathbf{X}'\mathbf{X})^{-1}$, respectively. All three criteria are functionals of the spectrum of $\mathbf{X}'\mathbf{X}$, i.e. of the eigenvalues $\mu_1, \mu_2, \cdots, \mu_n$ of $\mathbf{X}'\mathbf{X}$. A more general criterion is given by

$$\Phi_p(\mathbf{X}'\mathbf{X}) = \left(n^{-1} \sum_{i=1}^{n} \mu_i^{-p} \right)^{1/p}, \quad \text{where } p > 0.$$

Obviously A-optimality is the Φ_1-criterion. Furthermore, E-optimality is the limit of the Φ_p-criteria as $p \to \infty$, and D-optimality is equivalent to the limit of the Φ_p-criteria as $p \to 0$. Among these three, the E-criterion is the easiest to handle, and the A-criterion is perhaps the most difficult. For historical and technical reasons the D-criterion has been studied most extensively. The search for D-optimal designs is directly related to the Hadamard maximum determinant problem (see Brenner and Cummings (1972)). Furthermore, the D-criterion has the nice property that there always exists a D-optimal design matrix with ± 1 entries; see Galil and Kiefer (1980). Thus the search for D-optimal designs can be reduced to the set $\mathscr{D}'(N, n)$ of all $N \times n$ matrices (x_{ij}) with $x_{ij} = 1$ or -1 only. *This may not be true for other criteria.*

This paper is concerned with optimal designs when $N \equiv 2$ or 3 (mod 4). Readers are referred to Cheng (1980) for a discussion of results for $N \equiv 0$ or 1 (mod 4). When $N \equiv 2$ (mod 4), Payne (1974) showed that if there exists an $\mathbf{X}_2$ such that

$$\mathbf{X}_2'\mathbf{X}_2 = \begin{bmatrix} (N-2)\mathbf{I}_k + 2\mathbf{J}_k & \mathbf{0} \\ \mathbf{0} & (N-2)\mathbf{I}_{n-k} + 2\mathbf{J}_{n-k} \end{bmatrix}, \tag{1.1}$$

* Received by the editors February 8, 1983, and in final form November 29, 1983.

† Department of Statistics, University of California, Berkeley, California 94720. The work of this author was partially supported by the National Science Foundation under grant MCS-82-00909.

‡ University of Windsor, Windsor, Ontario, Canada N9B 3P4. The work of these authors was partially supported by the Natural Sciences and Engineering Research Council of Canada under grant A8518.

where $k=[n/2]$, $\mathbf{I}_k$ is the identity matrix of order k, and $\mathbf{J}_k$ is the $k\times k$ matrix of 1's, then $\mathbf{X}_2$ is D-optimal over $\mathcal{D}(N, n)$. Using Cheng's (1980) result, Jacroux, Masaro and Wong (1983) showed that the above $\mathbf{X}_2$ is also optimal over $\mathcal{D}'(N, n)$ with respect to a large class of criteria including all the Φ_p-criteria. However, except for the D-criterion, their result does not carry over to $\mathcal{D}(N, n)$; e.g., $\mathbf{X}_2$ is E-worse than the E-optimal design $\mathbf{X}$ which satisfies $\mathbf{X}'\mathbf{X}=(N-1)\mathbf{I}_n$.

Our knowledge about optimal designs for $N\equiv 3 \pmod 4$ is even more limited. So far the best result in this case is due to Galil and Kiefer (1980) who showed, improving a result of Payne (1974), that if $N\geqq 2n-5$, then a design $\mathbf{X}_3$ such that

$$\mathbf{X}_3'\mathbf{X}_3=(N+1)\mathbf{I}_n-\mathbf{J}_n \tag{1.2}$$

is D-optimal over $\mathcal{D}(N, n)$; such a design, however, is not always D-optimal when $N<2n-5$. Indeed, other than the D- and E-criteria, the problem of optimal designs in $\mathcal{D}(N, n)$ for $N\equiv 2$ or $3 \pmod 4$ is largely unexplored. Results of this kind virtually do not exist. The present paper is an attempt in this direction.

Throughout this paper, matrices satisfying (1.1) and (1.2) will be denoted by $\mathbf{X}_2$ and $\mathbf{X}_3$, respectively. It will be shown in § 2 that for sufficiently large N, if $\mathbf{X}_3$ exists, it is A-optimal over $\mathcal{D}(N, n)$. This result is extended to all Φ_p-criteria with $0\leqq p\leqq 1$ in § 3. The key result is Theorem 3.1 which shows that if $\mathbf{X}_3$ is A-optimal over $\mathcal{D}(N, n)$, then it is also optimal with respect to a large class of criteria including all the Φ_p-criteria, $0\leqq p\leqq 1$; in particular, it is D-optimal. It is also noted in § 2 that A- and D-optimal designs do not necessarily agree and that $\mathbf{X}_3$ is not always A-optimal. In the last section, the Φ_p-optimality of $\mathbf{X}_2$ is established for sufficiently large N and $0\leqq p\leqq 1$. A discussion of the existence and construction of $\mathbf{X}_2$ and $\mathbf{X}_3$ can be found in Galil and Kiefer (1980).

For convenience, we shall denote the set of all matrices of the form $\mathbf{X}'\mathbf{X}$ where $\mathbf{X}\in\mathcal{D}(N, n)$ (or $\mathcal{D}'(N, n)$) by $\mathcal{C}(N, n)$ (or $\mathcal{C}'(N, n)$, respectively).

2. *A*-optimality of $\mathbf{X}_3$. Throughout this section, we shall assume that $N\equiv 3 \pmod 4$. The following lemmas are useful for establishing the A-optimality of $\mathbf{X}_3$.

LEMMA 2.1. *Let $\mathbf{C}\in\mathcal{C}'(N, n)$ be such that $|c_{ij}|=c$ for all $i\neq j$. Then $\mathbf{C}$ is similar to $(N+c)\mathbf{I}_n-c\mathbf{J}_n$ or $(N-c)\mathbf{I}_n+c\mathbf{J}_n$.*

Proof. Let $\mathbf{C}=\mathbf{X}'\mathbf{X}\in\mathcal{C}'(N, n)$ be such that $|c_{ij}|=c$ for $i\neq j$, and let the ith column of $\mathbf{X}$ be $\mathbf{c}_i$. Define a matrix $\mathbf{Y}$ such that $\mathbf{Y}=(\mathbf{b}_1, \mathbf{b}_2, \cdots, \mathbf{b}_n)$, where $\mathbf{b}_i=\mathbf{c}_i$ if $\mathbf{c}_i$ has an even number of -1's, and $\mathbf{b}_i=-\mathbf{c}_i$ if $\mathbf{c}_i$ has an odd number of -1's. Then $\mathbf{X}'\mathbf{X}$ is similar to $\mathbf{Y}'\mathbf{Y}$ and each column of $\mathbf{Y}$ has an even number of -1's. Let d_{ij} be the (i, j)th entry of $\mathbf{Y}'\mathbf{Y}$. Then by the proof of Ehlich (1964, Lemma 3.1), $d_{ij}\equiv 3 \pmod 4$ for all i, j. Furthermore, $|d_{ij}|=|c_{ij}|=c$ for all $i\neq j$. Since c and $-c$ cannot both be congruent to 3 (mod 4), we have $d_{ij}=c$ for all $i\neq j$ or $d_{ij}=-c$ for all $i\neq j$, i.e., $\mathbf{Y}'\mathbf{Y}=(N+c)\mathbf{I}_n-c\mathbf{J}_n$ or $(N-c)\mathbf{I}_n+c\mathbf{J}_n$.

LEMMA 2.2. *Let $\mathbf{C}\in\mathcal{C}'(N, n)$. Then $\mathbf{C}$ is similar to a matrix $\mathbf{D}\in\mathcal{C}'(N, n)$ such that if $|d_{ij}|=1$, 3 or 5, then $d_{ij}=-1$, 3 or -5, respectively.*

Proof. Again by Ehlich (1964, Lemma 3.1), $\mathbf{C}$ *is similar to a matrix* $\mathbf{D}$ *with all* $d_{ij}\equiv 3 \pmod 4$. *Then* $\mathbf{D}$ *has the desired properties.*

LEMMA 2.3. *Let $\mathbf{C}\in\mathcal{C}'(N, n)$. If $\sum\sum_{i\neq j}c_{ij}^2\geqq n(n-1)N^2/(N-n+2)^2$, then $\operatorname{tr}\{(N+1)\mathbf{I}_n-\mathbf{J}_n\}^{-1}\leqq\operatorname{tr}\mathbf{C}^{-1}$, where $\operatorname{tr}\mathbf{C}^{-1}$ is defined to be $+\infty$ if $\mathbf{C}$ has no inverse.*

Proof. For any $\mathbf{C}\in\mathcal{C}'(N, n)$, we have $\operatorname{tr}\mathbf{C}=nN$ and $nN^2\leqq\operatorname{tr}\mathbf{C}^2\leqq n^2N^2$. For any B such that $nN^2\leqq B\leqq n^2N^2$, let $\mathcal{M}(B, N, n)$ be the set of all the symmetric nonnegative definite $n\times n$ matrices $\mathbf{M}$ such that $\operatorname{tr}\mathbf{M}=nN$ and $\operatorname{tr}\mathbf{M}^2=B$. Let $Z=\{(nB-n^2N^2)/(n-1)\}^{1/2}$. Then from Cheng (1978, Lemmas A2, A3, A6) $\operatorname{tr}\mathbf{C}^{-1}\geqq(n-1)/\mu+1/\lambda$, where $\mu=(nN-Z)/n$ and $\lambda=\{nN+(n-1)Z\}/n$. It suffices to show

$\operatorname{tr}\{(N+1)\mathbf{I}_n-\mathbf{J}_n\}^{-1}\leqq(n-1)/\mu+1/\lambda$, i.e., $(n-1)/(N+1)+1/(N-n+1)\leqq(n-1)/\mu+1/\lambda$. On substituting the expressions for μ and λ, this reduces to

$$(N-n+2)Z^2-n(n-2)Z-n^2N\geqq 0. \tag{2.1}$$

Since $nN/(N-n+2)$ is the positive root of the equation $(N-n+2)x^2-n(n-2)x-n^2N=0$, (2.1) holds provided $Z\geqq nN/(N-n+2)$. This is equivalent to $B-nN^2\geqq n(n-1)N^2/(N-n+2)^2$. Since for $\mathbf{C}\in\mathscr{C}'(N,n)$, $\sum\sum_{i\neq j}c_{ij}^2=\operatorname{tr}\mathbf{C}^2-nN^2$, the result follows.

Now we are ready to prove:

THEOREM 2.1. *For each n, there exists a positive integer $N_0(n)$ such that for all $N\geqq N_0(n)$, $\mathbf{X}_3$ is A-optimal in $\mathscr{D}(N,n)$.*

Proof. Let $\mathbf{C}\in\mathscr{C}'(N,n)$. If $\sum\sum_{i\neq j}c_{ij}^2=n(n-1)$, then by Lemma 2.1, $\mathbf{C}$ is similar to $(N+1)\mathbf{I}_n-\mathbf{J}_n$. Thus we may assume $\sum\sum_{i\neq j}c_{ij}^2\geqq n(n-1)+16$. It is straightforward to see that if $N\geqq(n-2)[n^2-n+16+\{n(n-1)(n^2-n+16)\}^{1/2}]/16$, then $n(n-1)+16>n(n-1)N^2/(N-n+2)^2$; therefore $\sum\sum_{i\neq j}c_{ij}^2>n(n-1)N^2/(N-n+2)^2$ and by Lemma 2.3, $\operatorname{tr}\mathbf{C}^{-1}>\operatorname{tr}(\mathbf{X}_3'\mathbf{X}_3)^{-1}$.

For $\mathbf{C}\in\mathscr{C}(N,n)\backslash\mathscr{C}'(N,n)$, we have $\operatorname{tr}\mathbf{C}\leqq nN-1$. Thus $\operatorname{tr}\mathbf{C}^{-1}\geqq n^2/\operatorname{tr}\mathbf{C}\geqq n^2/(Nn-1)$. Comparing the last term with $\operatorname{tr}(\mathbf{X}_3'\mathbf{X}_3)^{-1}=(n-1)/(N+1)+1/(N-n+1)$, we conclude that if $N\geqq n^2-2$, then $\operatorname{tr}\mathbf{C}^{-1}\geqq\operatorname{tr}(\mathbf{X}_3'\mathbf{X}_3)^{-1}$.

The proof is completed by taking

$$N_0(n)=\max\{(n-2)[n^2-n+16+\{n(n-1)(n^2-n+16)\}^{1/2}]/16,\ n^2-2\},$$

or

$$\max\{(n-2)(n^2-n+16)/8,\ n^2-2\}$$

for simplicity.

Thus $\mathbf{X}_3$ is A-optimal in $\mathscr{D}(N,n)$ if N is sufficiently large. Later we shall give an example showing that $\mathbf{X}_3$ is not always A-optimal. We remark that Lemmas 2.1, 2.2 and 2.3 are useful in proving or disproving the A-optimality of $\mathbf{X}_3$ over $\mathscr{D}'(N,n)$ when N is smaller than $(n-2)(n^2-n+16)/8$ since the three lemmas allow us to eliminate a large number of competitors; those that remain can be checked by a computer, as we shall illustrate in the examples below.

Consider $N=15$ and $n=6$. In this case $n(n-1)N^2/(N-n+2)^2\approx 55.8$. It is easy to see that all the matrices in $\mathscr{C}'(15,6)$ with $\sum\sum_{i\neq j}c_{ij}^2\leqq 55$ must have $|c_{ij}|=1$ for all $i\neq j$ except for at most a pair of off-diagonal elements with $|c_{ij}|=3$. By Lemmas 2.2 and 2.3, the only competitor in $\mathscr{C}'(15,6)$, up to equivalence, is the matrix

$$\mathbf{C}=\begin{bmatrix}15&3&-1&-1&-1&-1\\3&15&-1&-1&-1&-1\\-1&-1&15&-1&-1&-1\\-1&-1&-1&15&-1&-1\\-1&-1&-1&-1&15&-1\\-1&-1&-1&-1&-1&15\end{bmatrix}.$$

Computer calculation gives $\operatorname{tr}\mathbf{C}^{-1}=0.4151$, while $\operatorname{tr}(16\mathbf{I}_6-\mathbf{J}_6)^{-1}=0.4125$. Thus $\mathbf{X}_3$ is A-optimal in $\mathscr{D}'(15,6)$.

Similarly one can show that $\mathbf{X}_3$ is A-optimal in $\mathscr{D}'(15,7)$. Since $N\geqq(n-2)(n^2-n+16)/8$ for $N=15$ and $n=2,3,4,5$, it follows from Theorem 2.1 that $\mathbf{X}_3$ is also A-optimal in $\mathscr{D}'(15,n)$ for $n=2,3,4,5$. Thus we have shown that $\mathbf{X}_3$ is A-optimal in $\mathscr{D}'(15,n)$ for all $n\leqq 7$.

To save space, we remark that it has been shown that $\mathbf{X}_3$ is A-optimal in $\mathscr{D}'(N, n)$ for all $N \geqq 15$, $N \equiv 3 \pmod 4$ and all $n \leqq 7$.

To show the strength of our results, we conclude by giving an example which shows that in general $(N+1)\mathbf{I}_n - \mathbf{J}_n$ is not A-optimal in $\mathscr{C}'(N, n)$. This example points out that a matrix that is A-optimal in $\mathscr{C}'(N, n)$ need not be D-optimal in $\mathscr{C}'(N, n)$.

Let $N = 11$, $n = 7$. From a theorem of Galil and Kiefer (1980, p. 1299), the matrix $12\mathbf{I}_7 - \mathbf{J}_7$ is the *unique* D-optimal matrix in $\mathscr{C}'(11, 7)$. It should be noted that $12\mathbf{I}_7 - \mathbf{J}_7$ can be realized as $\mathbf{X}'\mathbf{X}$, where $\mathbf{X}$ is an 11×7 matrix with $x_{ij} = \pm 1$. Indeed, any $\mathbf{X}$ such that

$$\mathbf{H} = \begin{bmatrix} 1 & 1 \cdots 1 \\ 1 & \\ \vdots & \mathbf{X} \quad \mathbf{Y} \\ 1 & \end{bmatrix}$$

is a Hadamard matrix of order 12 will do. Such an $\mathbf{H}$ can be found in Hedayat and Wallis (1978). However let

$$\mathbf{Z} = \begin{bmatrix} 1 & 1 & -1 & -1 & -1 & -1 & -1 \\ -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ -1 & -1 & 1 & -1 & -1 & -1 & 1 \\ -1 & -1 & -1 & -1 & 1 & 1 & 1 \\ -1 & -1 & -1 & 1 & -1 & 1 & 1 \\ -1 & 1 & 1 & -1 & 1 & 1 & -1 \\ -1 & -1 & -1 & 1 & 1 & -1 & -1 \\ -1 & -1 & 1 & 1 & 1 & -1 & -1 \\ -1 & 1 & 1 & 1 & -1 & -1 & 1 \\ -1 & -1 & 1 & -1 & -1 & 1 & -1 \end{bmatrix};$$

then

$$\mathbf{Z}'\mathbf{Z} = \begin{bmatrix} 11 & 3 & -1 & -1 & -1 & -1 & -1 \\ 3 & 11 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & 11 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & 11 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & 11 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 & 11 & -1 \\ -1 & -1 & -1 & -1 & -1 & -1 & 11 \end{bmatrix}.$$

It is well known that if

$$\mathbf{M} = \begin{pmatrix} \mathbf{P} & \mathbf{Q} \\ \mathbf{Q}' & \mathbf{R} \end{pmatrix}$$

is a nonsingular matrix then

$$\mathbf{M}^{-1} = \begin{pmatrix} \mathbf{S} & \mathbf{T} \\ \mathbf{T}' & \mathbf{U} \end{pmatrix}$$

where $\mathbf{S}=\mathbf{P}^{-1}+\mathbf{P}^{-1}\mathbf{Q}(\mathbf{R}-\mathbf{Q}'\mathbf{P}^{-1}\mathbf{Q})^{-1}\mathbf{Q}'\mathbf{P}^{-1}$, $\mathbf{U}=(\mathbf{R}-\mathbf{Q}'\mathbf{P}^{-1}\mathbf{Q})^{-1}$ and $\mathbf{T}=-\mathbf{P}^{-1}\mathbf{Q}(\mathbf{R}-\mathbf{Q}'\mathbf{P}^{-1}\mathbf{Q})^{-1}$. Applying this result to $\mathbf{Z}'\mathbf{Z}$ we obtain

$$(\mathbf{Z}'\mathbf{Z})^{-1}=\begin{bmatrix}\begin{pmatrix}441/4312 & -98/4312\\ -98/4312 & 441/4312\end{pmatrix} & \mathbf{T}\\ \mathbf{T}' & (1/12)\mathbf{I}_5+(1/66)\mathbf{J}_5\end{bmatrix}.$$

Then $\operatorname{tr}(\mathbf{Z}'\mathbf{Z})^{-1}=441/2156+65/132=.69696$. But $\operatorname{tr}[12\mathbf{I}_7-\mathbf{J}_7]^{-1}=.7$. Thus $12\mathbf{I}_7-\mathbf{J}_7$ is not A-optimal in $\mathscr{C}'(11,7)$. Also the A-optimal matrix in $\mathscr{C}'(11,7)$ cannot be D-optimal since $12\mathbf{I}_7-\mathbf{J}_7$ is the unique D-optimal matrix in $\mathscr{C}'(11,7)$.

3. Φ_p-optimality of $\mathbf{X}_3$, $0\leqq p\leqq 1$. In this section, we shall extend the main result of § 2 to other criteria. A technique is developed to show that if a certain design is A-optimal, then it is also optimal with respect to a large class of other criteria. The method used is a modification of the result of Cheng (1978).

Throughout this section, we shall denote $\operatorname{tr}(\mathbf{X}_3'\mathbf{X}_3)^{-1}$ by S^*.

LEMMA 3.1. *For any positive numbers A, S and r such that $S>n^2/A$, $0\leqq r\leqq n$, the system of equations $(n-r)\mu+r\lambda=A$ and $(n-r)/\mu+r/\lambda=S$, $\mu\leqq\lambda$, has exactly one solution $\lambda(r;A,S)>\mu(r;A,S)>0$.*

Proof. By solving for μ in the first equation and substituting into the second, we obtain the equation $Sr\lambda^2+(n^2-2nr-SA)\lambda+rA=0$. The discriminant of the quadratic is $h(S)=(n^2-2nr-SA)^2-4ASr^2$. The result now follows by noting that $h(n^2/A)=0$ and $h'(S)=2A(SA-n^2)+4A(nr-r^2)>0$ if $S>n^2/A$.

LEMMA 3.2. *Let $\mu(r;A,S)$ and $\lambda(r;A,S)$ be as in Lemma* 3.1 *with $A\leqq nN$. Let f be a real-valued function defined on $[0,nN]$ such that*

(3.1)
(i) *f is continuous on $(0,nN)$ (we allow $\lim_{x\to 0^+}f(x)=f(0)=+\infty$);*
(ii) *$g''<0$ on $(1/nN,\infty)$, where $g(x)=f(1/x)$;*
(iii) *$f''>0$ on $(0,nN)$;*
(iv) *for $a<b$ in $(0,nN)$, $\{f(b)-f(a)\}/(b-a)<\{af'(a)+bf'(b)\}/(a+b)$;*

and $F(r;A,S)\equiv(n-r)f\{\mu(r;A,S)\}+rf\{\lambda(r;A,S)\}$. Then F is a strictly decreasing function of r, strictly increasing function of S and strictly decreasing function of A.

Proof. By differentiating the equations $(n-r)\mu+r\lambda=A$ and $(n-r)/\mu+r/\lambda=S$ with respect to r and S, and then solving the resulting equations for $\partial\mu/\partial r$, $\partial\lambda/\partial r$, $\partial\mu/\partial S$ and $\partial\lambda/\partial S$, we obtain

$$\frac{\partial\mu}{\partial r}=-\frac{(\lambda-\mu)\mu}{(n-r)(\lambda+\mu)},\quad \frac{\partial\lambda}{\partial r}=-\frac{(\lambda-\mu)\lambda}{r(\lambda+\mu)},$$

$$\frac{\partial\mu}{\partial S}=-\frac{\lambda^2\mu^2}{(n-r)(\lambda^2-\mu^2)},\quad \frac{\partial\lambda}{\partial S}=\frac{\lambda^2\mu^2}{r(\lambda^2-\mu^2)}.$$

Thus

$$\frac{\partial F}{\partial r}=-f(\mu)+(n-r)f'(\mu)\frac{\partial\mu}{\partial r}+f(\lambda)+rf'(\lambda)\frac{\partial\lambda}{\partial r}$$

$$=f(\lambda)-f(\mu)-\frac{(\lambda-\mu)\{\mu f'(\mu)+\lambda f'(\lambda)\}}{\lambda+\mu}$$

$$<0,\quad \text{by (iv).}$$

Similarly it follows from (iii) that

$$\frac{\partial F}{\partial S} > 0.$$

We remark that if $f'' < 0$ on $(0, nN)$, then F is a decreasing function of S. This fact will be used below in the proof of the decreasing monotonicity of F in A.

To show that F is a decreasing function of A, we write $F = (n-r)g(\mu') + rg(\lambda')$, where $\mu' = \mu^{-1}$, $\lambda' = \lambda^{-1}$ and $g(x) = f(x^{-1})$. Then $(n-r)\mu' + r\lambda' = S$ and $(n-r)/\mu' + r/\lambda' = A$. By (ii) and the remark in the last paragraph, we conclude that F is a strictly decreasing function of A. This completes the proof.

LEMMA 3.3. *Let* $H = \{(x_1, x_2, \cdots, x_n)\colon x_i > 0, \sum_{i=1}^n x_i = A, \sum_{i=1}^n x_i^{-1} = S\}$, *where* $A \leqq nN$ *and* $S \geqq n^2/A$. *Also let* $F_f : H \to R$ *be defined by* $F_f(x_1, x_2, \cdots, x_n) = \sum_{i=1}^n f(x_i)$, *where* f *is a real-valued function defined on* $[0, nN]$ *which satisfies* (3.1) *and the following condition*:

(3.2) *The equation* $x^2 f'(x) + \alpha x^2 - \beta = 0$ *has at most two solutions in* $(0, nN)$ *for all real numbers* α *and* β.

Then the minimum of $F_f(x_1, x_2, \cdots, x_n)$ *on* H *occurs at the point* $(\mu(n-1; A, S), \lambda(n-1; A, S), \cdots, \lambda(n-1; A, S))$, *where* $\mu(r; A, S)$ *and* $\lambda(r; A, S)$ *are as in Lemma* 3.1.

Proof. Since H is compact, the minimum of F_f on H is attained at some point $(a_1, a_2, \cdots, a_n)$. Clearly not all the a_i's are equal (since $S > n^2/A$), so by the symmetry of F_f, we may assume $a_1 < a_2$. Letting $g_1(x_1, x_2, \cdots, x_n) = \sum_{i=1}^n x_i - A$ and $g_2(x_1, x_2, \cdots, x_n) = \sum_{i=1}^n x_i^{-1} - S$; then $\det\{D_j g_i(a_1, a_2, \cdots, a_n)\}_{i,j=1,2} = a_1^{-2} - a_2^{-2} \neq 0$. So by Lagrange's theorem (see Apostol (1974, p. 381)), there exist numbers α and β such that $a_1, a_2, \cdots, a_n$ satisfy the following equations:

$$\frac{\partial}{\partial x_i}\{F_f(x_1, x_2, \cdots, x_n) + \alpha g_1(x_1, x_2, \cdots, x_n) + \beta g_2(x_1, x_2, \cdots, x_n)\} = 0, \quad i = 1, 2, \cdots, n. \tag{3.3}$$

Now (3.3) simplifies to $f'(x_i) + \alpha - \beta x_i^{-2} = 0$, $i = 1, 2, \cdots, n$. By (3.2), each x_i can take on at most two possible values. So the point $(a_1, a_2, \cdots, a_n)$ must be such that $a_i = a_1$ or a_2. The result now follows from Lemma 3.1 and Lemma 3.2. □

Now we are ready to prove the main result of this section.

THEOREM 3.1. *If* $\mathbf{X}_3$ *is A-optimal in* $\mathcal{D}(N, n)$, *then it also minimizes* $\Phi_f(\mathbf{X}'\mathbf{X}) \equiv \sum_{i=1}^n f(\mu_i)$ *over* $\mathcal{D}(N, n)$ *for all* f *satisfying* (3.1) *and* (3.2), *where* $\mu_1, \mu_2, \cdots, \mu_n$ *are the eigenvalues of* $\mathbf{X}'\mathbf{X}$.

Proof. For any $\mathbf{X} \in \mathcal{D}(N, n)$, let $S = \operatorname{tr}(\mathbf{X}'\mathbf{X})^{-1}$, $A = \operatorname{tr}(\mathbf{X}'\mathbf{X})$; also denote $\operatorname{tr}(\mathbf{X}_3'\mathbf{X}_3)^{-1}$ by S^*. Then by assumption, $S \geqq S^*$. So by Lemmas 3.2 and 3.3, we have

$$\begin{aligned}
\sum_{i=1}^n f(\mu_i) &\geqq (n-1)f(\lambda(n-1; A, S)) + f(\mu(n-1; A, S)) \\
&\geqq (n-1)f(\lambda(n-1; A, S^*)) + f(\mu(n-1; A, S^*)) \\
&\geqq (n-1)f(\lambda(n-1; nN, S^*)) + f(\mu(n-1; nN, S^*)) \\
&= \sum_{i=1}^n f(\mu_i^*),
\end{aligned}$$

where $\mu_1^*, \mu_2^*, \cdots, \mu_n^*$ are the eigenvalues of $\mathbf{X}_3'\mathbf{X}_3$. The last equality holds since

$\operatorname{tr} \mathbf{X}_3'\mathbf{X}_3 = nN$, $\operatorname{tr} (\mathbf{X}_3'\mathbf{X}_3)^{-1} = S^*$, and $\mathbf{X}_3'\mathbf{X}_3$ has two distinct eigenvalues with the smaller one being a simple root. This completes the proof.

Combining Theorems 2.1 and 3.1, we have:

COROLLARY 3.1. *For each n, there exists a positive integer $N_0(n)$ such that for all $N \geqq N_0(n)$, $\mathbf{X}_3$ minimizes $\sum_{i=1}^{n} f(\mu_i)$ over $\mathcal{D}(N, n)$ for all f satisfying the conditions in* (3.1) *and* (3.2).

We shall conclude this section by deriving a simple sufficient condition for (3.1) and (3.2).

LEMMA 3.4. *If $f:[0, nN] \to R$ is such that* (i), (ii), (iii) *in* (3.1) *hold, and $x^3 f''(x)$ is an increasing function on $(0, nN)$, then all the conditions in* (3.1) *and* (3.2) *hold.*

Proof. We need to verify (3.2) and condition (iv) in (3.1). Now the latter is equivalent to $\{af'(a)+bf'(b)\}(b-a)-\{f(b)-f(a)\}(a+b)>0$. For fixed a, let $h(b)=\{af'(a)+bf'(b)\}(b-a)-\{f(b)-f(a)\}(a+b)$. Then since $h(a)=0$, to show $h(b)>0$ for all $b>a$, it suffices to prove $h'(b)>0$ for all $b>a$. Now $h'(b)=f'(b)(b-a)+a\{f'(a)-f'(b)\}+bf''(b)(b-a)-f(b)+f(a)$. So it is enough to prove $h''(b)>0$ for all $b>a$. We have $h''(b)=3f''(b)(b-a)+bf'''(b)(b-a)$. Since $x^3f''(x)$ is an increasing function, $d\{x^3f''(x)\}/dx>0$, i.e.,

$$xf'''(x)+3f''(x)>0 \quad \text{for all } x \in (0, nN). \tag{3.4}$$

Therefore $h''(b)=\{bf'''(b)+3f''(b)\}(b-a)>0$ for all $b>a$. This proves condition (iv) in (3.1).

Now we prove (3.2). Suppose x and y are two distinct solutions of the equation $x^2f'(x)+\alpha x^2-\beta=0$. Then $x^2f'(x)+\alpha x^2-\beta=y^2f'(y)+\alpha y^2-\beta$ and hence

$$\frac{\{x^2f'(x)-y^2f'(y)\}}{x^2-y^2}=-\alpha. \tag{3.5}$$

Let $g(x)=xf'(\sqrt{x})$. We shall show that g is a strictly convex function; then for each fixed x, there is at most one y satisfying (3.5) and therefore (3.2) is proved. Now $g''(x)=4x^{-1/2}\{f'''(\sqrt{x})\sqrt{x}+3f''(\sqrt{x})\}$ which, by (3.4), is positive. This completes the proof.

It is easy to see that if $0<p<1$, then the function $f(x)=x^{-p}$ satisfies the conditions in Lemma 3.4. Therefore Theorem 3.1 and Corollary 3.1 hold for all the Φ_p-criteria with $0<p\leqq 1$. By passing to the limit or taking $f(x)=-\log x$, D-optimality also follows. We state this in the following

COROLLARY 3.2. *For each n, there exists a positive integer $N_0(n)$ such that for all $N \geqq N_0(n)$, $\mathbf{X}_3$ is Φ_p-optimal over $\mathcal{D}(N, n)$ for all $0 \leqq p \leqq 1$; in particular, it is D-optimal.*

As shown in Theorem 2.1, one can take $N_0(n)=\max\{(n-2)(n^2-n+16)/8, n^2-2\}$. This is by no means the smallest bound, But since our result is much stronger than D-optimality, the smallest $N_0(n)$ must be larger than $2n-5$, the bound found by Galil and Kiefer (1980) for the D-criterion.

Remark. Using a similar method, one can generalize Corollary 3.2 to show that for any n and $p>0$, there exists a positive integer $N(n, p)$ such that for all $N \geqq N(n, p)$, $\mathbf{X}_3$ is Φ_q-optimal over $\mathcal{D}(N, n)$ for all $0 \leqq q \leqq p$. However, since $\mathbf{X}_3$ is not E-optimal (even when N gets large), one has $\overline{\lim}_{p\to\infty} N(n, p)=\infty$. Such a result is not very useful practically. Furthermore, for $p \neq 1$, there is no simple way to calculate a bound for $N(n, p)$ as we did in § 2 for $p=1$.

4. Φ_p-optimality of $\mathbf{X}_2$, $0 \leqq p \leqq 1$. In this section, we shall prove a result similar to Corollary 3.2 for $\mathbf{X}_2$. We now assume $N \equiv 2 \pmod 4$.

THEOREM 4.1. *For any n, there exists $N_0(n)$ such that if $N \geqq N_0(n)$, then $\mathbf{X}_2$ is Φ_p-optimal over $\mathcal{D}(N, n)$ for all $0 \leqq p \leqq 1$.*

Proof. Jacroux, Masaro and Wong (1983) proved that $\mathbf{X}_2$ is optimal over $\mathscr{D}'(N, n)$ with respect to all the type 1 criteria of Cheng (1980); in particular it is Φ_p-optimal for all $p \geqq 0$. So it suffices to consider matrices $\mathbf{C}$ in $\mathscr{C}(N, n)\backslash\mathscr{C}'(N, n)$. We have to show that there exists an integer N_0 such that if $N \geqq N_0$, then for any $\mathbf{C} \in \mathscr{C}(N, n)\backslash\mathscr{C}'(N, n)$, $\Phi_p(\mathbf{C}) > \Phi_p(\mathbf{X}_2'\mathbf{X}_2)$ for all $0 \leqq p \leqq 1$.

Now if $\mathbf{C} \in \mathscr{C}(N, n)\backslash\mathscr{C}'(N, n)$, then $\operatorname{tr}\mathbf{C} \leqq Nn-1$. Since $\operatorname{tr}\{n^{-1}(Nn-1)\mathbf{I}_n\} = Nn-1$, we have

$$\Phi_p(\mathbf{C}) \geqq \Phi_p(n^{-1}(Nn-1)\mathbf{I}_n) \quad \text{for all } p \geqq 0. \tag{4.1}$$

By a direct comparison of $\operatorname{tr}\{n^{-1}(Nn-1)\mathbf{I}_n\}^{-1} = n^2/(Nn-1)$ with

$$\operatorname{tr}(\mathbf{X}_2'\mathbf{X}_2)^{-1} = \begin{cases} \dfrac{n-2}{N-2} + \dfrac{2}{N+n-2} & \text{if } n \text{ is even,} \\ \dfrac{n-2}{N-2} + \dfrac{1}{N+n-1} + \dfrac{1}{N+n-3} & \text{if } n \text{ is odd,} \end{cases}$$

one can easily show that there exists an integer N_0 such that

$$N \geqq N_0 \Rightarrow \Phi_1(n^{-1}(Nn-1)\mathbf{I}_n) > \Phi_1(\mathbf{X}_2'\mathbf{X}_2). \tag{4.2}$$

Now let $\mu_1^*, \mu_2^*, \cdots, \mu_n^*$ be the eigenvalues of $\mathbf{X}_2'\mathbf{X}_2$. Then $\Phi_p(\mathbf{X}_2'\mathbf{X}_2) = \{n^{-1}\sum_{i=1}^n (\mu_i^*)^{-p}\}^{1/p}$ is an increasing function of $p > 0$. Since all the eigenvalues of $n^{-1}(Nn-1)\mathbf{I}_n$ are equal, we have $\Phi_p\{n^{-1}(Nn-1)\mathbf{I}_n\} = n/(Nn-1)$ for all $p > 0$. By (4.2), if $N \geqq N_0$ and $0 < p \leqq 1$, then

$$\Phi_p(\mathbf{X}_2'\mathbf{X}_2) \leqq \Phi_1(\mathbf{X}_2'\mathbf{X}_2) < \Phi_1\{n^{-1}(Nn-1)\mathbf{I}_n\} = \Phi_p\{n^{-1}(Nn-1)\mathbf{I}_n\}.$$

Combining this with (4.1), we conclude that if $N \geqq N_0$, then $\mathbf{X}_2$ is Φ_p-optimal over $\mathscr{D}(N, n)$ for all $0 < p \leqq 1$. The D-optimality ($p = 0$) is obtained by passing to limit. □

We remark that there is nothing like Theorem 3.1 for $\mathbf{X}_2$; the A-optimality of $\mathbf{X}_2$ does not guarantee its Φ_p-optimality for $0 \leqq p \leqq 1$. Thus if $N_0(n)$ is the smallest integer such that $N \geqq N_0(n) \Rightarrow \mathbf{X}_2$ is A-optimal over $\mathscr{D}(N, n)$ and $N_0^*(n)$ is the smallest integer such that $N \geqq N_0^*(n) \Rightarrow \mathbf{X}_2$ is Φ_p-optimal over $\mathscr{D}(N, n)$ for all $0 \leqq p \leqq 1$, then it is not clear whether $N_0(n)$ is equal to $N_0^*(n)$. For $\mathbf{X}_3$, we know that the two numbers are equal. The proof of Theorem 4.1 indicates that an upper bound for $N_0(n)$ can be obtained by comparing $\operatorname{tr}(\mathbf{X}_2'\mathbf{X}_2)^{-1}$ with $\operatorname{tr}\{n(Nn-1)^{-1}\mathbf{I}_n\}$. This usually produces a value that is too large. A much better bound can be obtained by the following method. For any $\mathbf{X} \in \mathscr{D}(N, n)$, suppose there are k columns which contain zero entries. Without loss of generality, we may assume that $\mathbf{X} = (\mathbf{Y}_1\mathbf{Y}_2)$, where $\mathbf{Y}_1$ is $N \times k$ and consists of all the columns which contain zero entries. By a result of Fan (1954), the eigenvalues of $\mathbf{X}'\mathbf{X}$ majorize those of

$$\begin{pmatrix} \mathbf{Y}_1'\mathbf{Y}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{Y}_2'\mathbf{Y}_2 \end{pmatrix}.$$

Since all the diagonal elements of $\mathbf{Y}_1'\mathbf{Y}_1$ are $\leqq N-1$, we have

$$\operatorname{tr}(\mathbf{X}'\mathbf{X})^{-1} \geqq \operatorname{tr}\begin{pmatrix} \mathbf{Y}_1'\mathbf{Y}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{Y}_2'\mathbf{Y}_2 \end{pmatrix}^{-1} \geqq \operatorname{tr}\begin{pmatrix} (N-1)\mathbf{I}_k & \mathbf{0} \\ \mathbf{0} & \mathbf{Y}_2'\mathbf{Y}_2 \end{pmatrix}^{-1}.$$

Now all the entries of $\mathbf{Y}_2$ are ± 1, i.e., $\mathbf{Y}_2'\mathbf{Y}_2 \in \mathscr{C}'(N, n-k)$. By an argument similar to

that employed in Jacroux, Masaro and Wong (1983), we conclude that

$$\operatorname{tr}(\mathbf{X}'\mathbf{X})^{-1} \geqq \operatorname{tr}\begin{pmatrix} (N-1)\mathbf{I}_k & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & (N-2)\mathbf{I}_l + 2\mathbf{J}_l & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & (N-2)\mathbf{I}_{n-k-l} + 2\mathbf{J}_{n-k-l} \end{pmatrix}^{-1},$$

where $l = [(n-k)/2]$. Thus an upper bound of $N_0(n)$ is the smallest N such that

$$\operatorname{tr}\begin{pmatrix} (N-2)\mathbf{I}_t + 2\mathbf{J}_t & \mathbf{0} \\ \mathbf{0} & (N-2)\mathbf{I}_{n-t} + 2\mathbf{J}_{n-t} \end{pmatrix}^{-1}$$

$$\leqq \operatorname{tr}\begin{pmatrix} (N-1)\mathbf{I}_k & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & (N-2)\mathbf{I}_l + 2\mathbf{J}_l & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & (N-2)\mathbf{I}_{n-k-l} + 2\mathbf{J}_{n-k-l} \end{pmatrix}^{-1}$$

for all k such that $1 \leqq k \leqq n$, where $t = [n/2]$ and $l = [(n-k)/2]$. Table 1 shows some upper bounds of $N_0(n)$ obtained by the above method.

TABLE 1

n	4	5	6	7	8	9	10	11	12
an upper bound of $N_0(n)$	10	14	14	18	18	22	22	30	30

These bounds certainly are not sharp. In fact, it has been shown by Wong and Masaro (1982) that $\mathbf{X}_2$ is A-optimal over $\mathcal{D}(N, n)$ for all $n \leqq 6$ and all $N \geqq n$. Thus $N_0(4) = N_0(5) = N_0(6) = 6$. Although the example in § 2 shows that $\mathbf{X}_3$ is not always A-optimal, we have not been able to find an $\mathbf{X}_2$ which is not A-optimal in $\mathcal{D}(N, n)$! The above table is useful for eliminating a lot of cases that have to be considered in proving the A-optimality of $\mathbf{X}_2$.

REFERENCES

[1] T. M. APOSTOL, *Mathematical Analysis*, 2nd ed., Addison-Wesley, Reading, MA, 1974.

[2] J. BRENNER AND L. CUMMINGS, *The Hadamard maximum determinant problem*, Amer. Math. Monthly, 79 (1972), pp. 626-630.

[3] C. S. CHENG, *Optimality of certain asymmetrical experimental designs*, Ann. Statist., 6 (1978), pp. 1239-1261.

[4] ———, *Optimality of some weighing and* 2^n *fractional factorial designs*, Ann. Statist., 8 (1980), pp. 436-446.

[5] H. EHLICH, *Determinantenabschätzungen für binäre Matrizen*, Math. Z., 83 (1964), pp. 123-132.

[6] K. FAN, *Inequalities for eigenvalues of Hermitian matrices*, Nat. Bur. Standards Appl. Math. Ser., 39 (1954), pp. 131-139.

[7] Z. GALIL AND J. KIEFER, *D-optimum weighing designs*, Ann. Statist., 8 (1980), pp. 1293-1306.

[8] M. HARWIT AND N. J. A. SLOANE, *Hadamard Transform Optics*, Academic Press, New York, 1979.

[9] A. HEDAYAT AND W. D. WALLIS, *Hadamard matrices and their applications*, Ann. Statist., 6 (1978), pp. 1184-1238.

[10] M. JACROUX, J. MASARO AND C. S. WONG, *On the optimality of chemical balance weighing designs*, J. Statist. Plann. Inference, 8 (1983), pp. 231-240.

[11] J. KIEFER, *Optimality and construction of generalized Youden designs*, in A Survey of Statistical Designs and Linear Models, J. N. Srivastava, ed., North-Holland, Amsterdam, 1975, pp. 333-353.

[12] S. E. PAYNE, *On maximizing* $\det(A^TA)$, Discrete Math., 10 (1974), pp. 145-158.

[13] C. S. WONG AND J. C. MASARO, *A-optimal design matrices*, unpublished manuscript, 1982.

SIAM J. ALG. DISCR. METH.
Vol. 6, No. 2, April 1985

011

ON THE CUTWIDTH AND THE TOPOLOGICAL BANDWIDTH OF A TREE*

FAN R. K. CHUNG†

Abstract. We investigate the relations between the topological bandwidth $b^*(G)$ and the cutwidth $f(G)$ for a graph G. We show that for any tree T we have $b^* \leqslant f(T) \leqslant b^*(T)+\log_2 b^*(T)+2$. These bounds are "almost" best possible, since we will prove that for each n, there exists a tree T_n such that $b^*(T_n) = n$ and $f(T_n) \geqslant n+\log_2 n-1$, and the star S_{2n} with 2n edges satisfies $b^*(S_{2n}) = f(S_{2n}) = n$.

1. Introduction. Suppose G is a graph with vertex set $V(G)$ and edge set $E(G)$. A numbering π of G is a one-to-one mapping from $V(G)$ to the set of positive integers. Such a numbering can be viewed as describing a placement of the vertices of G on a line, so it is not surprising that graph numbering problems are frequently relevant to circuit layout and design. The following objective functions will be of interest in this paper.

(i) The bandwidth $b_\pi(G)$ of a numbering π is defined to be

$$b_\pi(G) = \max\{|\pi(u)-\pi(v)|: \{u,v\} \in E(G)\}$$

and the bandwidth $b(G)$ of G is the minimum of $b_\pi(G)$ over all numberings π of G.

(ii) The topological bandwidth $b^*(G)$ of a graph G is defined to be

$$b^*(G) = \min\{b(G'): G' \text{ is a refinement of } G\}$$

(A graph G' is said to be a refinement of G if G' is obtained from G by a finite number of edge subdivisions.)

(iii) Define

$$f_\pi(G) = \max_i |\{\{u,v\} \in E(G): \pi(u) \leqslant i < \pi(v)\}| .$$

Then the cutwidth [12] $f(G)$ of a graph G is defined to be

$$f(G) = \min_\pi f_\pi(G).$$

We will show that for any tree T the following holds:

$$b^*(T) \leqslant f(T) \leqslant b^*(T)+\log_2 b^*(T)+2 .$$

These bounds are "almost" best possible, since we will prove that for each n, there exists a tree T_n such that $b^*(T_n) = n$ and $f(T_n) \geqslant n+\log_2 n-1$, and the star S_{2n} with $2n$ edges satisfies $b^*(S_{2n}) = f(S_{2n}) = n$.

We remark that the upper bound does not hold for general graphs since for the complete graph K_n on n vertices we have $b^*(K_n) = n-1$ and $f(K_n) = \lceil (n^2-1)/4 \rceil$, though it can be shown that $b^*(G) \leqslant f(G)$ for general graphs G.

(A numbering of a graph is also called a linear arrangement of a graph [6]. The cutwidth of a graph is sometimes called the folding number of a graph [2].)

As to the algorithmic aspects, the bandwidth problem for graphs is known to be

*Received by the editors April 12,1983, and in revised form February 15, 1984. This paper was typeset at AT&T Bell Laboratories, Murray Hill, New Jersey, using the **troff** program running under the Unix™ operating system. Final copy was produced on July 27, 1984.

†Bell Communications Research, Murray Hill, New Jersey 07974.

NP-complete [6], [9] as is the bandwidth problem for trees [5]. The cutwidth problem for graphs is also *NP*-complete [4], while the cutwidth problem for trees can be solved in $O(n \log n)$ time [13] (also see [3] for degree restricted cases). The topological bandwidth problem for graphs is recently proved to be *NP*-complete [8].

We remark that the minimum sum problem of finding $\min_\pi \sum_{\{u,v\}\in E(G)} |\pi(u)-\pi(v)|$ is *NP*-complete for graphs [8] while there are polynomial time algorithms for the minimum sum problem for trees [7].

2. Preliminaries. In this section we will discuss several properties of numberings [2] that will be useful later.

Let π denote a numbering of a tree T mapping $V(T)$ to $\{1,...,n\}$ where $n = |V(T)|$. We say π satisfies

(i) The leaf property, if the vertices numbered by 1 and n are leaves.

(ii) The monotone property, if the following is true: Let P denote the path, called the basic path of π, in T connecting the two vertices numbered by 1 and n. Suppose P has vertices $v_0,v_1,...,v_t$ with v_i adjacent to v_{i+1}. Then π is monotone if the numberings of the vertices of P are monotone, i.e.,

$$\pi(v_i) < \pi(v_{i+1}) \quad \text{for } v = 0,1,...,t-1 \text{ or}$$
$$\pi(v_i) > \pi(v_{i+1}) \quad \text{for } v = 0,1,...,t-1 .$$

(iii) The block property, if the following is true: Let F denote the forest formed by removing the edges of P from T (but let the vertices stay). Then any maximal tree in F is numbered by a set of consecutive integers.

(iv) The weak block property, if the following is true: Let $\bar{T}$ denote a maximal subtree in F. Suppose $x = \min\{\pi(u): u \in \bar{T}\}$ and $\underline{y} = \max\{\pi(u): u \in \bar{T}\}$. Then any vertex v with $x \leqslant \pi(v) \leqslant y$ is either in $\bar{T}$ or on P.

(v) The hereditary property, if the induced numbering for each subtree $\bar{T}$ of F is an optimal numbering with respect to the objective function of interest. (The induced numbering π' of π on T' is the one-to-one mapping from $V(T')$ to the set $\{1,2,...,|V(T')|\}$ such that for any $\{u,v\}$ in $E(T')$, $\pi'(u) < \pi'(v)$ if $\pi(u) < \pi(v)$. π' is denoted by π/T'.)

It is easy to check that for a given tree T there exists a bandwidth numbering π with $b_\pi(T) = \underline{b}(T)$ satisfying the leaf property. Also there exists a numbering $\bar{\pi}$ for a refinement $\bar{T}$ of T with $b_{\bar{\pi}}(\bar{T}) = b^*(T)$ satisfying the leaf property, the monotone property, and the weak block property. There always exists a cutwidth numbering λ with $f_\lambda(T) = f(T)$ satisfying the leaf property, the monotone property, the block property and the hereditary property.

Let π denote a numbering for a tree T. Then for any subtree T' in T, the basic path $P(\pi,T')$ of T' is the path joining the two vertices with the largest and smallest numbers in T'. Let $F(\pi,T,1)$ denote the forest obtained by removing the edges (not the vertices) of $P(\pi,T)$ from T. Let $F(\pi,T,i)$ denote the forest obtained by removing the edges of the basic paths of all maximal subtrees in $F(\pi,T,i-1)$. Then we have the following:

LEMMA 1. *Suppose* λ *is a cutwidth numbering for* T. *Then* $f(T) = 1 + \max_{T'} f(T')$ *for* T' *ranging over all maximal subtrees of* $F(\lambda,T,1)$.

Proof. This follows immediately from the monotone property, the block property and the hereditary property of λ.

LEMMA 2. *Suppose* λ *is a cutwidth numbering for* T. *Then* $f(T) = i + \max_{T'} f(T')$ *for* T' *ranging over all maximal subtrees of* $F(\lambda,T,i)$.

LEMMA 3. *If T' is a refinement of T, then we have*

$$f(T) = f(T') .$$

Proof. This follows from the fact that any numbering π of T can be extended to be a numbering $\bar{\pi}$ of T' with $f_\pi(T) = f_{\pi'}(T)$. On the other hand, for any numbering $\bar{\pi}$ of T' the induced numbering $\bar{\pi}/T$ of $\bar{\pi}$ on T satisfies $f_{\bar{\pi}/T}(T) \leqslant f_{\bar{\pi}}(T')$.

LEMMA 4. $f(T) \leqslant |V(T)|/2$.

Proof. This follows from the leaf property that any maximal subtree in $F(\lambda,F,1)$ has at most $|V(T)|-2$ vertices. Thus by Lemma 1 and by induction on $n = |V(T)|$ we have

$$f(T) = 1 + \max_{T'} f(T') \leqslant 1 + \frac{|V(T)|-2}{2} = \frac{|V(T)|}{2} .$$

LEMMA 5. *Suppose T' is a refinement of T. Then $b^*(T')$ can be different from $b^*(T)$.* (See Fig. 1.)

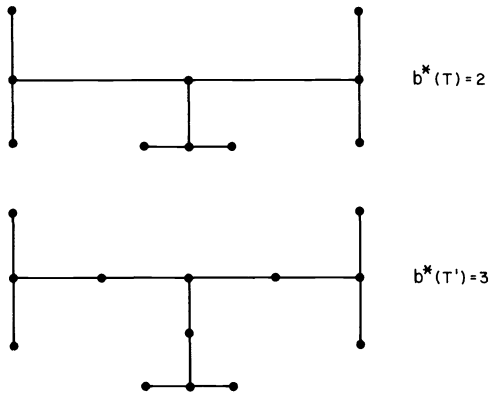


Fig. 1

Let us now define two functions, called the shifting function and the skipping function, from the set of integers Z to itself. The shifting function is $s_a(n) = a+n$ and the skipping function k_a is the order preserving function from Z to $Z - \{ia: i \in Z\}$.

LEMMA 6. *Suppose T is a tree which is the edge-disjoint union of a path P and a collection S of trees, say the ith vertex in P is in the ith tree in S. Then we have*

$$b^*(T) \leqslant 1 + \max_{T' \in S} b^*(T') .$$

Proof. Let $T_1,...,T_t$ denote the trees in S. Let T_i' be a refinement of T_i with a labeling π_i: $V(T_i') \to \{1,...,|V(T_i')|\}$ and $b_{\pi_i}(T_i') = b^*(T_i)$. We will combine the π_i to form a numbering π' for a refinement T' of T with $b_{\pi'}(T') = 1 + \max_i b^*(T_i) = 1+x$. Roughly speaking, the vertices of T_i' in T' are numbered in the same fashion as π_i except that the assigned values skip one out of every $x+1$ values. The numbering π' restricted to T_i' can be described as $s_{a_i}k_{x+1}\pi_i$ where

$$a_i = \lceil (\sum_{j<i} |V(T_i)|)(1 + \frac{1}{x}) \rceil + i|V(t)| .$$

Now we refine the basic path so that its vertices are numbered by a chain of numbers at most $x+1$ apart. Therefore we have

$$b^*(T) \leqslant b_\pi(T') = 1+x .$$

This completes the proof of Lemma 6.

LEMMA 7. *Suppose π is a numbering for a tree T and π satisfies the leaf property. Then we have*

$$b^*(T) \leqslant 1 + \max_{T'} b^*(T')$$

for T' ranging over all maximal subtrees in $F(\pi,T,1)$.

Proof. It follows from Lemma 6.

Let $F^*(\pi,T,1)$ denote the forest obtained by removing all vertices and edges in $P(\pi,T)$. Then we have the following.

LEMMA 8. *Suppose π is a bandwidth numbering of T. Then*

$$b(T) \geqslant 1 + \max_{T' \in F^*(\pi,T,1)} b(T') .$$

Proof. Suppose $b(T) = x$. For any vertex v in T with $\pi(v)+x \leqslant |V(T)|$ there is a vertex u in $P(\pi,T)$ such that $\pi(v) \leqslant \pi(u) \leqslant \pi(v)+x$. Thus for any T' in $F^*(\pi,T,1)$ the induced numbering of π on T' has bandwidth at most $x-1$.

3. The topological bandwidth is no larger than the cutwidth. It is easy to show that the topological bandwidth is no larger than the cutwidth numbering for a tree.

THEOREM 1. *$f(T) \geqslant b^*(T)$ for any tree T.*

Proof. We will prove this by induction on $|V(T)|$. Let λ denote the cutwidth numbering. Let T' denote a maximal subtree in $F(\lambda,T,1)$. We have

$$\begin{aligned} f(T) &\geqslant 1 + \max_{T'} f(T') && \text{(by Lemma 1)} \\ &\geqslant 1 + \max_{T'} b^*(T') && \text{(by induction and } |V(T')| < |V(T)|) , \\ &\geqslant b^*(T) && \text{(by Lemma 7)} . \end{aligned}$$

In fact, the topological bandwidth for a graph is no larger than its cutwidth. This has been observed by I. H. Sudborough and F. Makedon [11] among others. We will give the proof here.

THEOREM 2. *$f(G) \geqslant b^*(G)$ for any graph G.*

Proof. Let λ denote a cutwidth numbering of G. We will modify λ to obtain a numbering λ' of a refinement G' of G such that $b_{\pi'}(G') \leqslant f_\pi(G) = f(G) = x$. First we choose a subgraph G_1 of G as follows

Step 1: Set $C = \phi$.

Step 2: Choose an edge $\{u,v\}$ such that $\pi(u) \leqslant \pi(v)$ and u is the smallest vertex with $\pi(u) \geqslant \pi(w)$ for any w in a edge in C. Put $\{u,v\}$ into C and repeat Step 2. If no such edge exists, stop the process.

Clearly, the graph G_1 formed by edges in C has $f_\pi(G_1) = 1$. Also the graph $G-G_1$ obtained by removing edges in G_1 from G satisfies $f_\pi(G-G_1) = x-1$. (Otherwise, let i be the least number with $|\{\{u,v\} \in E(G-G_1): \pi(u) \leqslant i < \pi(v)\}| = x$. Then all

edges $\{u,v\}$ in G with $\pi(u) \leqslant i < \pi(v)$ are not in G_1. From Step 2 we know that there is no edge $\{u,v\}$ in G with $\pi(u) = i < \pi(v)$. Thus there are x edges $\{u,v\}$ with $\pi(u) < i < \pi(v)$. This implies $|\{\{u,v\} \in \pi(G-G_1): \pi(u) \leqslant i-1 < \pi(v)\}| = x$, contradicting the minimality of i.)

We can then repeat the process and partition G into $G_1, G_2, \ldots, G_x$, such that $f_\pi(G_i) = 1$ for $1 \leqslant i \leqslant x$. Now we consider a refinement G' of G as follows. For any edge $\{u,v\}$ in G_i with $\pi(u) < \pi(v)$, we subdivide $\{u,v\}$ into a path of $\pi(v) - \pi(u) + 1$ vertices, $u = u_0, u_1, \ldots, u_t = v$ where $t = \pi(v) - \pi(u)$. We define $\pi'(u_j)$ to be $x(\pi(u)+j) + i - 1$.

Clearly π' is a one-to-one function from $V(G')$ to Z. It is easily checked that $b_{\pi'}(G') = x$. Thus we have $f(G) = x = b_{\pi'}(G') \geqslant b^*(G)$.

4. The topological bandwidth for a tree is not equal to its cutwidth in general. For each integer n, we will construct a tree T_n satisfying $b^*(T_n) = n$ and $f(T_n) \geqslant n + \log_2 n - 1$. We will recursively build a rooted tree T_n^* (i.e., a tree with one special vertex) as follows: (i) T_1^* is a path with three vertices. The middle vertex is the root. (ii) For $n > 1$, T_n^* consists of a path P_n of 15 vertices and 15 copies of T_{n-1}^*. Each vertex in P_n is adjacent to the root of a copy of T_{n-1}^*. The root of T_n^* is the root of the T_{n-1}^* which is connected to the 8th vertex of P_n.

Let T_n denote the unrooted version of T_n^*.

CLAIM 1. $b^*(T_n) = n$.

Proof. We will prove this by induction on n. It is easily seen that $b^*(T_1) = 1$. Suppose a refinement $\overline{T}_i$ of T_i has bandwidth $\leqslant i$. We want to show that $b^*(T_{i+1}) = i+1$. Let π denote the numbering with (the refined) P_{i+1} as the basic path. Let T' denote a maximal subtree in $F^*(\pi, \overline{T}, 1)$. Then $T' \subseteq T_i$.

$$\begin{aligned} b^*(T_{i+1}) &\leqslant 1 + \max_{T'} b^*(T') \quad \text{(by Lemma 7)} \\ &\leqslant 1 + b^*(T_i) \leqslant 1 + i \end{aligned}$$

On the other hand, for any topological-bandwidth numbering π of T_{i+1}, $F^*(\pi, T_{i+1}, 1)$ must contain T_i. Thus we have

$$\begin{aligned} b^*(T_{i+1}) &\geqslant 1 + \max_{T' \in F^*(\pi,T,1)} b^*(T') \quad \text{(by Lemma 8)} \\ &\geqslant 1 + b^*(T_i) \\ &\geqslant 1 + i \ . \end{aligned}$$

Thus we have $b^*(T_{i+1}) = 1+i$.

CLAIM 2. $f(T_n) \geqslant n + \log_2 n - 1$.

Proof. This will be proved by induction on n. It is easy to see that $f(T_1) = 1$ and $f(T_2) = 3$. Suppose $f(T_j^*) \geqslant j + (1 + 1/j)\log_2 j - 1$ for $2 \leqslant j < i$. We want to prove $f(T_i^*) \geqslant i + (1 + 1/i)\log_2 i - 1$. Let π_i denote a cutwidth numbering of T_i. We say π_i is good if $P(\pi_i, T_i)$ contains at least 9 vertices of P_n. If π_i is good, then $F(\pi_i, T_i, 1)$ contains the tree which is the union of T_{i-1}^* and an edge incident to the root, denoted by $\tilde{T}_{i-1}$. Consider the restricted mapping π_{i-1} of π_i to $\tilde{T}_{i-1}$. For each j if π_{i-j} is good (i.e., $P(\pi_{i-j}, \tilde{T}_{i-j})$ contains 9 vertices of P_{n-j}), we consider $\tilde{T}_{i-j-1}$ (which is the union of T_{i-j-1}^* and $j+1$ additional edges incident to the root of T_{i-j-1}^*) and the restricted mapping π_{i-j-1} of π_{i-j} to T_{i-j-1}^* until π_{i-j_0} is not good. There are two possibilities.

CASE 1. $j_0 \leqslant i/2 + \log_2 i$ and $j_0 < i$. Since π_{i-j_0} is not good, $F(\pi_{i-j_0}, \tilde{T}_{i-j_0}, 1)$

contains a tree consisting of a path of length 3 joining to three copies of $T^*_{i-j_0-1}$. Thus $F(\pi_{i-j_0},\tilde{T}_{i-j_0},2)$ still contains a copy of $T^*_{i-j_0-1}$. We then have

$$f_{\pi_{i-j_0}}(\tilde{T}_{i-j_0}) \geqslant 2+f(T^*_{i-j_0-1})$$

and, by induction,

$$\begin{aligned} f(T_i^*) = f_{\pi_i}(T_i^*) &\geqslant j_0+f_{\pi_{i-j_0}}(\tilde{T}_{i-j_0}) \\ &\geqslant j_0+2+f(T^*_{i-j_0-1}) \\ &\geqslant j_0+2+i-j_0-1+(1+\frac{2}{i-j_0-1})\log_2(i-j_0-1)-1 \\ &\geqslant 1+i+(1+\frac{2}{i/2-\log_2 i-1})\log_2(\frac{i}{2}-\log i-1)-1 \\ &\geqslant i+(1+\frac{2}{i})\log_2 i-1. \end{aligned}$$

CASE 2. $j_0 > i/2 + \log_2 i$ or $j_0 = i$. Then $f(T_i^*) \geqslant j_0+f_{\pi_{i-j_0}}(\tilde{T}_{i-j_0})$. Note that $\tilde{T}_{i-j_0}$ contains a star S_{i+1} of $i+1$ edges. Thus

$$\begin{aligned} f(T_i^*) &\geqslant j_0 + f(S_{i+1}) \\ &\geqslant j_0 + \lceil\frac{i+1}{2}\rceil \\ &\geqslant i/2 + \log_2 i + \lceil\frac{i+1}{2}\rceil \\ &\geqslant i + \log_2 i + \frac{1}{2}. \end{aligned}$$

Therefore we have proved the following.

THEOREM 3. *For every positive integer n there exists a tree T satisfying*

$$\begin{aligned} &b^*(T) = n \qquad \textit{and} \\ &f(T) \geqslant b^*(T)+\log_2 b^*(T)-1. \end{aligned}$$

5. The difference between the topological bandwidth and the cutwidth for a tree is small. In this section, we will prove that the topological bandwidth for a tree can be bounded above by the sum of its cutwidth and a lower order term. The proof is somewhat complicated. We will give a sequence of observations from which the proof will follow. Suppose π is a bandwidth numbering. Let T' denote a maximal tree in $F(\pi,T,1)$. The numbering induced by π on T' has many special properties. Before we consider these helpful properties we will make some definitions.

Let π denote a numbering of T. We say π is an (x,y)-numbering of T if there is a multi-set $J(T)$ of y vertices (not necessarily distinct) of $V(T)$ such that for any edge $\{u,v\} \in E(T)$ with $\pi(u) < \pi(v)$, we have

$$|\pi(u)-\pi(v)| \leqslant x+|\{w \in J: \pi(u) < \pi(w) < \pi(v)\}|.$$

Furthermore, we say π is derived from a $(x+y,0)$-numbering $\bar{\pi}$ of $\bar{T}$ if π is the induced numbering of $\bar{\pi}$ on T for some $\bar{T}$ containing T. A tree having a (x,y)-numbering is a (x,y)-tree.

OBSERVATION 1. If the bandwidth of a tree T is x, then T is a $(x,0)$-tree.

OBSERVATION 2. Suppose π is a $(x,0)$-numbering of T and π satisfies the leaf property. Let T' denote a maximal tree in $F(\pi,T,1)$. Then T' is a $(x-1,1)$-tree while $J(T')$ is $V(T') \cap P(\pi,T)$.

Proof. For any value a with $1 \leqslant a < a+x \leqslant |V(T)|$ the set $\{u \in V(T): a < \pi(u) \leqslant a+x\}$ contains at least one vertex in $P(\pi,T)$, as does the set $\{u \in V(T): a \leqslant \pi(u) < a+x\}$. Thus the induced numbering π' of π on T' satisfies the property that for $\{u,v\} \in E(T')$ with $\pi(u) < \pi(v)$ we have

$$|\pi'(u)-\pi'(v)| \leqslant x-1+|\{u': \pi(u) < \pi(u') < \pi(v')\} \cap u_0|$$

where $u_0 = V(T') \cap P(\pi,T)$, since $|\{u,v\} \cap P(\pi,T)| \leqslant 1$.

OBSERVATION 3. Suppose T has a $(x,0)$-numbering. Then there is a refinement $\bar{T}$ of T having a $(x,0)$-numbering $\bar{\pi}$ such that for each i and each maximal subtree T' in $F(\bar{\pi},\bar{T},i)$ the induced numbering π' on T' satisfies the leaf property, the monotone property, and the weak block property.

Proof. This follows from the fact that we can untangle the maximal trees.

From now on we will only consider $(x,0)$-numberings satisfying the properties in Observation 3.

OBSERVATION 4. Suppose T has a $(x,0)$-numbering. Then there is a refinement $\bar{T}$ of T having a $(x,0)$-numbering $\bar{\pi}$ such that for each i all the trees T' in $F(\bar{\pi},\bar{T},i)$ are $(x-i,i)$-trees.

Proof. For any value a with $\min_{v \in V(T')}\pi(v) \leqslant a < a+x \leqslant \max_{u \in V(T)}\pi(u)$, the set $\{u \in V(T): a \leqslant \pi(u) < a+x\}$ contains at least one vertex in each basic path $P(\pi,T_j)$, $p \leqslant j \leqslant i$, $T_j \in F(\pi,T,j)$. Thus the induced numbering π' of π of T' satisfies the property that for $\{u,v\} \in E(T')$ with $\pi(u) < \pi(v)$, we have

$$|\pi'(u)-\pi'(v)| \leqslant x-i+|\{u': \pi(u) < \pi(u') < \pi(v)\} \cap J(T')|$$

where $J(T')$ is the multi-set $\bar{\bigcup}_j (V(T') \cap P(\pi,T_j)$ ($\{a\} \bar{\bigcup} \{a\}$ is defined to be $\{a,a\}$).

From now on we will only be interested in the (x,y)-numberings satisfying the leaf property, the monotone property and the weak block property.

OBSERVATION 5. Suppose T is a (x,y)-tree with a (x,y)-numbering π. Let $T_1,T_2,\ldots,T_t$ denote the maximal subtrees in $F(\pi,T,1)$. Then the T_i are $(x-1,y_i+1)$-trees where

$$J(T_i) = (J(T) \cap V(T_i)) \bar{\bigcup} (V(T_i) \cap P(\pi,T)), \quad |J(T_i)| = y_i+1$$

and $\sum_{i=1}^{t} y_i = y$.

We define $f(x,y) = \max\{f(T): T \text{ has an } (x,y)\text{-numbering}\}$.

It is easy to see that $f(x,y)$ is increasing in x and in y. We also write $f(x) = f(x,0)$.

OBSERVATION 6. $f(x,y) \leqslant 1+f(x-1,y+1)$.

Proof. This follows from Observation 5.

OBSERVATION 7. $f(x) \geqslant 1+f(x-1)$.

Proof. Let T be a tree with a $(x-1,0)$-numbering π and $f(T) = f(x-1,0)$. Consider a tree T' which is the union of 3 copies of T and a path P with three vertices adjacent to vertices of T. Obviously $f(T') \geqslant 1+f(T)$. T' is a $(x,0)$-tree since we can form a $(x,0)$-numbering π' on (a refinement of) T' so that for any vertex v in the ith copy of T we have $\pi'(v) = s_{a_i}k_a\pi(v)$ $a_i = i \cdot |V(T)|\lceil a/(a-1)\rceil$ and the vertices in P are numbered by a chain of numbers at most x apart. We then have $f(x) \geqslant f(T') \geqslant 1+f(x-1)$.

OBSERVATION 8. $f(x,1) \leqslant 1+f(x)$.

Proof. Suppose π is a $(x,1)$-numbering for a tree T and $u_0 = J(T)$. Let S consist of all edges $\{u,v\}$ of T such that $\pi(u) < \pi(u_0) < \pi(v)$. If $S = \varnothing$, then T is a $(x,0)$-tree and $f(T) \leqslant f(x)$. Suppose $S \neq \varnothing$. We now choose u_1,v_1,u_2,v_2 (not necessarily distinct) satisfying:

$$\pi(u_1) = \max\{\pi(u): \{u,v\} \in E(T), \ \pi(u) < \pi(u_0) < \pi(v)\},$$
$$\pi(v_1) = \min\{\pi(v): \{u_1,v\} \in E(T), \ \pi(u_0) < \pi(v)\},$$
$$\pi(v_2) = \min\{\pi(v): \{u,v\} \in E(T), \ \pi(u) < \pi(u_0) < \pi(v)\},$$
$$\pi(u_2) = \max\{\pi(u): \{u,v_2\} \in E(T): \ \pi(u) < \pi(u_0)\}.$$

Let $\bar{P}$ denote a path containing u_1,u_2,v_1 and v_2. Any tree T' in the forest F' formed by removing the edges of $\bar{P}$ is a $(x,0)$-tree since for any edge $\{u,v\}$ in $S \cap E(T')$ the set $\{u': \pi(u) < \pi(u') < \pi(v)\}$ must contain at least one vertex in $\{u_1,u_2,v_1,v_2\}-V(T')$. Thus by choosing a numbering with $\bar{P}$ (or its refinement) as the basic path we have

$$f(T) \leqslant 1+\max_{T' \in F'} f(T') \leqslant 1+f(x).$$

OBSERVATION 9. $f(0,y) \leqslant y/2$.

Proof. Suppose a tree T has a $(0,y)$-numbering π. If v is a vertex in $V(T)-J(T)$ and $\{u,v\} \in E(T)$, then $|\pi(u)-\pi(v)| \leqslant |\{w \in J : \pi(u) < \pi(w) < \pi(v)\}| \leqslant |\pi(u)-\pi(v)| - 1$, which is impossible. Thus we can have at most y nontrivial vertices (vertices with degree $\geqslant 1$). By Lemma 4 we have $f(0,y) \leqslant y/2$.

OBSERVATION 10. Suppose π is a $(x,0)$-numbering for T. Suppose T' in $F(\pi,T,i)$ is a $(x-i,j)$-tree, $j \leqslant i$. Then the induced numbering π' of π on T' can be derived from a $(x-i+j,0)$-numbering.

Proof. For $1 \leqslant k \leqslant i$, let T_k be the maximal tree in $F(\pi,T,k)$ containing T'. From the proof of Observation 4 we know that $|\bigcup (P(\pi,T_k) \cap V(T'))| = j' \leqslant j$. Let $\bar{T}$ denote a forest which is the union of j paths and T' such that a vertex in the kth path coincides with the vertex in $P(\pi,T_k) \cap V(T')$ if $P(\pi,\pi_k) \cap V(T') \neq \varnothing$. We can extend $\pi/V(T')$ to $\bar{T}$ and obviously $\bar{T}$ has a $(x-i+j,0)$-numbering.

OBSERVATION 11. Suppose T has a (x,y)-numbering π, and π is derived from a $(x+y,0)$-numbering. Suppose $f(T) > f(x)+1$. Then $y \geqslant x+1$.

Proof. Clearly it holds for $x=1$. Suppose it is true for $x' < x$. Suppose $f(T) > f(x)+1$ and $y \leqslant x$. Since by Observations 6 and 7 $f(T) \leqslant f(x-1,y+1)+1$, and $f(x-1)+1 \leqslant f(x)$, we then have $y \geqslant x-1$. This implies $y = x-1$, or x. From Observation 4 a subtree in $F(\pi,T,1)$ is a $(x-1,y+1)$-tree. Let T_0 denote the maximal subtree in $F(\pi,T,1)$ with the maximum cutwidth.

If T_0 is a $(x-1,y-1)$-tree, by Lemma 5 and Observation 7 we have $1+f(T_0) \geqslant f(T) > 1+f(x) \geqslant 2+f(x-1)$. This implies $y \geqslant x+1$ which is impossible. Thus one of the subtrees is a $(x-1,x+1)$-tree or a $(x-1,x)$-tree, (denoted by T_0) and the rest are $(x-1,1)$-trees (with one exception of a $(x-1,2)$-tree by Observation 5). Clearly the vertex u of T_0 on the basic path $P(\pi,T)$ is in $J(T_0)$. Let $\bar{P}$ denote the path containing the largest number of different vertices in $J(T_0)$. We consider the following three possibilities.

CASE 1. $J(T_0)$ has three or more distinct vertices. Choose a numbering π of a refinement of T_0 so that $\bar{P}$ is the basic path. Suppose $|V(\bar{P}) \cap J(T_0)| \geqslant 3$. Since all

trees in $F(\pi_0,T_0,1)$ are $(x-1,x-1)$-trees, we have $f(T) \leqslant 1+f(T_0) \leqslant 2+f(x-1,x-1) \leqslant 2+f(x-1) \leqslant 1+f(x)$, which is impossible. We may assume $V(P) \cap J(T_0) = \{v_1,v_2\}$. Again each subtree in F_0 can have at most $x-1$ vertices in J since the subtree contains v_i, $i = 1$ or 2, and does not contain any vertex in J. Thus we have

$$f(T) \leqslant 2+f(x-1,x-1) \leqslant 2+f(x-1) \leqslant f(x)+1 .$$

This is a contradiction. Therefore Case 1 cannot happen.

CASE 2. $J(T_0)$ has exactly one vertex i.e., $J(T_0)$ is a multi-set containing u, repeated y times. Let S denote the set of all ordered pairs (u',v') such that $\{u',v'\}$ is an edge and $\pi(u') \leqslant \pi(u) < \pi(v')$. If $S = \varnothing$, then T_0 is a $(x-1,0)$-tree and we have $f(T_0) \leqslant f(x-1)$. Thus $f(T) \leqslant 1+f(x-1) \leqslant 1+f(x)$, which is impossible. We may assume $S \neq \varnothing$. Let $(u',v') \in S$. Since π is derived from a $(x+y,0)$-numbering π', we know that the set $\{v: \pi'(u) < \pi'(v) \leqslant \pi'(u)+x+y\}$ contains at least $y+1$ vertices not in T_0 (one vertex on each basic path). Thus $\pi(v')-\pi(u) \leqslant x-1$. Similarly we can prove $\pi(u)-\pi(u') \leqslant x-1$. Therefore $\pi(v')-\pi(u') \leqslant 2(x-1)$. Thus T_0 is a $(x-1,x-1)$-tree and we have

$$f(T) \leqslant 2+f(x-1) \leqslant 1+f(x) .$$

Again this is a contradiction.

CASE 3. $J(T_0)$ has exactly two vertices, i.e. $J(T_0)$ consists of u, repeated i times and v, repeated $y-i$ times. If both i and $y-i$ are greater than one, the proof is similar to Case 1. If either i or $y-i$ is one, then the proof is similar to Case 2 and will be omitted.

Now we are ready to prove the main theorem.

THEOREM 4. *Suppose a tree T has topological bandwidth $b^*(T) = n$. Then $f(T) \leqslant n+\log_2 n+2$.*

Proof. We will prove by induction on n that $f(T) \leqslant n+\log_2(n-3)+2$ for a tree T with $b^*(T) = n$. It is true for $n \leqslant 4$ since $f(T) \leqslant n+f(0,n) \leqslant 3n/2$ by Observation 9. Let π denote the $(n,0)$-numbering of T. Then maximal subtrees in $F(\pi,T,i)$ are $(n-i,i)$-trees. Let T_i denote the maximal subtree in $F(\pi,T,i)$ with the largest cutwidth. Let z denote the largest integer satisfying

$$f(T_z) \leqslant f(n-z)+1 .$$

From Observation 8 we have $z \geqslant 1$. By definition we have $f(T_{z+1}) > 1+f(n-z-1)$. Using Observation 11 we have $z+1 \geqslant n-z-1$ which implies $z \geqslant n/2 - 1$. From Observation 5, we have

$$\begin{aligned}
f(T) &\leqslant z+f(T_z) \\
&\leqslant z+1+f(n-z) \quad \text{(by definition)} \\
&\leqslant z+1+(n-z)+\log_2(n-z)+2 \quad \text{(by induction)} \\
&\leqslant \frac{n}{2}+n-\frac{n}{2}+1+\log_2(n-\frac{n}{2}+1-3)+2 \quad \text{(because } z \geqslant \frac{n}{2}-1\text{)} \\
&\leqslant n+\log_2(\frac{n}{2}-2)+3 \\
&\leqslant n+\log_2(n-4)+2 \\
&\leqslant n+\log_2(n-3)+2 .
\end{aligned}$$

Thus we have shown that, if $b^*(T) = n$, then

$$f(T) \leqslant n+\log_2(n-3)+2 .$$

This completes the proof of Theorem 4.

REFERENCES

[1] P. Z. CHINN, J. CHVATALOVÁ, A. K. DEWDNEY, and N. E. GIBBS, *The bandwidth problem for graphs and matrices,* J. Graph Theory, 6 (1982), 223-254.

[2] F. R. K. CHUNG, *Some problems and results in labelings of graphs,* The Theory and Applications of Graphs, edited by G. Chartrand, John Wiley and Sons, New York, 1981, pp. 255-263.

[3] M. J. CHUNG, F. MAKEDON, I. H. SUDBOROUGH and J. TURNER, *Polynomial time algorithms for the* MIN CUT *problem on degree restricted trees,* Proc. of the 23rd Annual IEEE Symposium on the Foundations of Computer Science, 1982, pp. 262-271.

[4] M. R. GAREY, D. S. JOHNSON and L. STOCKMEYER, *Some simplified NP-complete graph problems,* Theoret. Comput. Sci., 1 (1976), 237-267.

[5] M. R. GAREY, R. L. GRAHAM, D. S. JOHNSON and D. E. KNUTH, *Complexity results for bandwidth minimization,* SIAM. J. Appl. Math., 34 (1978), 477-495.

[6] M. R. GAREY and D. S. JOHNSON, *Computers and Intractability: A Guide to the Theory of NP-Completeness,* W. H. Freeman, San Francisco (1979).

[7] M. GOLDBERG and I. KLIPKER, *An algorithm of a minimal placing of a tree on the line,* Sakharth, SSR Mech. Acad. Moambe, 83 (1976), 553-556.

[8] F. S. MAKEDON, C. H. PAPADIMITRIOU and I. H. SUDBOROUGH, *Topological bandwidth,* this Journal, 6 (1985), to appear.

[9] C. H. PAPADIMITRIOU, *The NP-completeness of the bandwidth minimization problem,* Computing, 16 (1976), pp. 263-270.

[10] L. J. STOCKMEYER, private communication (1974).

[11] I. H. SUDBOROUGH and F. MAKEDON, private communication (1982).

[12] J. TURNER, private communication (1982).

[13] M. YANNAKAKIS, *A polynominal algorithm for the min cut linear arrangement of trees,* Proc. the 24th Annual IEEE Symposium on Foundations of Computer Science, 1983, pp. 274-281.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 2, April 1985

012

OPTIMUM OVERRELAXATION PARAMETER FOR THE SOR METHOD FOR SOLVING THE EIGENVALUE PROBLEM*

HIDEO SAWAMI† AND HIROSHI NIKI†

Abstract. We study the eigenvalue problem $Ax = \lambda x$, where A is a consistently ordered positive definite matrix. The first eigenvalue of A is obtained with the eigenvector by the SOR method. We first introduce the Jacobi and SOR iteration matrices for the eigenvalue problem, and clarify that the spectral radii, that is the maximum eigenvalues, of both the matrices are unity, but the convergence rate, that is the ratio of the first two eigenvalues in radius, is smaller than unity.

Next, we consider the optimum overrelaxation parameter of the SOR method. The optimum accelerating parameter minimizing the convergence rate is obtained from the first two eigenvalues (in radius) of the Jacobi iteration matrix. Since the eigenvalues are not known a priori, we propose a practical SOR method: in this method, the estimated overrelaxation parameters are used instead of the optimum value.

Finally these results are confirmed by some numerical examples.

1. Introduction.

Using the SOR method, we solve the eigenvalue problem

$$Ax = \lambda x, \tag{1.1}$$

where A is an $n \times n$ consistently ordered positive definite matrix [1], λ is the first eigenvalue and x is the eigenvector corresponding to λ.

For simplicity, we assume that A is rewritten as

$$A = I - L - U, \tag{1.2}$$

where I, L, U are respectively the identity and strictly lower and upper triangular matrices. Our chief interest is in iterative methods for solving the eigenvalue problem; thus we first introduce the Jacobi iteration matrix for the eigenvalue problem $B(\lambda)$ satisfying

$$(I - B(\lambda))x = 0 \tag{1.3}$$

as follows:

$$B(\lambda) = \frac{1}{1-\lambda}(L+U). \tag{1.4}$$

A is a positive definite matrix and we assume that λ_i is the ith eigenvalue of A; that is, $Ax_i = \lambda_i x_i$ where x_i is the eigenvector corresponding to λ_i satisfying

$$0 < \lambda = \lambda_1 < \lambda_2 \leqq \cdots \leqq \lambda_{n-1} < \lambda_n. \tag{1.5}$$

We are thus able to obtain the ith eigenvalue μ_i of $B(\lambda)$ as

$$\mu_i = \frac{1-\lambda_i}{1-\lambda}, \tag{1.6}$$

where $B(\lambda)x_i = \mu_i x_i$, and $1 = \mu_1 > \mu_i$, $i = 2, 3, \cdots, n$.

From these results we have the following lemma.

* Received by the editors July 6, 1983. This work was presented at the SIAM Second Conference on the Applications of Discrete Mathematics held at Massachusetts Institute of Technology, Cambridge, Massachusetts, June 27–29, 1983.

† Department of Applied Mathematics, Okayama University of Science, Ridai-cho 1-1, Okayama-shi, 700 Japan.

LEMMA 1. *The Jacobi iteration for the eigenvalue problem converges if and only if* $\lambda_1 \leqq 2-\lambda_n$.

Proof. The Jacobi iteration for the eigenvalue problem is defined as

$$x(k+1) = B(\lambda)x(k), \tag{1.7}$$

where $x(k)$ is the estimated eigenvector at the kth iteration and we assume that the initial estimate $x(0) = \sum_{i=1}^{n} c_i x_i$, $c_1 \neq 0$. For the case $\lambda_1 < 2-\lambda_n$, we have

$$1 = \mu_1 > |\mu_i|, \qquad i = 2, 3, \cdots, n. \tag{1.8}$$

Accordingly $x(k)$ converges to the first eigenvector $c_1 x_1$ corresponding to the first eigenvalue λ_1.

For the case $\lambda_1 = 2-\lambda_n$, we have

$$1 = \mu_1 = -\mu_n > |\mu_i|, \qquad i = 2, 3, \cdots, n-1. \tag{1.9}$$

In this case, $x(k)$ converges to $c_1 x_1 + (-1)^k c_n x_n$. Therefore the first eigenvector is obtained from $x(k+1)$ and $x(k)$ by addition, and the Jacobi iteration converges in this sense.

Since the Jacobi method for the eigenvalue problem is used, in practice, to obtain both the eigenvector and the eigenvalue, we study the estimation scheme for the eigenvalue. In our Jacobi method for the eigenvalue problem, we use the Rayleigh quotient to estimate the eigenvalue as

$$\lambda(k) = (Ax(k), x(k))/(x(k), x(k)), \tag{1.10}$$

where $\lambda(k)$ is the kth estimate of λ and $(\cdot,\cdot)$ denotes the inner product. From (1.7) and (1.10) we define a practical Jacobi method for the eigenvalue problem. We outline this Jacobi method for the eigenvalue problem in the next section.

2. Jacobi method for the eigenvalue problem. This method is written as follows.

(i) Choose an initial estimate $x(0)$ which contains the eigenvector x_1.

(ii) Compute the Rayleigh quotient $\lambda(k)$ and obtain a new estimate for the eigenvector $x(k+1)$ as

$$x(k+1) = B(\lambda(k))x(k). \tag{2.1}$$

(iii) Repeat (ii) for $k = 0, 1, 2, \cdots$ until convergence.

From this outline we have the following theorem.

THEOREM 1. *The Jacobi method for the eigenvalue problem converges if and only if* $\lambda_1 \leqq 2-\lambda_n$.

Proof. Since λ_1 is the minimum eigenvalue of A, we have $\lambda_1 \leqq \lambda(k)$, and thus

$$\mu_1(k) > |\mu_i(k)|, \qquad i = 2, 3, \cdots, n, \tag{2.2}$$

for the case $\lambda_1 < 2-\lambda_n$, where $\mu_i(k) = (1-\lambda_i)/(1-\lambda(k))$ is the ith eigenvalue of $B(\lambda(k))$. Now assume that $\lambda(k) < 1$, and we also have $\mu_1(k) \geqq 1$. Accordingly $x(k)$ converges to the first eigenvector $c_1 \prod_{m=0}^{k-1} \mu_1(m) x_1$. For the case $\lambda_1 = 2-\lambda_n$, we have

$$\mu_1(k) = -\mu_n(k) > |\mu_i(k)|, \qquad i = 2, 3, \cdots, n-1. \tag{2.3}$$

In this case, $x(k)$ converges to

$$c_1 \prod_{m=0}^{k-1} \mu_1(m) x_1 + c_n \prod_{m=0}^{k-1} \mu_n(m) x_n$$

and $\mu_1(k)$ and $\mu_n(k)$ converge to μ_1 and μ_n respectively. The first eigenvector is obtained from $x(k+1)$ and $x(k)$ by addition.

3. SOR method for the eigenvalue problem. We define the SOR method for the eigenvalue problem as follows.

(i) Choose an initial estimate $x(0)$ which contains the eigenvector x_1 and the overrelaxation parameter ω.

(ii) Compute the Rayleigh quotient $\lambda(k)$ and obtain a new estimate of the eigenvector $x(k+1)$ as

$$x(k+1)=H(\omega,\lambda(k))x(k), \tag{3.1}$$

where $H(\omega,\lambda)$ is the SOR iteration matrix defined as

$$H(\omega,\lambda)=\left(I-\frac{\omega}{1-\lambda}L\right)^{-1}\left[(1-\omega)I+\frac{\omega}{1-\lambda}U\right]. \tag{3.2}$$

(iii) Repeat (ii) for $k=0,1,2,\cdots$ until convergence.

From this outline, we have the following lemma.

LEMMA 2. *The* SOR *iteration for the eigenvalue problem converges if and only if the Jacobi iteration for the eigenvalue problem converges.*

Proof. The SOR iteration uses λ instead of $\lambda(k)$; thus we have

$$(\eta_i+\omega-1)^2=\eta_i(\omega\mu_i)^2, \tag{3.3}$$

where η_i is the ith eigenvalue of $H(\omega,\lambda)$. From (1.9), we find that $\eta_1=1$ and $\eta_n=(1-\omega)^2$ for the case $\mu_1=-\mu_n$. Since $|\mu_i|<1$, $i=2,3,\cdots,n-1$ and $|\eta_i|<1$ for the case $\omega\in(0,2)$, the SOR iteration converges for the case $\omega\in(0,2)$. And if $\mu_1>|\mu_i|$, $i=2,3,\cdots,n$, we find that $1=\eta_1>|\eta_i|$ for the case $\omega\in(0,2)$. Thus the SOR iteration converges if the Jacobi iteration converges. From the above lemma, we have the following theorem.

THEOREM 2. *The optimum overrelaxation parameter* ω_{opt} *of the* SOR *iteration for the eigenvalue problem is given by*

$$\omega_{\text{opt}}=\frac{2}{1+\sqrt{1-\mu_2^2}} \tag{3.4}$$

for the case $\mu_1=-\mu_n=1>|\mu_i|$, $i=2,3,\cdots,n-1$, *or*

$$\omega_{\text{opt}}=\frac{2}{1+\sqrt{1-\mu^2}} \tag{3.5}$$

for the case $\mu_1=1>|\mu_i|$, $i=2,3,\cdots,n$, *where* $\mu^2=\max_i\mu_i^2$.

Proof. This theorem is easily obtained. Thus we only remark that the above results are obtained from the assumption $c_i\neq 0$, $i=1,2,\cdots,n$, and that the optimum overrelaxation parameter becomes smaller than the value of (3.4) or (3.5) if some c_i vanish.

After some tedious computation, we also have the following theorem.

THEOREM 3. *The* SOR *method for the eigenvalue problem converges if and only if* $\omega\in(0,2)$.

Proof. It is easily found that the first eigenvalue $\eta_1(k)>|\eta_i(k)|$ for all $i=2,3,\cdots,n$ since $\mu_1(k)>1$, and that the convergence of the $x(k)$ to the first eigenvector requires the overrelaxation parameter $\omega\in(0,2)$. Indeed, for arbitrary $\omega\notin(0,2)$, the convergence of $x(k)$ to the first eigenvector becomes slower if the estimate $\lambda(k)$ becomes closer to λ_1.

We remark that the first eigenvalue η_1 of $H(\omega,\lambda)$ is unity, the eigenvalue $\eta_1(k)$ of $H(\omega,\lambda(k))$ is greater than or equal to unity and the optimum overrelaxation parameter ω_{opt} minimizes the second eigenvalue η_2 or $H(\omega,\lambda)$. We also remark that

the convergence rate $|\eta_2/\eta_1| = \omega_{\text{opt}} - 1$ for the case $\mu_1 = -\mu_n$. Thus the Jacobi method is improved by the factor $\log_{10}(\omega_{\text{opt}} - 1)/\log_{10}(\mu_2)$ for the iteration number; this means that the SOR method is $2\sqrt{2}/\sqrt{\varepsilon}$-times faster than the Jacobi method for small values of $\varepsilon > 0$, where $\mu_2 = 1 - \varepsilon$, and $\sqrt{2}/\sqrt{\varepsilon}$-times faster than the Gauss–Seidel method.

4. Numerical examples. We define the optimum SOR method for the eigenvalue problem as follows.

(i) Choose an initial estimate $x(0)$ which contains the first eigenvector.

(ii) Compute the Rayleigh quotient $\lambda(k)$ $(k = 0, 1, 2, \cdots)$ and obtain a new estimate $x(k+1)$ using $H(\omega_{\text{opt}}, \lambda(k))$.

(iii) Repeat (ii) until convergence.

For practical use, we propose a technique [2] for solving the eigenvalue problem as follows.

(i) Choose an initial estimate $x(0, \omega_0)$ which contains the first eigenvector, and set the initial overrelaxation parameter ω_0 to unity.

(ii) Compute the ratio $\lambda(d, \omega_m) = \|x(d, \omega_m) - x(d-1, \omega_m)\| / \|x(d-1, \omega_m) - x(d-2, \omega_m)\|$ at the dth step using the mth overrelaxation parameter ω_m, where the SOR iteration is carried out until the following conditions are satisfied:

$$\lambda(j, \omega_m) < 1 \tag{4.1}$$

and

$$0 < [\lambda(d-1, \omega_m) - \lambda(d, \omega_m)]/[\lambda(d-2, \omega_m) - \lambda(d-1, \omega_m)] < 1 \tag{4.2}$$

for $j = d-2, d-1, d$ and $d \leqq 4$. We denote by $x(d, \omega_m)$ the estimated eigenvector at the dth iteration using the mth overrelaxation parameter ω_m.

(iii) The new overrelaxation parameter ω_{m+1} is determined as follows:

$$\omega_{m+1} = \lambda(d, \omega_m) + 1 \quad \text{if } \omega_m - 1 \geqq \lambda(d, \omega_m), \tag{4.3}$$

otherwise

$$\omega_{m+1} = [\omega_m + 2/(1 + \sqrt{1 - \mu^2})]/2, \tag{4.4}$$

where

$$\mu^2 = [\omega_m + \lambda(d, \omega_m) - 1]^2/[\lambda(d, \omega_m)\omega_m^2]. \tag{4.5}$$

(iv) Repeat from (ii) to (iii) until convergence.

The Gauss–Seidel method for the eigenvalue problem is defined like the optimum SOR method, using unity instead of the optimum overrelaxation parameter ω_{opt}.

Using the above three methods, we have solved the following eigenvalue problem as an example:

$$-\Delta\phi = \lambda\phi \quad \text{in } \Omega = (0, 1) \times (0, 1), \tag{4.6}$$

with the boundary condition $\phi = 0$ on $\partial\Omega$, where Δ is the two-dimensional Laplacian operator. The unit square domain $\bar{\Omega}$ is divided into a square mesh with mesh size h, and (4.6) is approximated by finite differencing with a five-point stencil. The results are tabulated in Table 1 for $h^{-1} = 4, 8, 16, 32, 64, 128$. The initial estimates $x(0)$ or $x(0, \omega_0)$ are taken to be 1 on the rectangular domain, the upper half of $\bar{\Omega}$; the rest are taken to be zero, and contain the first and the second eigenvectors. By doing so we have $c_1 \neq 0$ and $c_2 \neq 0$, and the improvement on the convergence are correctly demonstrated. Pagewise ordering is used for all of the methods. We remark that some of the iteration matrices have nonlinear elementary divisors, but since nonlinear

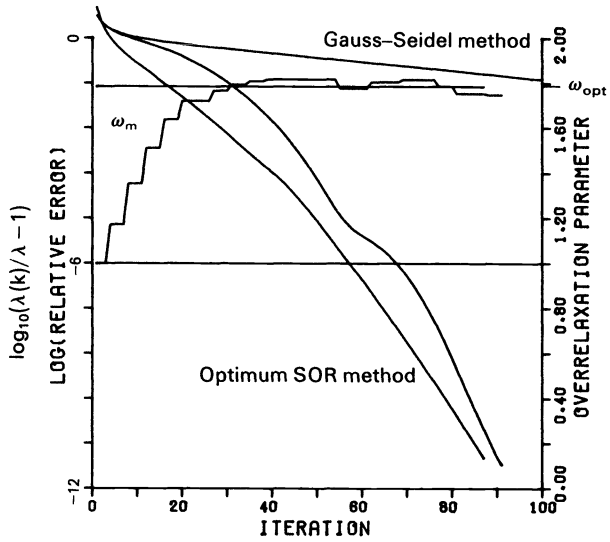


FIG. 1. *The relative error,* $\log_{10}(\lambda(k)/\lambda - 1)$ *and the accelerating parameter for the Gauss–Seidel, optimum* SOR *and our methods drawn versus the iteration number. The mesh size is* $h = \frac{1}{32}$.

TABLE 1

Iteration numbers required to satisfy the convergence test for the Gauss–Seidel, optimum SOR *and our methods with the values of* h^{-1}, μ_2 *and* ω_{opt}

h^{-1}	4	8	16	32	64	128
μ_2	0.5	0.88268	0.97099	0.99276	0.99819	0.99955
ω_{opt}	1.072	1.361	1.614	1.786	1.887	1.942
Gauss–Seidel	11	48	200	808	3,238	12,961
Optimum SOR	10	20	41	81	163	328
Our method	11	24	46	86	183	385

elementary divisors are not our purpose in this paper, we reduce their influence by the following convergence test:

$$|\lambda(k) - \lambda|/\lambda \leqq 10^{-10}. \tag{4.7}$$

As seen in Table 1, we found that the SOR method using the optimum accelerating parameter is by far the best; the improvement is drastic for small h. We found that our method using estimated overrelaxation parameters is very useful since the optimum accelerating parameter is in general unknown for the eigenvalue problem.

5. Conclusion. We have introduced the Jacobi, the SOR and our methods for the eigenvalue problem and show using the improvement factor and numerical examples that the convergence of the optimum SOR method is very best. Since the optimum overrelaxation parameter is obtained from the two of the eigenvalues and these values are not known a priori, we conclude that our estimation method for this parameter is useful in practice for solving the eigenvalue problem.

REFERENCES

[1] D. M. YOUNG, *Iterative Solution of Large Linear Systems,* Academic Press. New York, 1971.

[2] H. NIKI, H. SAWAMI AND T. ONDOH, *A note on a practical technique for solving elliptic equations,* J. Inst. Maths. Appl., 22 (1978), pp. 353–359.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 2, April 1985

013

MONOTONICITY AND OTHER PARADOXES IN SOME PROPORTIONAL REPRESENTATION SCHEMES*

EDWARD M. BOLGER†

Abstract. Single Transferable Voting, which does provide proportional representation, has recently been shown to exhibit certain paradoxes [2], [3], [5], [6], [7]. In this paper we examine some alternate methods designed to provide proportional representation while at the same time reducing the number of "wasted votes". Several versions of cumulative voting with transfer of surplus exhibit some forms of the monotonicity, No-Show, and New Voter paradoxes. If, in addition, low-ranking candidates are eliminated to further reduce wasted votes, then even the simplest forms of the monotonicity principle are violated.

Key words. proportional representation, voting paradoxes, transfer of surplus, elimination

AMS(MOS) subject classifications. 92, 90A

1. Introduction. A widely used scheme to provide proportional representation is the Hare Voting System, also known as Single Transferable Voting (STV). STV has recently come under attack because it exhibits certain paradoxes. Doron and Kronick [5] have given an example to show that STV exhibits the monotonicity (a winning candidate "moves up" in some of the rankings and becomes a losing candidate) paradox. Doron [6] has given an example in which the results change when a losing candidate is removed. Brams and Fishburn [2] demonstrate that STV also violates the "No-Show" or "New Voters" principle (i.e. if new voters enter the election and place x last in their rankings, the election process should not replace any winning candidate with x). For more results on paradoxes of preferential voting schemes, see Doron [7], Brams and Fishburn [2], Brams [3], Fishburn [8], and Smith [10].

One reason for the popularity of STV as a proportional representation scheme is that it usually results in few wasted votes. Cumulative voting, another popular proportional representation scheme, does not share this property. Votes may be "wasted" either on a candidate who receives far more votes than needed or on a candidate who has no chance of being elected.

It is tempting to modify cumulative voting by eliminating low-ranking candidates and by transferring "surplus" votes from candidates who have more votes than needed. We shall first consider a restricted form of cumulative voting in which voters distribute their votes equally among their candidates and we shall allow "transfer of surplus" in this voting system.

2. Multiple Voting with Transfer of Surplus (MVTS). MVTS is a proportional representation scheme described as follows.

There are n voters and m candidates. The number to be elected is e. Let $Q=(ne+1)/(e+1)$. Q is called the *quota*. Each voter selects a subset consisting of e or fewer candidates (thus choosing those the voter wishes to support). If the voter selects c candidates ($c \leqq e$), then on the first round each of these candidates receives e/c votes from this voter. The election procedure consists of the following four steps:

Step 1. Each candidate receiving a total of at least Q votes is declared elected. If there are e such candidates, the process terminates. If no candidate has more than

* Received by the editors July 6, 1983. This work was presented at the SIAM Second Conference on the Applications of Discrete Mathematics held at Massachusetts Institute of Technology, Cambridge, Massachusetts, June 27–29, 1983.

† Department of Mathematics and Statistics, Miami University, Oxford, Ohio 45056.

Q votes, go to Step 4. Let $v_i(A)$ be the number of votes currently assigned to candidate A by voter i. Let $v(A)$ be the total number of votes currently assigned to A. If $v(A) > Q$, then $v(A) - Q$ is called the surplus for candidate A. In this case, $[v(A) - Q]v_i(A)/v(A)$ is called voter i's share of this surplus.

Step 2. If $v(A) > Q$, distribute voter i's share of A's surplus equally among those of voter i's candidates who have not been declared elected. Thus, if B is one of i's candidates and if $v(B) < Q$, then after the transfer of A's surplus votes, voter i will assign to B

$$v_i(B) + \frac{1}{c_i}\frac{v_i(A)}{v(A)}[v(A) - Q]$$

votes, where c_i is the number of nonelected candidates supported by voter i. After distributing all such surpluses, set $v(A) = Q$ for each elected candidate A.

Step 3. For each nonelected candidate B, compute the new vote total, $v(B)$, and go back to step 1.

Step 4. If after all surpluses have been distributed, the number, say x, of elected candidates is less than e, then declare elected the remaining $e - x$ candidates with the highest vote totals. In case of a tie, include none of the "tied" candidates among the winners (in which case fewer than e candidates will be elected).

For example, suppose there are 25 voters to elect two of four candidates. Suppose the first 16 voters vote for A and B; the next 4 voters vote for A and C; the remaining voters vote for C and D. On the first count, A is declared elected and has 3 surplus votes. Each of the first 16 voters transfers $\frac{3}{20}$ votes to candidate B who will then have enough votes to make quota.

For the proof that MVTS does indeed guarantee proportional representation, see Bolger [1].

In order to conduct an election using MVTS, one needs a list which contains the name of each voter and the candidates this voter chooses to support. Such a list will be called a *voter profile.* If π is a voter profile, then $\mathcal{W}(\pi)$ denotes the set of winning candidates under π and $v^{\pi}(A)$ is the (final) number of votes assigned to A under π. If $\mathcal{B}$ is a subset of the set of candidates, then $\pi(\mathcal{B})$ will denote the number of voters who support those and only those candidates in $\mathcal{B}$.

3. Paradoxes of MVTS. We consider first the "New Voter" paradox.

DEFINITION. Suppose the voter profile π^+ is obtained from the profile π when one new voter is added and votes only for candidates who won under π. MVTS will be said to *violate the New Voter principle* if one of this new voter's candidates loses under π^+.

Let Q^+ be the quota under π^+. Then

$$Q^+ = Q + \frac{e}{e+1}.$$

Thus, it is clear that if candidate A made quota on the first count under π and if the new voter supports A, then A makes quota under π^+ (since A receives e/c votes from the new voter and $c \leqq e$).

LEMMA. *Suppose the new voter votes for X and k other candidates $C_1, C_2, \cdots, C_k$ where $X, C_1, C_2, \cdots, C_k$ all won under π. Then X can lose under π^+ only if*

$$k^2 - k(e-1) + 1 < 0.$$

Proof. Each of $X, C_1, \cdots, C_k$ receives on the first count $e/(k+1)$ additional votes. Since the quota increases by $e/(e+1)$ votes, the potential surplus to each of

$C_1, \cdots, C_k$ increases by $e/(k+1)-e/(e+1)$. If X does not make quota under π^+, the maximum number of additional votes some candidate other than X, $C_1, \cdots, C_k$ could receive is less than

$$k\left(\frac{e}{k+1}-\frac{e}{e+1}\right)$$

votes. On the other hand, X might lose some votes which had been transferred to X under π from some other winning candidates. Thus, the net gain to X is at least

$$\frac{e}{k+1}-[e-(k+1)]\frac{e}{e+1}$$

votes. Thus, if X loses,

$$k\left(\frac{e}{k+1}-\frac{e}{e+1}\right)>\frac{e}{k+1}-[e-(k+1)]\frac{e}{e+1}.$$

The result follows immediately.

Using the lemma, it is clear that if the new voter votes only for X, then X will still win. Moreover, by checking directly the relevant values of k and e, one can immediately prove:

THEOREM. MVTS *does not violate the New Voter principle if* $e \leqq 3$.

Unfortunately, MVTS does violate this principle for $e=4$.

Example. Let $e=4$ and $n=130$. Let $\pi(C_1, C_2, Y)=72$, $\pi(C_1, C_2, C_3)=6$, $\pi(C_3, X)=49$, $\pi(C_1, C_2, X)=2$, $\pi(C_1, C_2, C_3, Y)=1$. The final vote totals are: $v(C_1)=v(C_2)=v(C_3)=104.2$, $v(X)=103.40$, $v(Y)=103.27$. Suppose a new voter enters and votes for C_1, C_2, X. Now, Y beats X by a vote of 104.138 to 104.125.

We shall next consider situations in which either a voter or a candidate is removed from the voter profile.

DEFINITION. Suppose the voter profile π^- is obtained from the voter profile π when one voter who voted only for losing candidates decides not to vote. MVTS will be said to *violate the "No-Show" principle* if $\mathcal{W}(\pi) \neq \mathcal{W}(\pi^-)$.

Let Q^- be the quota under π^-. Then

$$Q^- = Q-\frac{e}{e+1}.$$

Here, too, it is clear that a candidate who made quota under π will still make quota under π^-. However, MVTS violates the No-Show principle even for $e=2$.

Example. $\pi(C_1)=2$, $\pi(C_2)=10$, $\pi(C_5)=1$, $\pi(C_1, C_3)=20$, $\pi(C_2, C_4)=1$. With $e=2$ and $n=34$, we have $Q=23$. The final vote counts give $v(C_1)=23$, $v(C_2)=21$, $v(C_3)=20+\frac{5}{6}$, $v(C_4)=1$, $v(C_5)=2$. Thus, $C_1, C_2 \in \mathcal{W}(\pi)$. If the voter who supported candidate C_5 drops out, then the new final vote counts are $22+\frac{1}{3}$, 21, $21+\frac{7}{18}$, 1 so that the new winners are C_1 and C_3.

Note that the vote totals for C_2 and C_3 were very close. The next theorem indicates that if MVTS violates the No-Show principle for $e=2$, then C_2 and C_3 differ by less than $\frac{2}{3}$ votes.

THEOREM. *If, for* $e=2$, *the voter profile* π *provides an example of the "No-Show" paradox with* $C_1, C_2 \in \mathcal{W}(\pi)$ *and* $C_1, C_3 \in \mathcal{W}(\pi^-)$, *then*

(i) C_1 *makes quota under* π^-

and

(ii) $v^\pi(C_2) < v^\pi(C_3)+\frac{2}{3}$.

Proof. The only way for C_3 to get more votes under π^- is to have more votes transferred from C_1 to C_3. The maximum number of such additional surplus votes is $\frac{2}{3}$.

COROLLARY. *Under the conditions of the above theorem, C_1 must have made quota under π.*

Proof. If C_1 did not make quota under π, then no votes were transferred. As a result, $v^{\pi}(C_2) \geqq 1 + v^{\pi}(C_3)$.

Another type of reduction paradox can occur when a losing candidate is removed. With ranked voting lists, it is usually assumed that the candidates ranked below the removed candidate all move up one place in the rankings. With MVTS and cumulative voting, there are several ways to proceed. For instance, one may assume that votes assigned to a removed candidate are also removed, and that the quota is lowered. Or, one may assume that each voter would have transferred his or her votes to the other candidates supported by this voter. In the following example, a losing candidate is removed and all the voters who voted for the removed candidate shift all of their votes to their other candidates, each of whom won under π.

Example. Suppose $e = 2$ and $n = 89$. Let $\pi(C_1) = 15$, $\pi(C_2) = 25$, $\pi(C_2, C_4) = 1$, $\pi(C_3) = 17$, $\pi(C_1, C_3) = 16$, $\pi(C_1, C_4) = 15$. The final vote totals are $v(C_1) = 59.67$, $v(C_2) = 51$, $v(C_3) = 50.35$, $v(C_4) = 16.33$. If C_4 is removed and if $\pi^-(C_1) = 30$, $\pi^-(C_2) = 26$, $\pi^-(C_3) = 17$ and $\pi^-(C_1, C_3) = 16$, then the final vote totals are 59.67, 52 and 53.44 so that C_1 and C_3 now win.

We turn now to discussions of monotonicity.

DEFINITION. Let $X \in \mathcal{W}(\pi)$. Suppose π' is obtained from π when one voter who voted under π now changes his or her mind and votes only for X. The election procedure is said to *violate monotonicity* if $X \notin \mathcal{W}(\pi')$.

THEOREM. MVTS *does not violate monotonicity.*

Proof. Let $X \in \mathcal{W}(\pi)$. If voter i had voted for X and r other candidates $C_1, C_2, \cdots, C_r$ and if voter i now votes only for X, then, on the first count, X will receive an additional $re/(r+1)$ votes. If voter i had not originally voted for X, then X will receive e more votes. However, X may lose some of the surpluses (if any) that the other voters had transferred from $C_1, C_2, \cdots, C_r$ to X. These combined losses cannot exceed the total number of votes taken from $C_1, C_2, \cdots, C_r$.

DEFINITION. Let $X \in \mathcal{W}(\pi)$. Suppose π' is obtained from π when one voter changes his or her mind and votes for X and k other candidates, each of whom belongs to $\mathcal{W}(\pi)$. If this voter had either not voted for X or had voted for X as well as at least $k+1$ other candidates, then the election procedure is said to *violate k-monotonicity* if $X \notin \mathcal{W}(\pi')$.

Since 0-monotonicity is the same as monotonicity, we need only consider values of k for which $1 \leqq k \leqq e-1$.

THEOREM. MVTS *is 1-monotonic for* $e = 2$.

Proof. Suppose voter i had not voted for X and now votes for X and C_1, each of whom belonged to $\mathcal{W}(\pi)$. X gains 1 vote on the first count. X may lose almost 1 vote if voter i had previously voted only for C_1 (since C_1 loses one vote on the first count and has one less vote to transfer to other candidates). In this case, no other candidate can get more votes unless X makes quota under π'. On the other hand, if voter i had not previously voted for C_1, then there may be one additional surplus vote, almost all of which could be transferred to some other candidate. But in this case X gains more than one vote since X gets (from voter i) some of this extra surplus vote.

Unfortunately, MVTS is neither 1-monotonic nor 2-monotonic when $e = 3$.

Example. $\pi(C_1, Y) = 60 = \pi(C_2, X), \pi(C_2) = 9, \pi(C_1, C_2, Y) = 9, \pi(C_1, C_2) = 4, \pi(C_1) = 3$. The final vote totals are: $v(C_1) = 109 = v(C_2)$, $v(X) = 105.68$, $v(Y) = 104.91$. Suppose one voter switches from C_2 to C_1 and X. Y now beats X by a vote of 105.97 to 105.54.

Example. $\pi(C_1, C_3, X) = 1$, $\pi(C_1, C_2, Y) = 72$, $\pi(X) = 24 = \pi(C_1, C_2)$, $\pi(C_4) = 22$, $\pi(C_1, C_2, X) = 1 = \pi(C_1, C_2, C_3)$. The final vote totals are $v(C_1) = v(C_2) = 109$, $v(C_3) = 2.04$, $v(C_4) = 66$, $v(X) = 74.04$, $v(Y) = 73.95$. Suppose one voter switches from C_4 to C_1, C_2, X. Now Y beats X by a vote of 75.22 to 75.10.

4. Elimination of low-ranking candidates. Even with transfer of surplus, many votes may still be wasted on losing candidates. Indeed, in the famous "Sharpsville Railroad" example (see Brams [4] and Glasser [9]), the majority wastes many of its votes by spreading them among too many candidates.

We shall modify MVTS as follows. If after all transfers of surplus votes, fewer than e candidates have made quota, then the candidate with the lowest vote total shall be eliminated. (In case of a tie, the candidate to be eliminated shall be chosen at random.) The votes currently assigned by voter i to the eliminated candidate shall be distributed equally among voter i's remaining candidates (those either not elected or not eliminated). The process continues (including transfer of surplus created by elimination of candidates) until either e candidates have made quota or until k candidates have made quota and only $e-k$ candidates remain. This election procedure will be called Multiple Voting with Transfer of surplus and Elimination of low-ranking candidates (MVTE).

In view of Fishburn's [8] results on the nonmonotonicity of elimination schemes for ranked lists, one would expect MVTE to violate monotonicity even in the simplest case where $e = 2$ and some voters change their minds and vote only for candidate X who was a winner under π.

Example. Let $e = 2$ and $n = 448$ so that $Q = 299$. Let $\pi(C_1) = 148$, $\pi(X, Y) = 90$, $\pi(C_2, Y) = 88$, $\pi(X) = 55$, $\pi(C_2) = 56$, $\pi(Y) = 10$, $\pi(C_1, Y) = 1$. The first count yields

C_1	C_2	X	Y
297	200	200	199

so that Y is eliminated. The second count yields

C_1	C_2	X
298	288	290

so that C_1 and X win. Suppose now that one voter who voted only for C_2 now votes only for X. The new results are:

C_1	C_2	X	Y
297	198	202	199
297		202	287

Now, C_1 and Y win.

Recall that if MVTS violated the No-Show principle for $e=2$, then the affected candidates differed by less than $\frac{2}{3}$ votes. This is not true for MVTE. In the following example, X wins by more than 7 votes under π but loses by more than 6 votes under π^-.

Example. $\pi(C_1, X)=1$, $\pi(C_1)=30$, $\pi(X, Y)=15$, $\pi(C_2, Y)=9$, $\pi(C_1, Y)=40$, $\pi(X)=25$, $\pi(C_2)=28$. The election proceeds as follows:

C_1	C_2	X	Y
101	65	66	64
99	65	66.02	64.79
99	74	81.02	—
99	—	81.02	—

Suppose now that one voter who voted for C_2 and Y drops out. Then we get

C_1	C_2	X	Y
101	64	66	63
98.33	64	66.03	64.05
98.33	—	66.03	72.05
98.33	—	—	87.05

The first violation of the New Voter principle occurred for $e=4$ under MVTS. With MVTE the first such violation occurs with $e=2$.

Example. $\pi(C_1, X)=2$, $\pi(C_1)=30$, $\pi(X, Y)=15$, $\pi(C_2, Y)=9$, $\pi(C_1, Y)=40$, $\pi(X)=25$, $\pi(C_2)=28$. C_1 and $X \in \mathcal{W}(\pi)$. If a new voter enters and votes for C_1 and X, then C_1 and Y now win.

5. Cumulative Voting with Transfer of Surplus. An alternate scheme to reduce the number of wasted votes without using an elimination procedure is Cumulative Voting with Transfer of Surplus (CVTS). (See Bolger [1].) This scheme is similar to MVTS except that each voter may distribute e votes among e or fewer candidates in any way the voter chooses. For example, if $e=7$, a voter may assign 4 votes to one candidate and 1.5 votes to each of two other candidates. Moreover, if $v(A)>Q$, distribute voter i's share of A's surplus "proportionately" among those of voter i's candidates who have not been declared elected. That is, if v_i' is the number of votes cast by voter i for his or her nonelected candidates, then transfer to candidate B

$$\frac{v_i(B)}{v_i'} \cdot \frac{v_i(A)}{v(A)} \cdot [v(A)-Q]$$

votes.

Example. Let $e=6$ and $n=99$ so that $Q=85$. A group of 49 voters could assign 85 votes to each of three candidates and 39 votes to a fourth candidate, thus guaranteeing the election of three candidates while having a chance of electing a fourth candidate who might be assigned some votes by the other voters or who might pick up some surplus votes if some of the other voters vote for one or more of the group's candidates who made quota. With MVTS, a group of 49 voters could only support 3 candidates if it wished to guarantee the election of 3 candidates. In this case, some votes would definitely be wasted.

Unfortunately, CVTS may violate the New Voter principle for $e \geqq 2$ and k-monotonicity for $e \geqq 2$.

Example. Let $e = 2$ and $n = 92$ so $Q = 61.67$.

Number of voters	Number of votes assigned
60	60 votes to each of C_1, Y
30	60 votes to X
1	2 votes to C_1
1	0.5 votes to X, 1.5 votes to C_2

Results:

C_1	C_2	X	Y
62	1.5	60.5	60
61.67	1.5	60.5	60.32

Winners: C_1, X.

Suppose one new voter enters and casts 1.9 votes for C_1 and 0.1 votes for X.

Results:

C_1	C_2	X	Y
63.9	1.5	60.6	60
62.33	1.5	60.6	61.5

Winners: C_1, Y.

Example. Let π be the original profile in the above example. Suppose that the last voter (who had voted for C_2 and X) changes and casts 1 vote for each of C_1 and X.

Results:

C_1	X	Y
63	61	60
61.67	61.02	61.27

Winners: C_1, Y.

If the avoidance of paradoxes is more important than wasted votes, then MVTS is preferable to CVTS whereas if reducing the number of wasted votes is more important; then CVTS may be preferable.

6. Unlimited Voting with Transfer of Surplus. It is natural to ask the question: "Why restrict the number of candidates each voter may support?" If each voter is given e votes to distribute, why not allow this voter to vote for more than e candidates?

UVTS is similar to MVTS except that a voter may support as many candidates as desired. If a voter supports c candidates, then each of these candidates receives e/c votes. Since we may have c greater than e, the ratio e/c may be less than 1.

For example, if $e = 1$ and if there are 30 voters who cannot decide between two candidates of whom they each approve, then with UVTS these 30 voters may assign 15 votes to each candidate.

Unlike MVTS, UVTS violates 1-monotonicity for $e = 2$.

Example. Let $e = 2$ and $n = 92$.

Number of voters	Number of votes assigned
60	60 votes to each of C_1 and Y
30	60 votes to X
1	2 votes to C_1
1	0.5 votes to each of C_2, C_3, C_4, X

Results:

C_1	C_2	C_3	C_4	X	Y
62	0.5	0.5	0.5	60.5	60
61.67	0.5	0.5	0.5	60.5	60.32

Winners: C_1, X.

Suppose now the last voter changes to C_1 and X. Then the results are:

C_1	X	Y
63	61	60
61.67	61.02	61.27

Winners: C_1, Y.

THEOREM. UVTS *does not violate the New Voter principle for* $e \leqq 3$.

Proof. Same proof as for MVTS.

If we add the elimination of low-ranking candidates of UVTS, then the resulting scheme shall be called Unlimited Voting with Transfer of surplus and Elimination of low-ranking candidates (UVTE). As with MVTE, UVTE violates the New Voter principle if $e \geqq 2$. However, UVTE violates the monotonicity and No-Show principles even for $e = 1$.

Example.

Number of voters	Candidates	Number of votes to each candidate
91	X, Y	45.5
88	Y, Z	44
55	X	55
56	Z	56
10	Y	10

Results:

X	Y	Z
100.5	99.5	100
145.5	—	144

Winner: X.

Suppose now that one "Z only" voter switches to X. Now the results are:

X	Y	Z
101.5	99.5	99
101.5	143.5	—

Winner: Y.

Example:. Let π be the original voter profile in the above example and suppose one "Z only" voter drops out. Then Z will be eliminated and Y will win.

7. Summary. Table 1 summarizes the results on paradoxes of the voting systems considered.

Note. CVTE is CVTS with elimination of low-ranking candidates and "proportional" distribution of the eliminated candidate's votes. As far as the paradoxes are concerned, CVTE is similar to MVTE.

TABLE 1
Values of e for which the paradoxes occur.

	Paradoxes			
	New Voter	Monotonicity	k-Monotonicity	No-Show
MVTS	$e \geqq 4$	—	$e \geqq 3$	$e \geqq 2$
MVTE	$e \geqq 2$	$e \geqq 2$	$e \geqq 2$	$e \geqq 2$
CVTS	$e \geqq 2$	—	$e \geqq 2$	$e \geqq 2$
CVTE	$e \geqq 2$	$e \geqq 2$	$e \geqq 2$	$e \geqq 2$
UVTS	$e \geqq 4$	—	$e \geqq 2$	$e \geqq 2$
UVTE	$e \geqq 2$	$e \geqq 1$	$e \geqq 1$	$e \geqq 1$

REFERENCES

[1] E. M. BOLGER, *Proportional representation*, in Political and Related Models, Brams, Lucas, Straffin eds., Modules in Applied Mathematics 2, Springer-Verlag, New York, 1982.

[2] S. J. BRAMS AND P. C. FISHBURN, *Paradoxes of preferential voting*, mimeographed manuscript, New York Univ. and Bell Telephone Laboratories, 1982.

[3] S. J. BRAMS, *The* AMS *nomination procedure is vulnerable to "truncation of preferences"*, Notices of the American Mathematical Society, 29 (1979), pp. 136–138.

[4] ———, *Game Theory and Politics*, Free Press, New York, 1975.

[5] G. DORON AND R. KRONICK, *Single transferable vote: an example of a perverse social choice function*, Amer. J. Polit. Sci., 21 (1977), pp. 303–311.

[6] G. DORON, *Is the Hare voting scheme representative*?, J. Politics, 41 (1979), pp. 918–922.

[7] ———, *The Hare voting system is inconsistent*, Political Studies, 27 (1979), pp. 283–286.

[8] P. C. FISHBURN, *Monotonicity paradoxes in the theory of elections*, mimeographed manuscript, Bell Telephone Laboratories, Murray Hill, NJ, 1982.

[9] G. J. GLASSER, *Game theory and cumulative voting for corporate directors*, Management Sci., 5 (1959), pp. 151–156.

[10] J. H. SMITH, *Aggregation of preferences with variable electorate*, Econometrica, 41 (1973), pp. 1027–1041.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 2, April 1985

014

A METHOD TO COMPUTE MINIMAL POLYNOMIALS*

BARBARA R. PESKIN† AND DAVID R. RICHMAN‡

Abstract. Let $f(X)$ and $g(X)$ be polynomials with coefficients in an arbitrary field K. Assume that $f(X)$ is irreducible and let r be a root of $f(X)$. We describe a new algorithm for computing the minimal polynomial of $g(r)$ over K. The novelty of our algorithm is that it begins by computing the polynomial $p(X, Y)$ of smallest degree such that $p(f, g) = 0$.

Key words. minimal polynomial, irreducible polynomial, finite field

AMS(MOS) subject classifications. primary 68C20; secondary 16-04, 68C25

1. Introduction. Let $f(X)$ and $g(X)$ be polynomials with coefficients in the field K. Assume that $f(X)$ is irreducible and let r be a root of $f(X)$. Thus $f(X)$ is the minimal polynomial of r over K. We are interested in computing the minimal polynomial of $g(r)$. A survey of methods to perform this computation can be found in [5, pp. 112–117]. Minimal polynomial computations are also described in [4, pp. 142–143], [7] and [9]. Marsh [12] has used minimal polynomial computations to generate tables of irreducible polynomials over $GF(2)$. Tables of this type are important for generating cyclic codes; see [5, pp. 150–153] and [14, pp. 206–211]. They are also important for generating linear switching circuits; see [8] and [14, Chap. 7].

The purpose of this paper is to describe a new method to compute the minimal polynomial of $g(r)$. This method grew out of work by the first author while investigating certain group actions on power series. The novelty of our method is that it begins by computing the polynomial $p(X, Y)$ of minimal degree such that $p(f(X), g(X)) = 0$.

Our algorithm is easiest to explain when the degrees of $f(X)$ and $g(X)$ are relatively prime. In § 2 we describe the algorithm in that case, and in § 3 we prove that it works. In § 4 we describe the algorithm in general. We also describe how to determine, given three polynomials h, f, g in $K[X]$, whether h lies in $K[f, g]$. In § 5 we estimate the speed of the algorithm when the degrees are relatively prime. We compare our algorithm with a minimal polynomial algorithm which is commonly implemented and provide evidence that our algorithm is faster when deg g is small compared to deg f.

2. The algorithm when the degrees are relatively prime. Let f and g be monic polynomials such that the degree of f is relatively prime to the degree of g. Compute a sequence of polynomials $R_1, R_2, \cdots$ as follows.

Step 1.

Set $R_1 = g^{\deg f} - f^{\deg g}$. Proceed to step 2.

Step $i+1$.

If $R_i = 0$, stop. Otherwise, as will be shown in the next section, we may choose a scalar c_i and nonnegative integers p_i, q_i so that

$$\text{degree } (R_i - c_i f^{p_i} g^{q_i}) < \text{degree } R_i. \tag{1}$$

Set $R_{i+1} = R_i - c_i f^{p_i} g^{q_i}$ and proceed to the next step.

* Received by the editors July 11, 1983. This work was presented at the SIAM Second Conference on the Applications of Discrete Mathematics held at Massachusetts Institute of Technology, Cambridge, Massachusetts, June 27–29, 1983.

† Department of Mathematics, Mount Holyoke College, South Hadley, Massachusetts 01075.

‡ Department of Mathematics and Statistics, University of South Carolina, Columbia, South Carolina 29208.

At the end of this process we get, for some m, the equation $R_m = 0$, i.e.

$$g^{\deg f} - f^{\deg g} - c_1 f^{p_1} g^{q_1} - \cdots - c_m f^{p_m} g^{q_m} = 0. \tag{2}$$

Let r be a root of $f(X)$; setting $X = r$ in the above equation gives

$$g(r)^{\deg f} - \sum_{\substack{i=1 \\ p_i=0}}^{m} c_i g(r)^{q_i} = 0. \tag{3}$$

Observe that, for each i, $0 \leqq q_i \leqq \deg R_i / \deg g$ and that $\deg R_i \leqq \deg R_1 < (\deg f)(\deg g)$. Therefore

$$0 \leqq q_i < \deg f \quad \text{for each } i.$$

Consequently, there is a polynomial $P(X)$ of degree $\deg f$ corresponding to (3) for which $P(g(r)) = 0$. It will be shown in the next section that when f is irreducible, $P(X)$ is necessarily a power of the minimal polynomial of $g(r)$. The following example shows that $P(X)$ is not always equal to the minimal polynomial of $g(r)$. When the derivative $P'(X)$ of $P(X)$ is not the zero polynomial, the minimal polynomial of $g(r)$ is given by $P(X)/gcd(P(X), P'(X))$.

Example. Let $f(X) = X^4 - 10X^2 + 1$, $g(X) = X^3 - 11X$, and assume that the scalar field is the field of rational numbers. The algorithm produces the polynomials $R_1, R_2, \cdots$ as follows.

$$R_1 = g^4 - f^3 = -14X^{10} + 423X^8 - 4264X^6 + 14338X^4 + 30X^2 - 1,$$

$$R_2 = R_1 + 14g^2 f = -25X^8 + 524X^6 - 2910X^4 + 1724X^2 - 1,$$

$$R_3 = R_2 + 25f^2 = 24X^6 - 360X^4 + 1224X^2 + 24,$$

$$R_4 = R_3 - 24g^2 = 168X^4 - 1680X^2 + 24,$$

$$R_5 = R_4 - 168f = -144,$$

$$R_6 = R_5 + 144 = 0.$$

Observe that

$$R_6 = 144 - 168f - 24g^2 + 25f^2 + 14g^2 f - f^3 + g^4 = 0.$$

If r is a root of f, then evaluating the above expression at r gives

$$144 - 24g(r)^2 + g(r)^4 = 0.$$

Thus $g(r)$ is a root of the polynomial $X^4 - 24X^2 + 144 = (X^2 - 12)^2$. The minimal polynomial of $g(r)$ is $X^2 - 12$.

3. Justification of the algorithm when the degrees are relatively prime. Most of the propositions in this section are familiar results in commutative algebra. We include them here for the sake of completeness.

PROPOSITION 1. *Let f and g be nonconstant elements of $K[X]$ and let $N = [K(f, g): K(g)]$; then $K[f, g] = K(g) + K[g]f + \cdots + K[g]f^{N-1}$.*

Proof. The monic minimal polynomial of f over $K(g)$ has coefficients in $K[g]$ because $K[X]$ is an integral extension of $K[g]$ and because $K[g]$ is integrally closed; for the definition and properties of integral extensions, see [11, pp. 9–11 and pp. 32–33].

Since furthermore the degree of the minimal polynomial of f over $K(g)$ is N, f^N lies in $K[g] + \cdots + K[g]f^{N-1}$. By induction any power of f lies in $K[g] + \cdots + K[g]f^{N-1}$. The proposition is an immediate consequence of this.

PROPOSITION 2. *Let f and g be elements of $K[X]$ whose degrees are relatively prime. For any nonzero h in $K[f, g]$, there exist nonnegative integers p, q such that* $\deg h = \deg f^p g^q$. *Furthermore* $[K(f, g): K(g)] = \deg g$.

Proof. Let $D = \deg g$. Observe that

$$[K(f, g): K(g)] \leqq [K(X): K(g)] \leqq D$$

(in fact $[K(X); K(g)] = D$, but we do not need this). By Proposition 1, there exist elements $h_0(X), \cdots, h_{D-1}(X)$ in $K[X]$ such that

$$h = h_0(g) + \cdots + h_{D-1}(g) f^{D-1}.$$

Since *gcd* $(\deg f, D) = 1$, the nonzero terms in the sequence $h_0(g), \cdots, h_{D-1}(g) f^{D-1}$ have distinct degrees. Hence, for some p, $\deg h = \deg h_p(g) f^p$. If $q = \deg h_p(X)$, then $\deg h = \deg f^p g^q$.

The fact that the degrees of $1, f, \cdots, f^{D-1}$ are pairwise incongruent mod D implies that the sequence $1, f, \cdots, f^{D-1}$ is linearly independent over $K(g)$. Therefore $[K(g, f): K(g)] \geqq D$; hence equality holds. This completes the proof.

Proposition 2 implies that, in the algorithm presented in the previous section, it is possible to find elements c_i, p_i, q_i satisfying relation (1) whenever $R_i \neq 0$.

PROPOSITION 3. *Let* $N = [K(f, g): K(g)]$.

(i) *Let $q(X, Y)$ be an element of $K[X, Y]$ of the form $q(X, Y) = X^N +$ an element of lower X degree. Assume that $q(f, g) = 0$; then $q(X, Y)$ is irreducible in $K[X, Y]$.*

(ii) *If $p(X, Y)$ and $\bar{p}(X, Y)$ are irreducible elements of $K[X, Y]$ such that $p(f, g) = \bar{p}(f, g) = 0$, then $\bar{p}(X, Y)$ is a scalar multiple of $p(X, Y)$.*

Proof. (i) Observe that $q(T, g)$ is the minimal polynomial of f over $K(g)$. Therefore it is irreducible in $K(g)[T]$, so $q(X, Y)$ is irreducible in $K(Y)[X]$. Hence the X degree of any factor of q is either 0 or N. On the other hand, because of the form of q, any factor of X degree 0 must be a scalar. Therefore q is irreducible in $K[X, Y]$.

(ii) By Gauss' lemma [10, p. 125, Lemma 3.27], p and $\bar{p}$ are both irreducible in $K(Y)[X]$. Therefore both $p(T, g)$ and $\bar{p}(T, g)$ must be equal, modulo factors in $K(g)$, to the minimal polynomial of f over $K(g)$. Hence $\bar{p}(X, Y)/p(X, Y)$ lies in $K(Y)$. Similarly, by comparing $\bar{p}$ and p to the minimal polynomial of g over $K(f)$, it follows that $\bar{p}/p$ lies in $K(X)$. Hence $\bar{p}/p$ is a scalar.

Note that if $p(X, Y)$ is irreducible and if $p(f, g) = 0$, then it is the polynomial of minimal total degree relating f and g.

PROPOSITION 4. *Assume that f is irreducible. Let $p(X, Y) \in K[X, Y]$ be the polynomial of minimal total degree such that $p(f, g) = 0$. Let r be a root of $f(X)$; then $p(0, Y)$ is a power of the minimal polynomial of $g(r)$ over K.*

Proof. Let $m(X)$ denote the minimal polynomial of $g(r)$ over K. By the natural isomorphism between $K[X]/f$ and $K[r]$, $m(g)$ must lie in $fK[X]$. Define w by the equation $m(g) = fw$. Since $K[X]$ is integral over $K[g]$, there exists an integer $n > 0$ and elements $c_1, \cdots, c_n$ in $K[g]$ such that

$$w^n = c_1 w^{n-1} + \cdots + c_n.$$

Multiplying this equation by f^n shows that $m(g)^n$ is in $fK[f, g]$. Therefore there exists an element $H(X, Y)$ in $K[X, Y]$ such that

$$m(g)^n - fH(f, g) = 0.$$

Therefore $p(X, Y)$ divides $m(Y)^n - XH(X, Y)$, so $p(0, Y)$ divides $m(Y)^n$. Since $m(Y)$ is irreducible, $p(0, Y)$ must be a power of $m(Y)$. This finishes the proof.

Propositions 2 and 3 imply that (2) is the polynomial of minimal total degree relating f and g. Therefore the polynomial produced by the algorithm of § 2 coincides with the polynomial $p(0, Y)$ mentioned in Proposition 4. Therefore it is a power of the minimal polynomial of $g(r)$ over K.

4. The general algorithm. Let f and g be nonconstant polynomials with coefficients in the field K. The goal of this section is to describe an algorithm to compute the polynomial $p(X, Y)$ of minimal total degree such that $p(f, g) = 0$. This algorithm works whether or not f is irreducible; when f is irreducible and when r is a root of f, the minimal polynomial of $g(r)$ over K can be obtained from $p(X, Y)$ by Proposition 4.

The outline of the general algorithm is as follows. For $n = 1, 2, \cdots$ it determines if f^n lies in $K[g]+\cdots+K[g]f^{n-1}$, stopping at the first n for which it does. By Proposition 1 the number n at which the algorithm stops is precisely the degree N of the field extension $[K(f, g): K(g)]$. The algorithm also expresses f^N as an element of $K[g]+\cdots+K[g]f^{N-1}$. By Proposition 3 this expression is equivalent to the minimal polynomial $p(X, Y)$.

To carry out the steps outlined above it is convenient to have a "nice" K basis for $K[g]+\cdots+K[g]f^{n-1}$, for each n satisfying $0 \leqq n < N$. The next proposition defines the type of basis that the algorithm will use.

PROPOSITION 5. *Let* $0 \leqq n < N$ (*where* $N = [K(f, g): K(g)]$). *Let* $w_0, \cdots, w_m$ *be nonzero elements of* $K[g]+\cdots+K[g]f^n$ *which satisfy the following conditions.*

(1) *The degrees of the* w_i *are pairwise incongruent* mod deg g.

(2) *For any nonzero element* y *in* $K[g]+\cdots+K[g]f^n$, *there is a subscript* $i = i(y)$ *such that* deg $w_i \equiv$ deg y mod deg g *and such that* deg $w_i \leqq$ deg y.

Define $S = \{w_i g^j: 0 \leqq i \leqq m, j \geqq 0\}$, *then* S *is a* K *basis for* $K[g]+\cdots+K[g]f^n$. *Furthermore* $n = m$.

Proof. By condition (1), the elements of S have distinct degrees, so the elements of S are linearly independent. Let y be a nonzero element of $K[g]+\cdots+K[g]f^n$; we will show by induction on deg y that y lies in span S. By condition (2) there exists an element s in S such that deg $s =$ deg y. Therefore, for some scalar c, deg$(y - cs) <$ deg y. By the induction hypothesis $y - cs$ is in span S; hence y is in span S. This proves that S is a basis.

The fact that S is a K basis for $K[g]+\cdots+K[g]f^n$ implies that $w_0, \cdots, w_m$ is a $K(g)$ basis for $K(g)+\cdots+K(g)f^n$. Since $n < N$, the elements $1, f, \cdots, f^n$ are linearly independent over $K(g)$. Thus $w_0, \cdots, w_m$ and $1, f, \cdots, f^n$ are bases for the same space, so $m = n$.

Remarks. 1. Let B be a K basis for $K[g]+\cdots+K[g]f^n$ such that the elements of B have distinct degrees. Define

$$\bar{B} = \{\bar{b} \in B: \text{there is some } b \in B \text{ such that deg } b \equiv \text{deg } \bar{b} \bmod \text{deg } g \text{ and such that deg } b < \text{deg } \bar{b}\}.$$

Observe that the elements of $B - \bar{B}$ satisfy the conditions of Proposition 5.

2. When deg f and deg g are relatively prime, the proof of Proposition 2 implies that the elements $1, f, \cdots, f^n$ satisfy the conditions of Proposition 5.

Let S be a subset of $K[X]-\{0\}$ whose elements have distinct degrees. For any y in $K[X]$, define $R(y, S)$ as follows. If deg y is different from the degree of any element of S, set $R(y, S) = y$. In general define

$$R(y, S) = y - c_1 s_1 - \cdots - c_t s_t,$$

where $s_1, \cdots, s_t$ are elements of S and $c_1, \cdots, c_t$ are scalars such that deg $y >$

$\deg(y - c_1 s_1) > \cdots > \deg(y - c_1 s_1 - \cdots - c_t s_t)$ and such that $\deg(y - c_1 s_1 - \cdots - c_t s_t)$ is different from the degree of any element of S.

Observe that $R(y, S) = 0$ if and only if y lies in span S.

In the special case that S is of the form $S = \{w_i g^j : 0 \leqq i \leqq n, 0 \leqq j\}$, where the degrees of the w_i are pairwise incongruent mod deg g, one can compute $R(y, S)$ as follows. Let i be the index such that $\deg w_i \equiv \deg y \bmod \deg g$; then choose j and c so that $\deg(y - c w_i g^j) < \deg y$. Repeating this process produces $R(y, S)$.

The goal of the following algorithm is to construct, for each integer n satisfying $0 \leqq n < N$, elements $w_{0,n}, \cdots, w_{n,n}$ in $K[g] + \cdots + K[g]f^n$ which satisfy the conditions of Proposition 5. The algorithm also produces the minimal polynomial $p(X, Y)$.

THE ALGORITHM. For $n = 0, 1, \cdots$ we recursively construct a polynomial M_n and a sequence of polynomials $W_n = (w_{0,n}, \cdots, w_{n,n})$ as follows.

$n = 0$

Set $M_0 = w_{0,0} = 1$. Proceed to the stage $n = 1$.

$n > 0$

Define $S = S_{n-1} = \{w_{i,n-1} g^j : 0 \leqq i \leqq n-1, j \geqq 0\}$. Define $M_n = R(M_{n-1} f, S)$. If $M_n = 0$, stop. Otherwise define a sequence of polynomials $v_0 = v_{0,n}, v_1 = v_{1,n}, \cdots$ as follows:

$$v_0 = M_n,$$

$$v_1 = R(g v_0, S \cup \{v_0\}),$$

$$\vdots$$

$$v_{j+1} = R(g v_j, S \cup \{v_0, \cdots, v_j\}).$$

$$\vdots$$

As will be shown below, there is a subscript j for which v_j is nonzero and for which $\deg v_j$ is not congruent mod deg g to the degree of any element of W_{n-1}. Let J be the smallest such subscript and set

$$T = \{w_{0,n-1}, \cdots, w_{n-1,n-1}, v_0, \cdots, v_{J-1}\}.$$

For $0 \leqq i \leqq n-1$ define $w_{i,n}$ to be the element t of T of smallest degree such that $\deg t \equiv \deg w_{i,n-1} \bmod \deg g$. Define $w_{n,n} = v_J$. Proceed to stage $n+1$.

To justify the algorithm we need some propositions.

PROPOSITION 6. *Fix $n > 0$. Assume that the set S_{n-1}, defined in the algorithm above, is a K basis for $K[g] + \cdots + K[g]f^{n-1}$. Either $M_n = 0$ or every term in the sequence $v_0 = v_{0,n}, v_1 = v_{1,n}, \cdots$ is nonzero. Furthermore, if $M_n \neq 0$, there is a subscript J for which v_J is not congruent* mod deg g *to the degree of any element of W_{n-1}.*

Proof. An easy induction argument shows, for every j, that S_j is contained in $K[g] + \cdots + K[g]f^j$ and that

$$M_j \in f^j + K[g]f^{j-1} + \cdots + K[g]. \tag{4}$$

Assume that $M_n \neq 0$; then fM_{n-1} is not in span S_{n-1}. Therefore, by (4) and by the hypothesis on S_{n-1}, f^n is not in $K[g] + \cdots + K[g]f^{n-1}$. Hence, by Proposition 1, $n < N$.

Since $n < N$, the elements $1, f, \cdots, f^n$ are linearly independent over $K(g)$. Therefore the set $S_{n-1} \cup \{v_0, v_1, \cdots\}$ is linearly independent over K. In particular each v_j is nonzero.

By the definition of the R function, $\deg v_j$ can never be both greater than, and congruent mod deg g to, some element of W_{n-1}. Furthermore the numbers of $\deg v_0, \deg v_1, \cdots$ are distinct. For each $i = 0, \cdots, n-1$ there are only finitely many positive integers congruent mod deg g to, but strictly less than, $\deg w_{i,n-1}$. Therefore there is a subscript J for which $\deg v_J$ is not congruent mod deg g to the degree of any element of W_{n-1}.

PROPOSITION 7. *Assume that* $0 \leqq n < N$; *then* S_n *is a* K *basis for* $K[g]+\cdots+K[g]f^n$.

Proof. Proceed by induction on n. Obviously S_0 is a basis for $K[g]$. Suppose now that $n>0$. Since $n<N$, fM_{n-1} is not in span S_{n-1}, so $M_n \neq 0$. Observe that the elements $v_0 = v_{0,n}, \cdots, v_J = v_{J,n}$ have the form

$$\begin{aligned} v_0 &= f^n + \text{element of span } S, \text{ (where } S = S_{n-1}), \\ v_1 &= gf^n + \text{an element of span } (S \cup \{f^n\}), \\ &\vdots \\ v_J &= g^J f^n + \text{an element of span } (S \cup \{f^n, \cdots, g^{J-1}f^n\}). \end{aligned}$$

Let $y \in K[g]+\cdots+K[g]f^n$. There is a polynomial $q(X)$ such that

$$y - q(g)v_J \in K[g]+\cdots+K[g]f^{n-1}+Kf^n+\cdots+Kg^{J-1}f^n.$$

If $J>1$ there is a scalar c_1 such that

$$y - q(g)v_J - c_1 v_{J-1} \in K[g]+\cdots+K[g]f^{n-1}+Kf^n+\cdots+Kg^{J-2}f^n.$$

Continuing in this way we can find scalars $c_2, \cdots, c_J$ such that

$$y - q(g)v_J - c_1 v_{J-1} - \cdots - c_J v_0 \in K[g]+\cdots+K[g]f^{n-1}.$$

By the induction hypothesis, this element lies in span S. Hence, setting

$$B = \{g^m w_{i,n-1}, v_0, \cdots, v_{J-1}, g^m v_J : 0 \leqq i \leqq n-1, m \geqq 0\},$$

the set B spans $K[g]+\cdots+K[g]f^n$. By the definition of J, $\deg v_J$ is not congruent mod $\deg g$ to the degree of any element of W_{n-1}; therefore the elements of B have distinct degrees. Thus B is a basis for $K[g]+\cdots+K[g]f^n$ with distinct degrees. By the first remark following Proposition 5, the elements of W_n satisfy the conditions of Proposition 5. Therefore S_n is a basis for $K[g]+\cdots+K[g]f^n$.

Propositions 1 and 7 imply that the algorithm stops when $n=N$. The equation $M_N = R(M_{N-1}f, S) = 0$ is equivalent to an expression for f^N as an element of $K[g]+\cdots+K[g]f^{N-1}$. By Proposition 3, this is the same as the minimal polynomial $p(X, Y)$.

Several observations concerning the general algorithm may be helpful.

If $\deg f$ and $\deg g$ are relatively prime, then $M_n = f^n$ and $W_n = (1, f, \cdots, f^n)$ whenever $0 \leqq n < N$. The computation of $R(M_{N-1}f, S)$ which occurs in the last stage of the general algorithm coincides in this case with the algorithm of § 2.

Results of Abhyankar [1, pp. 366–374], [15] imply that when the characteristic of K does not divide $gcd(\deg f, \deg g)$ and when $0 < n < N$, $W_n = (M_0, \cdots, M_n)$. Thus the algorithm simplifies considerably, for there is no need to compute more than the first term of the v_i sequence. In general this simplification need not hold; for example if $f = X^6 + X$, $g = X^4$ and $K = GF(2)$, then $W_2 = (1, X^2, X)$ and $(M_0, M_1, M_2) = (1, X^6+X, X^2)$.

The efficiency of the algorithm can be improved in a number of ways. For example, at the nth stage, rather than setting $M_n = R(M_{n-1}f, S)$, we can search all the previous M_i, find the index i for which $\deg M_i M_{n-i}$ is minimal, and set $M_n = R(M_i M_{n-i}, S)$. Since $\deg M_i M_{n-i}$ may be smaller than $\deg M_{n-1}f$, it probably takes fewer arithmetic operations to compute $R(M_i M_{n-i}, S)$ from $M_i M_{n-i}$ than it does to compute $R(M_{n-1}f, S)$ from M_{n-1}. Another improvement can be made by changing the definition of the sets S_n. The definition given above involves computing $v_0, \cdots, v_J$ and then throwing away some subset of $W_{n-1} \cup \{v_0, \cdots, v_J\}$. It is wasteful to throw away any elements as they might help in the computation of M_i's and v_j's at later stages. Instead, one can define

S_n recursively by

$$S_n = S_{n-1} \cup \{v_0, \cdots, v_{J-1}, v_J g^j : j \geqq 0\};$$

there is no need to bother with the W_n sequences. Of course, by the preceding remark, this definition of S_n differs from the original one only when the characteristic of K is nonzero.

Finally, we consider the following question: given a polynomial h, does h lie in $K[f, g]$? To answer this question, compute $R(h, S_{N-1})$; h lies in $K[f, g]$ precisely when $R(h, S_{N-1}) = 0$. This question, especially in the case $h = X$, is of interest in algebraic geometry; see [1], [2].

5. Some remarks on the speed of the algorithm. We estimate the number of multiplications performed by the algorithm when $\deg f = n$, $\deg g = m$, $m < n$, and $gcd(m, n) = 1$. This estimate will reflect the amount of time needed to run the algorithm.

In general the algorithm computes the polynomials $f^i g^j$ for all i, j such that $in + jm \leqq nm$. The number of scalar multiplications needed to do this is of the same order of magnitude as the number of multiplications needed to compute $f^i g^j$ for all i, j such that $0 \leqq i \leqq m$ and $0 \leqq j \leqq n$.

We assume that polynomials are multiplied in the naive manner, so to multiply polynomials of degree p and q, $(p+1)(q+1)$ scalar multiplications are required. Assume that $i \geqq 1$ and that f^j has been computed for $j \leqq i$. There are several pairs of positive integers A, B such that $f^{i+1} = (f^A)(f^B)$; the number of scalar multiplications is minimized when $A = i$ and $B = 1$, and in this case $(n+1)(in+1)$ scalar multiplications are required. Therefore to compute the sequence $f^2, f^2, \cdots, f^m$, the number of scalar multiplications needed is $\sum_{i=1}^{m-1} (n+1)(in+1) = 0(n^2 m^2)$.

Now assume that i and j are not both zero. There are several ways to express $f^i g^{j+1}$ as the product $f^i g^{j+1} = (f^A g^B)(f^C g^D)$ so that $0 < A + B, C + D < i + j + 1$. Assuming that $f^A g^B$ and $f^C g^D$ are already known, the number of scalar multiplications in the product $(f^A g^B)(f^C g^D)$ is minimized when $A = i$, $B = j$, $C = 0$ and $D = 1$, since $m < n$. In this case $(in + jm + 1)(m+1)$ scalar multiplications are required. Therefore to compute $f^i g^j$ for $0 \leqq i \leqq m$, $1 \leqq j \leqq n$, the number of scalar multiplications needed is

$$\sum_{\substack{i=0 \\ (i,j) \neq (0,0)}}^{m} \sum_{j=0}^{n-1} (in + jm + 1)(m+1) = O(m^3 n^2).$$

Thus to compute the polynomials $f^i g^j$, $O(m^3 n^2)$ scalar multiplications are required.

Assume that $f^p g^q$ has been computed for each pair p, q satisfying $np + mq \leqq nm$. If c is a scalar, at most $nm + 1$ scalar multiplications are needed to compute $cf^p g^q$. Therefore, for each i, at most $nm + 1$ scalar multiplications are needed to compute the polynomials R_i that is defined in the beginning of § 2. Since the degree of R_1 is less than nm and since the sequence $\deg R_1, \deg R_2, \cdots$ is strictly decreasing, there are at most nm subscripts i for which $R_i \neq 0$. Hence the number of scalar multiplications needed to compute the sequence $R_1, R_2, \cdots$ is $O(n^2 m^2)$. Since it took $O(m^3 n^2)$ multiplications to compute the polynomials $f^p g^q$, the algorithm uses altogether $O(m^3 n^2)$ scalar multiplications.

The greatest common divisor of a polynomial and its derivative can be computed quickly: ($O(n \log^2 n)$ arithmetic operations for a polynomial of degree n, using the finite Fourier transform, and $O(n^2 \log n)$ operations using naive polynomial multiplication; see [3, 308]). Therefore, when K is a field of characteristic 0 or a finite field whose size is prime, the problem of extracting the actual minimal polynomial from a

power of the minimal polynomial does not significantly affect the estimate $O(m^3n^2)$. This problem may become significant when K is an extension of $GF(p)$ of large degree.

Let g_j denote the polynomial of degree less than n which is congruent to $g^j \bmod f$. One approach to finding the minimal polynomial of $g(r)$, where r is a root of f, is to compute a linear dependence among the elements $1, g_1, \cdots, g_n$; this approach is described in [5, p. 112]. Using Gaussian elimination, the linear dependence can be computed using $O(n^3)$ multiplications. Using accelerated methods of matrix multiplication [3, pp. 240–242] and [6], the number of multiplications can be improved to $O(n^{2.49\cdots})$; there are however problems with these accelerated methods [3, p. 226]. Recall that the algorithm of § 2 uses $O(m^3n^2)$ multiplications. This suggests that when m is sufficiently small compared to n and when $gcd(n, m) = 1$, the algorithm of this paper is faster than the "linear dependence" algorithm.

We conjecture that there is a minimal polynomial algorithm, for arbitrary f and g, such that the number of multiplications used is bounded by a function of the form $c(m)n^2$.

When K is the set of rational numbers, the numerators and denominators occurring in the coefficients of the f^ig^j's are usually much larger than those of the minimal polynomial of $g(r)$. To avoid overflow errors and to increase efficiency, we recommend first computing the coefficients modulo some large prime power; this approach is described in [12].

Acknowledgment. We would like to thank Professor R. J. McEliece for helping us with the last section of this paper.

REFERENCES

[1] S. S. ABHYANKAR, *On the semigroup of a meromorphic curve*, I, Proc. International Symposium on Algebraic Geometry, Kyoto, Nagata ed., Kinokuniya Book-Store Co., Ltd., Tokyo, 1978, pp. 249–414.

[2] S. S. ABHYANKAR AND T. T. MOH, *Embeddings of the line in the plane*, J. Reine Angew. Math., 276 (1975), pp. 149–166.

[3] A. V. AHO, J. E. HOPCROFT AND J. D. ULLMAN, *The Design and Analysis of Computer Algorithms*, Addison-Wesley, Reading, MA, 1974.

[4] A. A. ALBERT, *Fundamental Concepts of Higher Algebra*, Univ. Chicago Press, Chicago, 1956.

[5] E. R. BERLEKAMP, *Algebraic Coding Theory*, McGraw-Hill, New York, 1968.

[6] D. COPPERSMITH AND S. WINOGRAD, *On the asymptotic complexity of matrix multiplication*, SIAM J. Comput., 11 (1982), pp. 472–492.

[7] D. E. DAYKIN, *Generation of irreducible polynomials over a finite field*, Amer. Math. Monthly, 72 (1960), pp. 646–648.

[8] B. ELSPAS, *The theory of autonomous linear sequential networks*, IRE Trans. Circuit Theory, CT-6 (1959), pp. 45–60.

[9] S. W. GOLOMB, *Irreducible polynomials, synchronization codes, primitive necklaces and the cyclotomic algebra* in Combinatorial Mathematics and its Applications, R. C. Bose and T. Dowling, eds., Univ. North Carolina Press, Chapel Hill, 1969, pp. 358–370.

[10] I. N. HERSTEIN, *Topics in Algebra*, Blaisdell, Waltham, MA, 1964.

[11] I. KAPLANSKY, *Commutative Rings*, rev. edn., Univ. Chicago Press, Chicago, 1974.

[12] E. V. KRISHNAMURTY, T. M. RAO AND K. SUBRAMANIAN, *Finite segment p-adic number systems with applications to exact computations*, Proc. Indian Acad. Sci., 81A (1975), pp. 58–79.

[13] R. W. MARSH, *Table of irreducible polynomials over* GF(2) *through degree* 19, U.S. Dept. of Commerce, Office of Technical Services, Washington, DC, 1957.

[14] W. W. PETERSON AND E. J. WELDON, JR., *Error-Correcting Codes*, second edn., MIT Press, Cambridge, MA, 1972.

[15] D. RICHMAN, *On the computation of minimal polynomials*, preprint.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 2, April 1985

015

DISJOINT PATHS—A SURVEY*

NEIL ROBERTSON† AND P. D. SEYMOUR†‡

Abstract. We describe without proof polynomially bounded algorithms for the following problems:
(i) (*k is a fixed integer, and S is a fixed surface*). With input a graph G which may be drawn in S and k pairs of vertices of G, decide if there are k vertex-disjoint paths of G, each joining one of the pairs of vertices;
(ii) (*H is a fixed planar graph*). With input a graph G, decide if G can be reduced to a graph isomorphic to H by deletion and contraction of edges.

1. Introduction. Over the past two years we have been working on a conjecture of Wagner, that given any infinite set of graphs, one of its members is isomorphic to a "minor" of another. (H is a *minor* of G if H can be obtained from a subgraph of G by contraction.) The pursuit of this conjecture has led to two theorems which provide polynomially bounded algorithms for certain graph-theoretic problems, and we sketch these algorithms here. Full details will appear in [5], [8], [9].

We begin with the following problem.

DISJOINT CONNECTING PATHS.

Instance. A graph G, and pairs $(s_1, t_1), \cdots, (s_k, t_k)$ of vertices of G.

Question. Do there exist paths $P_1, \cdots, P_k$ of G, pairwise vertex-disjoint, such that P_i has ends s_i and t_i $(1 \leqq i \leqq k)$?

Karp [3] showed that this problem is NP-complete, and Lynch [4] proved the same even if G is restricted to be planar. But both these results require that k be part of the input; if k is fixed, the existence of a polynomial algorithm for the problem appears much more likely.

If we consider the same problem except that we input a directed graph, and ask that $P_1, \cdots, P_k$ be directed paths, the problem is known to be NP-complete for fixed k, and even for $k = 2$, as was shown by Fortune, Hopcroft and Wyllie [2]. One should not be discouraged by this, for the undirected case with which we are concerned appears to be significantly easier than the directed case. In the undirected case, for example, there *is* a polynomial algorithm when $k = 2$, as Seymour [9] and Shiloach [10] independently discovered. (For fixed $k \geqq 3$, the problem is still open.) The first of our two results is that if k is fixed and G is restricted to be planar (or more generally, to lie on any fixed surface) then there is a polynomially bounded algorithm for DISJOINT CONNECTING PATHS.

We should mention, however, that this result is of little practical significance. The degree of the polynomial is of the form

$$k^{k^{k^{\cdot^{\cdot^{\cdot^{k}}}}}}.$$

This is a constant, since k is fixed, but it is too large to be useful. It is mainly from the viewpoint of the theory of NP-completeness that the result is interesting.

* Received by the editors July 25, 1983, and in final revised form April 2, 1984. A version of this work was presented at the SIAM Second Conference on the Applications of Discrete Mathematics held at Massachusetts Institute of Technology, Cambridge, Massachusetts, June 27-29, 1983.

† Department of Mathematics, Ohio State University, Columbus, Ohio 43210.

‡ Current address, Bell Communications Research, Murray Hill, New Jersey 07974. The research of this author was partially supported by the National Science Foundation under grant MCS 8103440.

Our second algorithm concerns the following problem.

GRAPH MINOR.

Instance. Graphs G and H.

Question. Does G have a minor isomorphic to H?

This is easily seen to be NP-complete if H is part of the input; for example, if H is chosen to be a circuit graph with $|V(G)|$ vertices, the question becomes "does G have a Hamiltonian circuit?", which is NP-complete. However, for any small fixed graph H (with up to, say, 10 edges) GRAPH MINOR is polynomially solvable. For example, if H consists of a single loop, we must test "is G a forest?". If H is the complete graph with four vertices, we must test "is G a series-parallel graph?" (Dirac [1]). We conjecture that for all fixed H GRAPH MINOR is polynomially solvable; and our second main result is that this is so if H is planar. Unfortunately, the degree of the polynomial is an iterated exponential in $|V(H)|$, and so once again the algorithm is not a practical one.

Both algorithms use the same idea; that if G is sufficiently "uniform" the answer to the question is "yes", and if it is not we can reduce the problem to a few significantly simpler ones by "splitting" a small number of vertices of G.

Section 2 is devoted to splitting vertices, and the two algorithms are described in §§ 3 and 4. Section 5 contains some concluding remarks.

2. Splitting vertices. Suppose that we are given a graph G and pairs of vertices $(s_1, t_1), \cdots, (s_k, t_k)$, and we wish to solve DISJOINT CONNECTING PATHS. Let us say that the list of pairs $(s_1, t_1), \cdots, (s_k, t_k)$ is *feasible* if the paths exist. We may assume that G is simple, since loops and parallel edges have no effect on our problem. Let v be a vertex of G, with neighbors $a_1, \cdots, a_m, b_1, \cdots, b_n$. Let G' be the graph obtained from G by deleting v and adding two new vertices a, b where a is adjacent to $a_1, \cdots, a_m$ and b to $b_1, \cdots, b_n$. This construction of G' from G is called *splitting* v. Now G' has more vertices than G, but we shall use the construction in situations where DISJOINT CONNECTING PATHS is easier to solve in G' than in G. Thus it is useful that the problem for G can be transformed into problems for G'.

We assume that $v \neq s_1, t_1, \cdots, s_k, t_k$. (If not, the transformation is even easier.) Let G_a be the graph obtained from G' by deleting b, and define G_b similarly.

THEOREM 2.1. *The list* $(s_1, t_1), \cdots, (s_k, t_k)$ *is feasible in* G *if and only if either*

(i) $(s_1, t_1), \cdots, (s_k, t_k)$ *is feasible in* G_a *or* G_b;

(ii) *for some* $(1 \leqq i \leqq k)$,

$$(s_1, t_1), \cdots, (s_{i-1}, t_{i-1}), (s_i, a), (b, t_i), (s_{i+1}, t_{i+1}), \cdots, (s_k, t_k)$$

is feasible in G'; *or*

(iii) *for some* i $(1 \leqq i \leqq k)$,

$$(s_1, t_1), \cdots, (s_{i-1}, t_{i-1}), (s_i, b), (a, t_i), (s_{i+1}, t_{i+1}), \cdots, (s_k, t_k)$$

is feasible in G'.

The proof is easy—we simply check through the ways in which v could be used in a set of disjoint paths of G—and is left to the reader.

We observe that this reduces our original question about G to a set of $2k+2$ questions about G_a, G_b and G'. That this reduction can be useful will be shown in the next two sections.

3. Disjoint paths in planar graphs. For simplicity, we first discuss the algorithm for DISJOINT CONNECTING PATHS for graphs restricted to be planar; the case of a general fixed surface is similar but more complicated.

Let $S(h)$ be the surface formed by removing h open discs with disjoint closures from the 2-sphere. Then $S(h)$ has a boundary composed of h disjoint 1-spheres—we call these 1-spheres the *cuffs* of $S(h)$. If $s_1, t_1, \cdots, s_k, t_k$ are vertices of a planar graph G, there is an integer h such that G can be drawn in $S(h)$ with $s_1, t_1, \cdots, s_k, t_k$ in the boundary of $S(h)$ and with the drawing using no other points of the boundary of $S(h)$. We call this a *proper* drawing. We shall show by induction on h that there is a polynomial algorithm to test if $(s_1, t_1), \cdots, (s_k, t_k)$ is feasible in G.

There are two necessary conditions for feasibility which we now discuss.

Connectivity condition. For any subset $X \subseteq V(G)$, there are at most $|X|$ values of i $(1 \leqq i \leqq k)$ such that every path in G from s_i to t_i meets X.

Planarity condition. There are pairwise disjoint arcs (continuous images of [0, 1]) in $S(h)$, $Q_1, \cdots, Q_k$ say, such that Q_i joins s_i and t_i $(1 \leqq i \leqq k)$.

For example, if $k = 2$ and $h = 1$, and s_1, s_2, t_1, t_2 occur on the cuff in that order, the planarity condition fails. A second example is when $k = 3$, $h = 2$ and s_1, s_2, s_3 occur on the first cuff, and t_1, t_2, t_3 on the second cuff, both in the same clockwise order.

It is easy to see that the above two conditions are necessary for feasibility. They are not in general sufficient, however. For example, with G as in Fig. 1, both conditions are satisfied, but the pairing is not feasible.

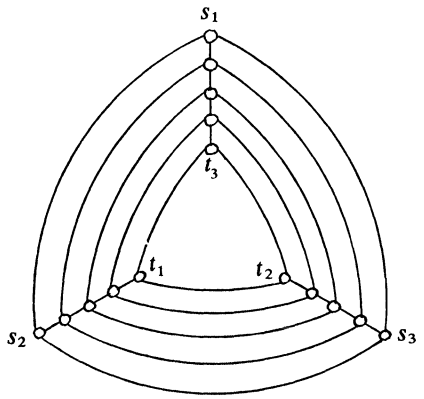


FIG. 1.

The following theorem appears, rather surprisingly, to be new. Its proof is not hard (see [8]).

THEOREM 3.1. *If $h = 1$ and the connectivity and planarity conditions are satisfied then the pairing is feasible.*

In a proper drawing of G on $S(h)$, the points of $S(h)$ not used by the drawing are partitioned into connected pieces called *regions.* (Recall that $S(h)$ is a sphere with discs removed so that portions of cuffs can be part of the boundary of a region.) A sequence $R_0, R_1, R_2, \cdots, R_n = R_0$ of regions is a *ring* if $R_1, \cdots, R_n$ are all distinct, and for $i = 1, \cdots, n$ R_{i-1} and R_i are incident with a common vertex v_i, and $v_1, \cdots, v_n$ are all distinct. Given any ring we may "cut" along it in the obvious way, separating the surface into two parts which we call *hemispheres.* There is a corresponding splitting of some of the vertices of G, as in Fig. 2.

A ring is *central* if both hemispheres contain at least two of the original cuffs, or equivalently if they both have fewer cuffs than the original surface (counting the new one). A ring is *r-peripheral* if one of the hemispheres contains exactly one of the original cuffs, and that cuff contains exactly r vertices of the graph. A *handcuff* is composed

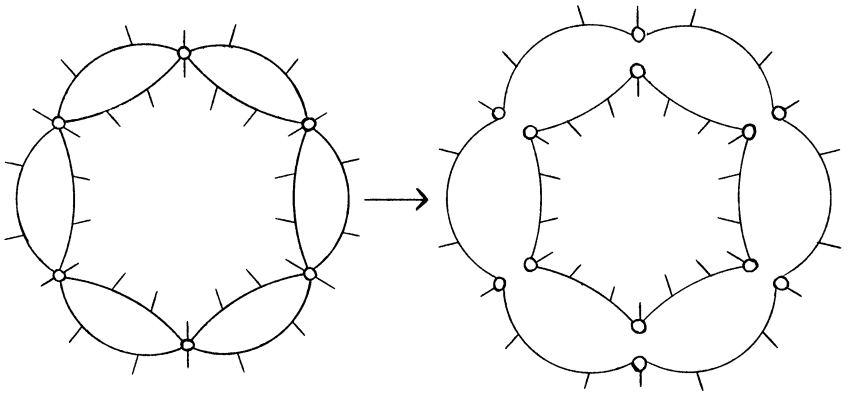


FIG. 2.

of two disjoint peripheral rings, surrounding different cuffs, and a chain of regions between them. The theorem on which our algorithm is based is the following.

THEOREM 3.2. *If $h \geqq 4$, there is a number $f(h, k)$ such that if*

(i) *the connectivity and planarity conditions are satisfied,*

(ii) *every central ring and every handcuff has at least $f(h, k)$ regions, and*

(iii) *for $r \geqq 0$, every r-peripheral ring has at least r regions, then the pairing is feasible.*

The proof of this is quite complicated—see [9]. Using it in an algorithm however is easy.

ALGORITHM

Step 1. Check the connectivity and planarity conditions. If these are not both satisfied, the pairing is not feasible. If they are both satisfied, perform

Step 2. Test if there is a central ring or a handcuff with fewer than $f(h, k)$ regions, or for some $r > 0$ an r-peripheral ring with fewer than r regions. If not and $h \geqq 4$ then the pairing is feasible. If not and $h < 4$ we use special methods not described here. If we find a ring or handcuff which is shorter than it should be, we perform

Step 3. Cut along the ring, (or, similarly, along the handcuff) splitting its vertices, using Theorem 2.1. We reduce the solution of the original problem to the solution of several (a great many, but bounded by a function of k and h) problems on graphs with proper drawings either in $S(h-1)$, or in $S(h)$ but with fewer than k pairs of vertices to be joined, or in $S(2)$. By induction on h, and for fixed h on k, there are polynomial time algorithms for these problems, and hence for the original problem.

For the case when $h = 3$, we use the same method except that we must augment Theorem 3.2(ii) with a θ-shaped configuration. If $h = 1$ or 2 we can solve the problem directly [8].

For the more general problem where G may be drawn in a fixed surface S but is not necessarily planar, we need an extension of (3.2). In this case we redefine a central ring to be a ring which either separates at least two cuffs from at least two other cuffs, or which is not "null-homotopic" in the surface obtained from S by "capping" the cuffs with discs. With this definition (3.2) remains true, except that the number $f(h, k)$ becomes $f(h, k, S)$: and we proceed by induction of the genus of S, and for fixed genus on h, and for fixed genus and h on k.

We have seen then that for planar graphs there is a polynomial algorithm for our problem if k is fixed, but not if k is variable (assuming $P \neq NP$). It is not known whether it is enough to fix h instead of k. This is true if $h = 1$ or 2.

4. Planar minors. A *subdivision* of a graph H is a graph obtained from H by repeatedly replacing edges by pairs of edges in series. We say that G *topologically contains* H if G has a subgraph which is isomorphic to a subdivision of H. If G topologically contains H then H is isomorphic to a minor of G, but the converse does not hold, as is easily seen.

On the other hand, we have the following.

THEOREM 4.1. *Let H be a graph. There is a finite list of graphs $H_1, \cdots, H_n$ such that for any graph G, the following are equivalent*:

(i) *G has a minor isomorphic to H.*

(ii) *G topologically contains one of $H_1, \cdots, H_n$.*

This is a standard result, and is easily proved. $H_1, \cdots, H_n$ may be taken to be those graphs H' (up to isomorphism) such that

(a) starting from H' and contracting edges which are not loops we may obtain a graph isomorphic to H;

(b) H' is not a subdivision of any other graph with property (a).

It follows from Theorem 4.1 that if there is in general for all fixed H a polynomial algorithm to test "does G topologically contain H?" then for fixed H there is a polynomial algorithm for GRAPH MINOR. Furthermore, there is a polynomial algorithm for the first problem if for fixed k there is a polynomial algorithm for DISJOINT CONNECTING PATHS, as is easily seen. (We try all ways of choosing vertices of G to represent vertices of H.) However DISJOINT CONNECTING PATHS is not yet polynomially solved. What we have instead is (for a fixed planar graph H and integer k) a polynomial algorithm which, with input a graph G and k pairs of vertices of G, either tests if the pairing is feasible, or discovers that G has a minor isomorphic to H. Evidently such an algorithm can be used to polynomially solve GRAPH MINOR when H is fixed and planar. Our algorithm is based on the following theorem.

THEOREM 4.2. *For any planar graph H there is a number N with the following property. For every graph G with no minor isomorphic to H, and every subset $X \subseteq V(G)$, there is a separation (V_1, V_2) of G such that*

$$|(V_1 - V_2) \cap X|, |(V_2 - V_1) \cap X| \leqq \tfrac{2}{3}|X|$$

and $|V_1 \cap V_2| \leqq N$.

(A *separation* (V_1, V_2) of G is a pair of subsets of $V(G)$ such that $V_1 \cup V_2 = V(G)$ and such that no edge of G joins a vertex of $V_1 - V_2$ with a vertex of $V_2 - V_1$.)

We remark that if H is nonplanar, there is no such number N; for example, if G is a large square "grid" of side $\gg N$ (for any given N), then it has no separation as in Theorem 4.2 and yet has no minor isomorphic to H, since H is nonplanar.

Theorem 4.2 with $X = V(G)$ suggests a "divide and conquer" method for GRAPH MINOR, and that is our method. Thus, let H be a fixed planar graph. Choose N as in Theorem 4.2; and let k be a fixed integer. We may assume that $k \gg N$.

ALGORITHM. We have as input then a graph G, and k pairs $(s_1, t_1), \cdots, (s_k, t_k)$ of vertices of G.

Step 1. We test if there is a separation (V_1, V_2) of G with $|V_1 \cap V_2| \leqq N$ and $|V_1 - V_2|, |V_2 - V_1| \leqq \frac{2}{3}|V(G)|$. If not then G has a minor isomorphic to H by (4.2). If we find such a separation we perform

Step 2. Split the vertices of $V_1 \cap V_2$ using Theorem 2.1 in the obvious way. The solution to our original problem is thus reduced to solving several problems about graphs each with at most

$$\tfrac{2}{3}|V(G)| + N$$

vertices. However, the number "k" is increased in these smaller problems. To bring it back under control, we perform

Step 3. In each of these smaller problems G', $(s_1', t_1'), \cdots, (s_{k'}', t_{k'}')$ say, test if there is a separation (V_1', V_2') of G' with $|V_1' \cap V_2'| \leq N$ and

$$|(V_1' - V_2') \cap X|, \qquad |(V_2' - V_1') \cap X| \leq \tfrac{2}{3}|X|,$$

where $X = \{s_1', t_1', \cdots, s_{k'}', t_{k'}'\}$. If not then G' and hence G has a minor isomorphic to H. If there is such a separation, we split again, reducing to problems on graphs with at most $\frac{2}{3}|V(G)| + N$ vertices and with at most k pairs of vertices to be joined by disjoint paths. Then we return to step 1.

It is easy to check that this algorithm is indeed polynomially bounded.

5. Some remarks. Let H, H' be two fixed graphs, where H is planar. The argument of the previous section yields a polynomial algorithm, which given any graph G, outputs either

(i) G has a minor isomorphic to H',

(ii) G has no minor isomorphic to H', or

(iii) G has a minor isomorphic to H.

We deduce if $H_1, \cdots, H_n$ is any finite sequence of graphs, and one of them is planar, then there is a polynomially bounded algorithm to test "does G have a minor isomorphic to one of $H_1, \cdots, H_n$?"

It follows that

THEOREM 5.1. *Let $\mathbb{F}$ be a set of graphs, closed under isomorphism and under taking minors, and suppose some planar graph is not in $\mathbb{F}$. Then there is a polynomial algorithm to test membership of $\mathbb{F}$.*

For it is proved in [7] that the set of minor-minimal graphs not in $\mathbb{F}$ is finite, up to isomorphism. Let this set be $\{H_1, \cdots, H_n\}$ say. One of these is planar, since some planar graph is not in $\mathbb{F}$. But a graph is in $\mathbb{F}$ if and only if it has no minor isomorphic to one of $H_1, \cdots, H_n$, and this can be tested in polynomial time, as we have seen.

In Theorem 5.1 it is not at all obvious that there should be any algorithm at all to test membership of $\mathbb{F}$; for there are only countably many algorithms, and one might expect there to be uncountably many choices for $\mathbb{F}$. It is entertaining that not only is there an algorithm but there is a polynomially bounded one.

Finally, it is natural to ask whether the hypothesis that some planar graph not be in $\mathbb{F}$ is necessary. We think not, that Theorem 5.1 holds with this hypothesis omitted. This conjecture turns out to be equivalent to the conjunction of the conjecture of Wagner and our conjecture on GRAPH MINOR previously mentioned.

REFERENCES

[1] G. A. DIRAC, *A property of 4-chromatic graphs and remarks on critical graphs*, J. London Math. Soc., 27 (1952), pp. 85-92.

[2] S. FORTUNE, J. HOPCROFT AND J. WYLLIE, *The directed subgraph homeomorphism problem*, J. Theoret. Comput. Sci., 10 (1980), pp. 111-121.

[3] R. M. KARP, *On the complexity of combinatorial problems*, Networks, 5 (1975), pp. 45-68.

[4] J. F. LYNCH, *The equivalence of theorem proving and the interconnection problem*, ACM SIGDA Newsletter 5:3 (1976).

[5] N. ROBERTSON AND P. D. SEYMOUR, *Graph minors* II: *algorithmic aspects of tree-width*, submitted.

[6] ———, *Graph minors* IV: *tree-width and well-quasi-ordering*, submitted.

[7] ———, *Graph minors* V: *excluding a planar graph*, submitted.

[8] ———, *Graph minors* VI: *disjoint paths across a disc*, submitted.

[9] ———, *Graph minors* VII: *disjoint paths on a surface*, submitted.

[10] P. D. SEYMOUR, *Disjoint paths in graphs*, Discrete Math., 29 (1980), pp. 293-309.

[11] Y. SHILOACH, *A polynomial solution to the undirected two paths problem*, J. Assoc. Comput. Mach., 27 (1980), pp. 445-456.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 2, April 1985

016

AMORTIZED COMPUTATIONAL COMPLEXITY*

ROBERT ENDRE TARJAN†

Abstract. A powerful technique in the complexity analysis of data structures is *amortization*, or averaging over time. Amortized running time is a realistic but robust complexity measure for which we can obtain surprisingly tight upper and lower bounds on a variety of algorithms. By following the principle of designing algorithms whose amortized complexity is low, we obtain "self-adjusting" data structures that are simple, flexible and efficient. This paper surveys recent work by several researchers on amortized complexity.

ASM(MOS) subject classifications. 68C25, 68E05

1. Introduction. Webster's [34] defines "amortize" as "to put money aside at intervals, as in a sinking fund, for gradual payment of (a debt, etc.)." We shall adapt this term to computational complexity, meaning by it "to average over time" or, more precisely, "to average the running times of operations in a sequence over the sequence." The following observation motivates our study of amortization: In many uses of data structures, a sequence of operations, rather than just a single operation, is performed, and we are interested in the total time of the sequence, rather than in the times of the individual operations. A worst-case analysis, in which we sum the worst-case times of the individual operations, may be unduly pessimistic, because it ignores correlated effects of the operations on the data structure. On the other hand, an average-case analysis may be inaccurate, since the probabilistic assumptions needed to carry out the analysis may be false. In such a situation, an amortized analysis, in which we average the running time per operation over a (worst-case) sequence of operations, can yield an answer that is both realistic and robust.

To make the idea of amortization and the motivation behind it more concrete, let us consider a very simple example. Consider the manipulation of a stack by a sequence of operations composed of two kinds of unit-time primitives: *push*, which adds a new item to the top of the stack, and *pop*, which removes and returns the top item on the stack. We wish to analyze the running time of a sequence of operations, each composed of zero or more pops followed by a push. Assume we start with an empty stack and carry out m such operations. A single operation in the sequence can take up to m time units, as happens if each of the first $m-1$ operations performs no pops and the last operation performs $m-1$ pops. However, altogether the m operations can perform at most $2m$ pushes and pops, since there are only m pushes altogether and each pop must correspond to an earlier push.

This example may seem too simple to be useful, but such stack manipulation indeed occurs in applications as diverse as planarity-testing [14] and related problems [24] and linear-time string matching [18]. In this paper we shall survey a number of settings in which amortization is useful. Not only does amortized running time provide a more exact way to measure the running time of known algorithms, but it suggests that there may be new algorithms efficient in an amortized rather than a worst-case sense. As we shall see, such algorithms do exist, and they are simpler, more efficient, and more flexible than their worst-case cousins.

* Received by the editors December 29, 1983. This work was presented at the SIAM Second Conference on the Applications of Discrete Mathematics held at Massachusetts Institute of Technology, Cambridge, Massachusetts, June 27–29, 1983.

† Bell Laboratories, Murray Hill, New Jersey 07974.

The paper contains five sections. In § 2 we develop a theoretical framework for analyzing the amortized running time of operations on a data structure. In § 3 we study three uses of amortization in the analysis of known algorithms. In § 4 we discuss two new data structures specifically designed to have good amortized efficiency. Section 5 contains conclusions.

2. Two views of amortization. In order to analyze the amortized running time of operations on a data structure, we need a technique for averaging over time. In general, on data structures of low amortized complexity, the running times of successive operations can fluctuate considerably, but only in such a way that the average running time of an operation in a sequence is small. To analyze such a situation, we must be able to bound the fluctuations. We shall consider two ways of doing this.

The first is the *banker's view* of amortization. We assume that our computer is coin-operated; inserting a single coin, which we call a *credit*, causes the machine to run for a fixed constant amount of time. To each operation we allocate a certain number of credits, defined to be the *amortized time* of the operation. Our goal is to show that all the operations can be performed with the allocated credits, assuming that we begin with no credits and that unused credits can be carried over to later operations. If desired we can also allow borrowing of credits, as long as any debt incurred is eventually paid off out of credits allocated to operations.

Saving credits amounts to averaging forward over time, borrowing to averaging backward. If we can prove that we never need to borrow credits to complete the operations, then the actual time of any initial part of a sequence of operations is bounded by the sum of the corresponding amortized times. If we need to borrow but such borrowing can be paid off by the end of the sequence, then the total time of the operations is bounded by the sum of all the amortized times, although in the middle of the sequence we may be running behind. That is, the current elapsed time may exceed the sum of the amortized times by the current amount of net borrowing.

In order to keep track of saved or borrowed credits, it is generally convenient to store them in the data structure. Regions of the structure containing credits are unusually hard to access or update (the credits saved are there to pay for extra work); regions containing "debits" are unusually easy to access or update. It is important to realize that this is only an accounting device; the programs that actually manipulate the data structure contain no mention of credits or debits.

The banker's view of amortization was used implicitly by Brown and Tarjan [8] in analyzing the amortized complexity of 2, 3 trees and was developed more fully by Huddleston and Mehlhorn [15], [16] in their analysis of generalized B-trees. We can cast our analysis of a stack in the banker's framework by allocating two credits per operation. The stack manipulation maintains the invariant that the number of saved credits equals the number of stacked items. During an operation, each pop is paid for by a saved credit, the push is paid for by one of the allocated credits, and the other allocated credit is saved, corresponding to the item stacked by the push.

Our second view of amortization is that of the *physicist*. We define a *potential function* Φ that maps any configuration D of the data structure into a real number $\Phi(D)$ called the *potential* of D. We define the *amortized time* of an operation to be $t+\Phi(D')-\Phi(D)$, where t is the actual time of the operation and D and D' are the configurations of the data structure before and after the operation, respectively. With this definition we have the following equality for any sequence of m operations:

$$\sum_{i=1}^{m} t_i = \sum_{i=1}^{m} (a_i - \Phi_i + \Phi_{i-1}) = \Phi_0 - \Phi_m + \sum_{i=1}^{m} a_i,$$

where t_i and a_i are the actual and amortized times of the ith operation, respectively, Φ_i is the potential after the ith operation, and Φ_0 is the potential before the first operation. That is, the total time of the operations equals the sum of their amortized times plus the net decrease in potential from the initial to the final configuration.

We are free to choose the potential function in any way we wish; the more astute the choice, the more informative the amortized times will be. In most cases of interest, the initial potential is zero and the potential is always nonnegative. In such a situation the total amortized time is an upper bound on the total time. (This corresponds to the banker's view of amortization with no borrowing.)

The physicist's view of amortization was proposed by D. Sleator (private communication). To fit the stack manipulation example into the physicist's framework, we define the potential of a stack to be the number of items it contains. Then a stack operation consisting of k pops and one push on a stack initially containing i items has an amortized time of $(k+1)+(i-k+1)-i=2$. The initial potential is zero and the potential is always nonnegative, so m operations take at most $2m$ pushes and pops.

The banker's and physicist's views of amortization are entirely equivalent; we can choose whichever view gives us more intuition about the problem at hand. It is perhaps more natural to deal with fractional amounts of time using the physicist's view, whereas the banker's view is more concrete, but both will yield the same bounds.

3. Amortized analysis of known algorithms. Amortization has been used in the analysis of several algorithms more complicated than the stack manipulation example. In this section we shall examine three such applications. We study the examples in the order of their conceptual complexity, which coincidentally happens to be reverse chronological order.

Our first example is the "move-to-front" list updating heuristic. Consider an abstract data structure consisting of a table of n items, under the operation of accessing a specified item. We assume that the table is represented by a linear list of the items in arbitrary order, and that the time to access the ith item in the list is i. In addition, we allow the possibility of rearranging the list at any time (except in the middle of an access), by swapping any pair of contiguous items. Such a swap takes one unit of time.

We are interested in whether swapping can reduce the time for a sequence of accesses, and whether there is a simple heuristic for swapping that achieves whatever improvement is possible. These questions are only interesting if the access sequence is nonuniform, e.g. some items are accessed more frequently than others, or there is some correlation between successive accesses. Among the swapping heuristics that have been proposed are the following:

Move-to-front. After an access, move the accessed item to the front of the list, without changing the relative order of the other items.

Transpose. After an access of any item other than the first on the list, move the accessed item one position forward in the list by swapping it with its predecessor.

Frequency count. Swap after each access as necessary to maintain the items in non-decreasing order by cumulative access frequency.

The frequency count heuristic requires keeping track of access frequencies, whereas the other two rules depend only on the current access. There has been much research on these and similar update rules, the overwhelming majority of it average-case analysis [6], [7], [17], [23], [26]. All of the average-case studies known to the author are based on the assumption that the accesses are independent identically distributed random variables, i.e. for each successive access, each item i has a fixed probability p_i of being accessed. The usual measure of interest is the asymptotic average access time as a

function of $p_1, p_2, \cdots, p_n$, i.e. the average access time as m, the number of accesses, goes to infinity. (Letting m go to infinity eliminates the effect of the initial ordering.)

Under these assumptions, the optimum strategy is to begin with the items in nondecreasing order by probability and leave them that way. The law of large numbers implies that the asymptotic average access time of the frequency count heuristic is minimum, and it has long been known that move-to-front is within a factor of two of minimum [17]. Rivest [23] showed that asymptotically transpose is never worse than move-to-front, although Bitner [7] showed that it converges much more slowly to its asymptotic behavior.

Bentley and McGeogh [6] performed several experiments on real data. Their tests indicate that in practice the transpose heuristic is inferior to frequency count but move-to-front is competitive with frequency count and sometimes better. This suggests that real access sequences have a locality of reference that is not captured by the standard probabilistic model, but that significantly affects the efficiency of the various heuristics. In an attempt to derive more meaningful theoretical results, Bentley and McGeogh did an amortized analysis. Consider any sequence of accesses. Among static access strategies (those that never reorder the list), the strategy that minimizes the total access time is that of beginning with the items in decreasing order by total access frequency. Bentley and McGeogh showed that the total access time of move-to-front is within a factor of two of that of the optimum static strategy, if move-to-front's initial list contains the items in order of first access. Furthermore frequency count but not transpose shares this property.

(Note that the move-to-front heuristic spends only about half its time doing accesses; the remainder is time spent on the swaps that move accessed items to the front of the list. Including swaps, the total time of move-to-front is at most four times the total time of the optimum static algorithm.)

Sleator and Tarjan [26], using the approach presented in § 2, extended Bentley and McGeogh's results to allow comparison between arbitrary dynamic strategies. In particular, they showed that for any initial list and any access sequence, the total time of move-to-front is within a constant factor (four) of minimum over all algorithms, including those with arbitrary swapping. Thus move-to-front is optimum in a very strong, uniform sense (to within a constant factor on *any* access sequence). Neither transpose nor frequency count shares this property.

To obtain the Sleator–Tarjan result we use the physicist's view of amortization. Consider running an arbitrary algorithm A and the move-to-front heuristic MTF in parallel on an arbitrary access sequence, starting with the same initial list for both methods. Define the potential of MTF's list to be the number of inversions in MTF's list with respect to A's list, where an inversion is a pair of items whose order is different in the two lists. The potential is initially zero and always nonnegative. It is straightforward to show that, with this definition of potential, the amortized time spent by MTF on any access is at most four times the actual time spent by A on the access.

The factor of four bound can be refined and extended to allow A and MTF to have different initial lists and to allow the access cost to be a nonlinear function of list position. The problem of minimizing page faults, which is essentially a version of list updating with a nonlinear access cost, can also be analyzed using amortization. Sleator and Tarjan's paper [26] contains the details.

Another use of amortization is in the analysis of insertion and deletion in balanced search trees. A balanced search tree is another way of representing a table, more complicated than a linear list but with faster access time. Extensive discussions of search trees can be found in many computer science texts (e.g. [2], [17], [32]), and we shall assume some familiarity with their properties. Generally speaking, a table

can be stored as a search tree if the items can be totally ordered, e.g. the items are real numbers, which are orderable numerically, or strings, which are orderable lexicographically. We store the items in the nodes of a tree in symmetric order. Depending upon the exact scheme used, the items may be stored in either the internal or the external nodes, with one or several items per node. We access an item by following the path from the tree root to the node containing the item. Thus the time to access an item is proportional to the depth in the tree of the node containing it.

Balanced search trees are constrained by some sort of local *balance condition* so that the depth of an n-node tree, and thus the worst-case access time, is $O(\log n)$. Typical kinds of balanced trees include AVL or height-balanced trees [1], trees of bounded balance or weight-balanced trees [21], and various kinds of a, b trees including 2, 3 trees [2] and B-trees [5]]. (In an a, b tree for integers a and b such that $2 \leqq a \leqq \lceil b/2 \rceil$, all external nodes have the same depth, and every internal node has at least a and at most b children, except the root, which if internal has at least two and at most b children.) Indeed, the varieties of balanced trees are almost endless.

Maintaining a dynamic table (i.e. a table subject to insertions and deletions) as a balanced search tree requires storing local "balance information" at each tree node and, based on the balance information, rebalancing the tree using local transformations after each insertion or deletion. For standard kinds of balanced trees, the update transformations all take place along a single path in the tree, and the worst-case time for an insertion or deletion is $O(\log n)$. For some kinds of balanced search trees, however, the amortized update time is $O(1)$. Brown and Tarjan [8] showed that m consecutive insertions or m consecutive deletions in an n-node 2, 3 tree take $O(n+m)$ total rebalancing time, giving an $O(1)$ amortized time per update if $m = \Omega(n)$. This bound does not hold for intermixed insertions and deletions unless the insertions and deletions are far enough apart that they do not interact substantially. Maier and Salveter [20] and independently Huddleston and Mehlhorn [15, 16] showed that m arbitrarily intermixed insertions and deletions in an n-node 2, 4 tree, or indeed in an a, b tree with $a \leqq \lfloor b/2 \rfloor$, take $O(n+m)$ total rebalancing time.

To give the flavor of these results, we shall sketch an amortized analysis of insertions in balanced binary trees [31], [32], also known as "symmetric binary B-trees" [4], "red-black trees" [13], or "half-balanced trees" [22]. (See Fig. 1.) A balanced binary tree is a binary tree (each internal node has exactly two children: a left child and a right child) in which each internal node is colored either *red* or *black*, such that

(i) all paths from the root to an external node contain the same number of black nodes, and

(ii) any red node has a black parent. (In particular, the root, if internal, is black.)

Balanced binary trees are equivalent to 2, 4 trees: we obtain the 2, 4 tree corresponding to a balanced binary tree by contracting every red node into its parent.

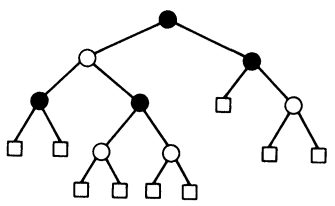


FIG. 1. *A balanced binary tree. Circles are internal nodes; squares are external. Internal nodes are solid if black, hollow if red.*

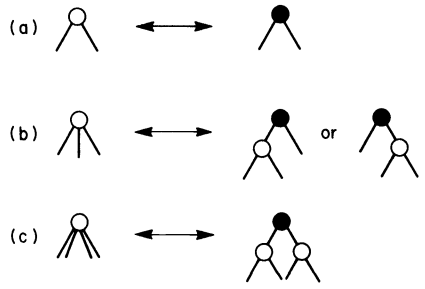


FIG. 2. *Correspondence between nodes of a* 2, 4 *tree and nodes of a balanced binary tree.*
(a) 2-*node.*
(b) 3-*node.*
(c) 4-*node.*

(See Fig. 2.) This correspondence is not one-to-one: a 2, 4 tree can correspond to several different balanced binary trees, because there are two different configurations corresponding to a 3-node (a node with three children).

We shall not go into the details of how a table can be represented by a balanced binary tree and why the depth of an n-node balanced binary tree is $O(\log n)$. (See [4], [13], [22], [31], [32].) For our purposes it suffices to know that the effect of an insertion is to convert some external node into a red internal node with two external children. This may produce a violation of property (ii). To restore (ii), we walk up the path from the violation, repeatedly applying the appropriate case from among the five cases illustrated in Figs. 3 and 4. Cases 2a, b, c are terminating: applying either of them restores (ii). Cases 1a, b are (possibly) nonterminating: after applying either of them we must look for a new violation.

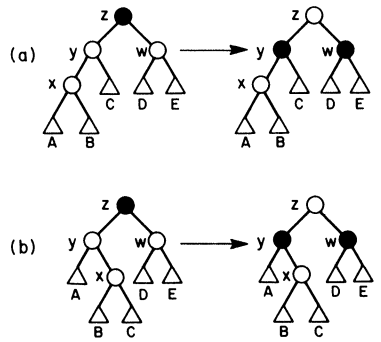


FIG. 3. *Nonterminating cases of a balanced binary tree insertion. Triangles denote subtrees whose root is either black or external. Node z may or may not be the root. Each case has a symmetric variant, not shown. After applying either case, we must check whether the parent of z is red.*
(a) *Case* 1a: *color flip.*
(b) *Case* 1b: *color flip.*

The net effect of rebalancing is to change the color of one or more nodes and possibly to change the structure of the tree by a "single rotation" (Case 2b) or a "double rotation" (Case 2c). We can prove that the total time for m consecutive insertions in a tree of n-nodes is $O(n+m)$ by using the banker's view of amortization. We maintain the invariant that every black node contains either 0, 1, or 2 credits, depending on whether it has one red child, no red children, or two red children, respectively. To satisfy the invariant initially we must add $O(n)$ credits to the tree.

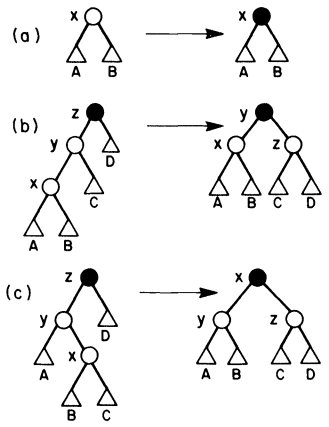


FIG. 4. *Terminating cases of a balanced binary tree insertion. Each case has a symmetric variant, not shown.*

(a) *Case* 2a: *color change at the root.*

(b) *Case* 2b: *single rotation.*

(c) *Case* 2c: *double rotation.*

(This accounts for the $O(n)$ term in the bound.) Each of Cases 2a, b, c requires the addition of $O(1)$ credits to the tree, but such a case terminates an insertion. Each of Cases 1a, b, if nonterminating, releases a credit from the tree to pay for the transformation.

This argument is an adaption of those in [15], [16], [20]. It is not hard to extend the argument to prove an $O(n+m)$ time bound for arbitrarily intermixed insertions and deletions if the deletion algorithm is suitable. (See [31], [32] for a suitable deletion algorithm.)

The $O(n+m)$ bound on update time does not take into account the time necessary to search for the positions at which the insertions and deletions are to take place. The practical importance of this bound is in situations where the search time is significantly faster than $O(\log n)$, as can occur if the search tree is augmented to allow searching from "fingers" [8], [12], [16], [19]. Search trees with fingers provide one way to take advantage of locality of reference in an access sequence, and a generalization of the argument we have sketched shows that in appropriate kinds of balanced trees with fingers, the total rebalancing time is bounded by a constant factor times the total search time, if we perform an arbitrary sequence of intermixed accesses, insertions, and deletions [16]. Brown and Tarjan [8] list several applications of such trees. The amortized approach to fingers [8], [16] is significantly simpler than the worst-case approach [12], [19].

Our third and most complicated example of amortization is in the analysis of path compression heuristics for the disjoint set union problem, sometimes called the "union-find problem" or the "equivalence problem." We shall formulate this problem as follows. We wish to represent a collection of disjoint sets, each with a distinguishing name, under two kinds of operations:

find (x): Return the name of the set containing element x.

unite (A, B): Form the union of the two sets named A and B, naming the new set A. This operation destroys the old sets named A and B.

We shall assume that the initial sets are all singletons. To solve this problem, we represent each set by a tree whose nodes are the elements in the set. Each node points to its parent; the root contains the set name. To carry out *find* (x), we follow the path

of parent pointers from x to the root of the tree containing it, and return the name stored there. To carry out *unite* (A, B), we locate the nodes containing the names A and B and make one the parent of the other, moving the name A to the new root if necessary.

This basic method is not very efficient; a sequence of m operations beginning with n singleton sets can take $O(nm)$ time, for an amortized bound of $O(n)$ per operation. We can improve the method considerably by adding heuristics to the *find* and *unite* algorithms to reduce the tree depths. When performing *unite*, we use *union by size*, making the root of the smaller tree point to the root of the larger. After performing *find* (x), we use *path compression*, changing the parent of x and all its ancestors except the tree root to be the root. (See Fig. 5.)

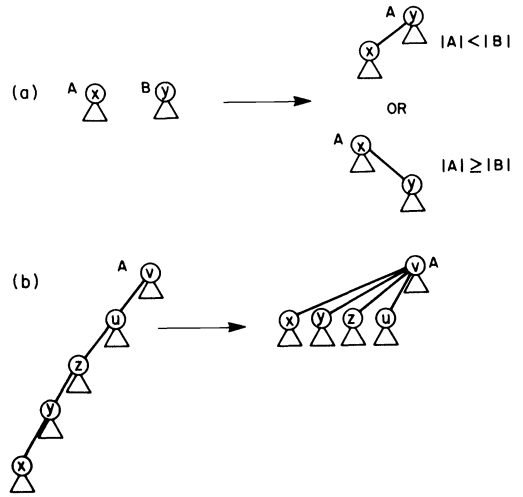


FIG. 5. *Implementation of set operations. Triangles denote subtrees.*
(a) *unite* (A, B).
(b) *find* (x).

Union by size was proposed by Galler and Fischer [11]; McIlroy and Morris devised path compression [2]. The set union algorithm with both heuristics is extremely hard to analyze. Tarjan [29] derived an $O(m\alpha(m, n))$ bound for m operations starting with n singletons, assuming $m = \Omega(n)$. Here α is a functional inverse of Ackermann's function. Tarjan's proof is a complicated amortized analysis that uses debits as well as credits. For a version of the proof in the banker's framework see [32]. The bound is tight to within a constant factor in the worst case for a large class of pointer multiplication algorithms [30]. Tarjan and van Leeuwen [33] extended the bound to allow values of m much smaller than n (the generalized bound is $O(n + m\alpha(n + m, n))$) and to cover a variant of union by size and several variants of path compression. Recently Gabow and Tarjan [10] found a linear-time algorithm for a special case of disjoint set union in which there is appropriate advance knowledge about the unions. Their algorithm combines path compression with table look-up on small sets and requires the power of a random access machine. The method apparently does not extend to the general problem.

4. New "self-adjusting" data structures. Data structures efficient in the worst case, such as the various kinds of balanced trees, have several disadvantages. The maintenance of a structural constraint, such as a balance condition, consumes both storage space (though possibly only one bit per node) and running time. Restructuring

after an update tends to be complicated, involving a number of cases. Perhaps more significantly, such data structures are inflexible in that they cannot take advantage of whatever nonuniformity there may be in the usage pattern.

The idea of amortization suggests another way to design data structures. Each time we access the structure, we modify it in a simple, uniform way, with the intent of decreasing the time required for future operations. This approach can produce a data structure with very simple access and update procedures that needs no extra storage for structural information and adapts to fit the usage pattern. An example of such a data structure is a linear list with the move-to-front rule, studied in § 3. Previous authors have used the term "self-organizing" for such data structures. We shall call them *self-adjusting*. In this section we describe two self-adjusting data structures recently invented by Sleator and Tarjan [25], [27], [28].

The first structure, the *skew heap*, is for the representation of meldable heaps, also called "priority queues" [17] and "mergable heaps" [2]. Suppose we wish to maintain a collection of disjoint sets called *heaps*, each initially containing a single element selected from a totally ordered universe, under two operations:

delete min (h): Delete and return the minimum element in heap h.
meld (h_1, h_2): Add all elements in heap h_2 to h_1, destroying h_2.

To represent a heap, we use a binary tree, each internal node of which is a heap element. We arrange the elements in *heap order*: the parent of any node is smaller than the node itself. Thus the root of the tree is the smallest element. Melding is the fundamental operation. We carry out *delete min* (h) by deleting the root of h, replacing h by the meld of its left and right subtrees, and returning the deleted node. We carry out *meld* (h_1, h_2) by walking down the right paths from the roots of h_1 and h_2, merging them. The left subtrees of nodes along these paths are unaffected by the merge. The merge creates a potentially long right path in the new tree. As a heuristic to keep right paths short, we conclude the meld by swapping left and right children of all nodes except the deepest along the merge path. (See Fig. 6.) We call the resulting data structure a *skew heap*.

Skew heaps are a self-adjusting version of the leftist queues of Crane [9] and Knuth [17]; leftist queues are heap-ordered binary trees maintained so that the right path down from any node is a shortest path to an external node. In skew heaps, the amortized times of *delete min* and *meld* are $O(\log n)$, where n is the total number of elements in the heap or heaps involved. To prove this, we define the *weight* of an internal node x to be the total number of its internal node descendants, including x itself. We define a node x to be *heavy* if it is not a root and its weight is more than half the weight of its parent. We maintain the invariant that every heavy right child has a credit. An analysis of the effect of *delete min* and *meld* gives the $O(\log n)$ bound [25], [27].

Skew heaps require only a single top-down pass for melding, in contrast to leftist heaps, which need a top-down pass followed by a bottom-up pass. If we modify skew heaps so that melding is bottom-up, we can reduce the amortized time for *meld* to $O(1)$ while retaining the $O(\log n)$ bound for *delete min*. In an amortized sense, to within a constant factor, bottom-up skew heaps are optimum among all comparison-based methods for representing heaps. Preliminary experiments indicate that they are efficient in practice as well as in theory. For details of these results, see [27].

Our second structure, the *splay tree*, is a self-adjusting form of binary search tree. Consider the table look-up problem that we solved in § 2 using a self-adjusting list. As discussed in § 3, if the items are totally orderable, we can also represent such a

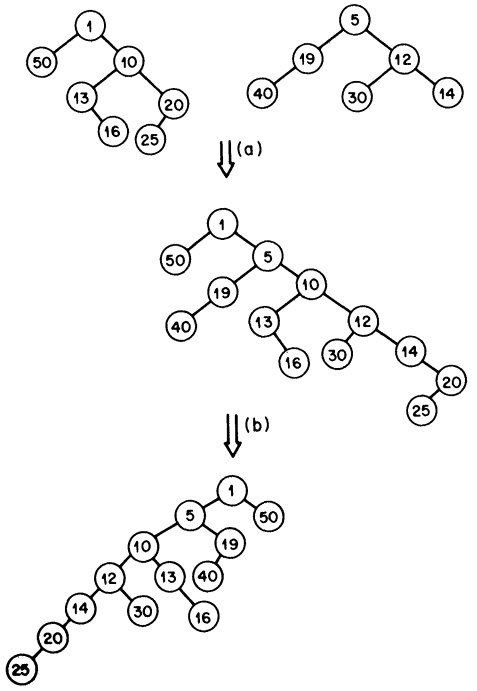


FIG. 6. *A meld of two skew heaps. External nodes are not shown.*
(a) *Merge of the right paths.*
(b) *Swapping of children along the path formed by the merge.*

table by a binary tree: Each item is an internal node of the tree, with items arranged in *symmetric order*: for any item x, all items in its left subtree are less than x and all items in its right subtree are greater than x. To access an item x, we compare x to the tree root, stop if the root is x, and otherwise proceed recursively in the left subtree if x is less than the root, in the right subtree if x is greater than the root. The time to access x is proportional to its depth in the tree.

As a heuristic to keep the tree depth small, each time we access a node x we *splay* it. To splay x, we repeatedly apply the appropriate one of the cases among those in Fig. 7, continuing until x is the root of the tree. In effect, we walk up the path from x two nodes at a time, performing rotations as we go up that move x to the root and move the rest of the nodes on the access path about halfway or more toward the root. (See Fig. 8.) We call the resulting data structure a *splay tree.*

The amortized time of an access in an n-node splay tree is $O(\log n)$. To prove this, we define the potential of a splay tree to be the sum over all internal nodes x of $\log w(x)$, where $w(x)$ is the weight of x, defined to be the number of (internal node) descendants of x, including x itself. The algorithm and the analysis extend to handle insertion, deletion, and more drastic update operations. Several variants of the splay heuristic have the same efficiency (to within a constant factor) [25], [27]. Several heuristics for search trees proposed earlier [3], [7] are not as efficient in an amortized sense.

Splay trees are not only as efficient in an amortized sense as balanced trees, but also as efficient as static optimum search trees, as an extension of the analysis shows. In this they are like lists with move-to-front updating; they automatically adapt to fit the access frequencies. The result showing that splay trees are as efficient as optimum

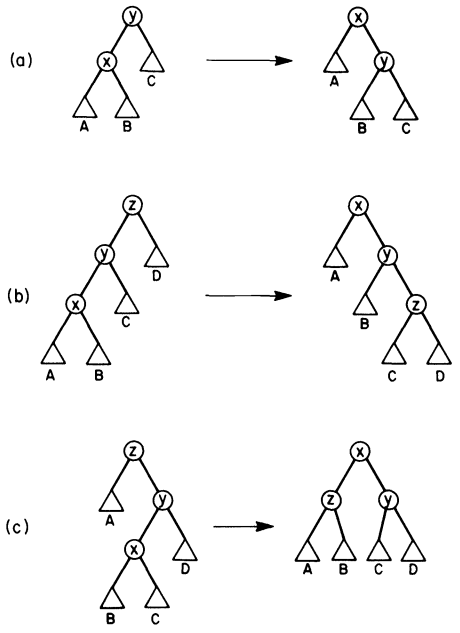


FIG 7. *Case of splay step at node x. Each case has a symmetric variant (not shown). In cases* (b) *and* (c), *if node z is not the root, the splay continues after the step.*

(a) *Terminating single rotation. Node y is the root.*

(b) *Two single rotations.*

(c) *Double rotation.*

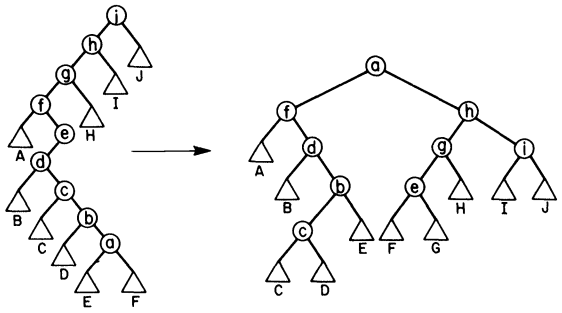


FIG. 8. *Splay at node a.*

trees is analogous to Bentley and McGeough's result comparing move-to-front with an optimum static ordering. We conjecture that a stronger result analogous to Sleator and Tarjan's result for move-to-front holds; namely splay trees minimize the amortized access time to within a constant factor among all search-tree-based algorithms. We are currently attempting to prove an appropriate formalization of this conjecture. As a special case, the truth of the conjecture would imply that splay trees are as efficient as the finger search trees mentioned in § 3, and thus that one can obtain the advantages of fingers using an ordinary search tree, without extra pointers. Details of the properties of splay trees and several applications to more elaborate self-adjusting data structures appear in [27].

5. Conclusions. We have seen that amortization is a powerful tool in the algorithmic analysis of data structures. Not only does it allow us to derive tighter bounds for known algorithms, but it suggests a methodology for algorithm development that leads to new simple, efficient, and flexible "self-adjusting" data structures. Amortization also provides a robust way to study the possible optimality of various data structures. It seems likely that amortization will find many more uses in the future.

REFERENCES

[1] G. M. ADELSON-VELSKII AND E. M. LANDIS, *An algorithm for the organization of information*, Soviet Math. Dokl., 3 (1962), pp. 1259–1262.

[2] A. V. AHO, J. E. HOPCROFT AND J. D. ULLMAN, *The Design and Analysis of Computer Algorithms*, Addison-Wesley, Reading, MA, 1974.

[3] B. ALLEN AND I. MUNRO, *Self-organizing search trees*, J. ACM, 25 (1978), pp. 526–535.

[4] R. BAYER, *Symmetric binary B-trees: data structure and maintenance algorithms*, Acta. Inform., 1 (1972), pp. 290–306.

[5] R. BAYER AND E. MCCREIGHT, *Organization of large ordered indexes*, Acta Inform., 1 (1972), pp. 173–189.

[6] J. L. BENTLEY AND C. MCGEOGH, *Worst-case analysis of self-organizing sequential search heuristics*, Proc. 20th Allerton Conference on Communication, Control, and Computing, to appear.

[7] J. R. BITNER, *Heuristics that dynamically organize data structures*, SIAM J. Comput., 8 (1979), pp. 82–110.

[8] M. R. BROWN AND R. E. TARJAN, *Design and analysis of a data structure for representing sorted lists*, SIAM J. Comput., 9 (1980), pp. 594–614.

[9] C. A. CRANE, *Linear lists and priority queues as balanced binary trees*, Technical Report STAN-CS-72-259, Computer Science Dept., Stanford University, Stanford, CA, 1972.

[10] H. N. GABOW AND R. E. TARJAN, *A linear-time algorithm for a special case of disjoint set union*, J. Comput. Sys. Sci., submitted.

[11] B. A. GALLER AND M. J. FISCHER, *An improved equivalence algorithm*, Comm. ACM, 7 (1964), pp. 301–303.

[12] L. J. GUIBAS, E. M. MCCREIGHT, M. F. PLASS AND J. R. ROBERTS, *A new representation for linear lists*, Proc. Ninth Annual ACM Symposium on Theory of Computing, 1977, pp. 49–60.

[13] L. J. GUIBAS AND R. SEDGEWICK, *A dichromatic framework for balanced trees*, Proc. Nineteenth Annual IEEE Symposium on Foundations of Computer Science, 1978, pp. 8–21.

[14] J. HOPCROFT AND R. TARJAN, *Efficient planarity testing*, J. ACM, 21 (1974), pp. 549–568.

[15] S. HUDDLESTON AND K. MEHLHORN, *Robust balancing in B-trees*, Proc. 5th GI-Conference on Theoretical Computer Science, Lecture Notes in Computer Science 104, Springer-Verlag, New York, 1981, pp. 234–244.

[16] S. HUDDLESTON AND K. MEHLHORN, *A new data structure for representing sorted lists*, Acta Inform., 17 (1982), pp. 157–184.

[17] D. E. KNUTH, *The Art of Computer Programing, Vol.* 3: *Sorting and Searching*, Addison-Wesley, Reading, MA, 1973.

[18] D. E. KNUTH, J. H. MORRIS JR. AND V. R. PRATT, *Fast pattern matching in strings*, SIAM J. Comput., 6 (1977), pp. 323–350.

[19] S. R. KOSARAJU, *Localized search in sorted lists*, Proc. Thirteenth Annual ACM Symposium on Theory of Computing, 1978, pp. 62–69.

[20] D. MAIER AND S. C. SALVETER, *Hysterical B-trees*, Inform. Proc. Letters, 12 (1981), pp. 199–202.

[21] J. NIEVERGELT AND E. M. REINGOLD, *Binary search trees of bounded balance*, SIAM J. Comput., 2 (1973), pp. 33–43.

[22] H. OLIVIE, *A new class of balanced search trees: half-balanced binary search trees*, RAIRO Inform. théorique/Theoretical Informatics, 6 (1982), pp. 51–71.

[23] R. RIVEST, *On self-organizing sequential search heuristics*, Comm. ACM, 19 (1976), pp. 63–67.

[24] P. ROSENSTIEHL AND R. E. TARJAN, *Gauss codes, planar Hamiltonian graphs, and stack-sortable permutations*, J. Algorithms, to appear.

[25] D. D. SLEATOR AND R. E. TARJAN, *Self-adjusting binary trees*, Proc. Fifteenth Annual ACM Symposium on Theory of Computing, 1983, pp. 235–245.

[26] ———, *Amortized efficiency of list update and paging rules*, Comm. ACM, to appear.

[27] D. D. SLEATOR AND R. E. TARJAN, *Self-adjusting heaps*, SIAM J. Comput., 15 (1986), to appear.
[28] ———, *Self-adjusting binary search trees*, to appear.
[29] R. E. TARJAN, *Efficiency of a good but not linear set union algorithm*, J. ACM, 22 (1975), pp. 215–225.
[30] ———, *A class of algorithms which require nonlinear time to maintain disjoint sets*, J. Comput. Sys. Sci., 18 (1979), pp. 110–227.
[31] ———, *Updating a balanced search tree in $O(1)$ rotations*, Inform. Proc. Letters, 16 (1983), pp. 253–257.
[32] ———, *Data Structures and Network Algorithms*, CBMS Regional Conference Series in Applied Mathematics 44, Society for Industrial and Applied Mathematics, Philadelphia, 1983.
[33] R. E. TARJAN AND J. VAN LEEUWEN, *Worst-case analysis of set union algorithms*, J. ACM, to appear.
[34] *Webster's New World Dictionary of the American Language*, College Edition, World, Cleveland, Ohio, 1964.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 2, April 1985

017

A COMPREHENSIVE MODEL OF DYNAMIC PROGRAMMING*

PAUL HELMAN† AND ARNON ROSENTHAL‡

Abstract. We present a new model of problems solvable by *discrete dynamic programming*. The formalism of the model is based on "nonassociative regular expressions" and a *generalized notion of comparability*. We *formally define dynamic programming* in this setting, and study its efficiency. We obtain theorems showing dynamic programming to be optimally efficient for a general class of problems. Our model generalizes previous work in that it naturally includes problems of a *nonassociative and nonsequential* nature (e.g., "parenthesization problems" and nonserial dynamic programming). A key aspect of the model is that it separates a problem's *structure* from the required *computation*. This serves to make similarities between problems more apparent.

AMS(MOS) subject classifications. 90C39, 49C20, 68C05, 68C25, 68D05

1. Introduction. Since its development by Karp and Held [9], the *discrete decision process* (DDP) has been the standard formalism for decomposable combinatorial optimization problems. The DDP is a triple $\langle \Sigma, L, f \rangle$, where

Σ is a finite alphabet of "primitive decisions",

L is a regular language of "feasible policies" over Σ,

f is the objective function $f: L \to \{\text{Reals}\}$.

The problem is to find a feasible policy which minimizes f.

Before a DDP can be solved by dynamic programming, it must be represented as a *sequential decision process* (SDP). An SDP is a finite automaton whose transition function is augmented by a cost function which agrees with the objective f on feasible policies, i.e., members of the automaton's language L. The functional equations associated with dynamic programming are derived from the SDP.

Our model extends and modifies the DDP/SDP in several fundamental ways:

(i) The DDP/SDP is inherently associative, since it is based on a formalism equivalent to regular expressions. We develop a nonassociative formalism to exploit the observation that dynamic programming does not need the associativity of string concatenation. Our model can view "parenthesization problems" (e.g., matrix multiplication, optimal binary tree problems [7]) as consisting of parenthesization subtrees, which can be "grafted" nonassociatively. The *principle of optimality* [1] is abstracted to allow this type of combination.

(ii) Unlike [2], [3], [5], [8], [9], [10], we divorce structure from computation. This enables us to separate the enumeration of the policies (i.e., generating and combining them), from the computation performed on the policies (e.g., cost evaluation and selection of the optimal). To this end, problems and enumerations are stated in a *problem structure*, independent of the actual computation. This allows for the *unification of algorithms* which perform different computations on the same problem structure, using the same enumeration scheme.

Our approach makes it apparent that a handful of distinct problem structures and enumerations underlie the vast majority of dynamic programming applications. These problem structures are clearly identified when a problem is formalized in our model.

* Received by the editors July 5, 1983, and in revised form February 10, 1984. This work was presented at the SIAM Second Conference on the Applications of Discrete Mathematics held at Massachusetts Institute of Technology, Cambridge, Massachusetts, June 27-29, 1983.

† Computer Science Department, University of New Mexico, Albuquerque, New Mexico 87131.

‡ Computer Corporation of America, Boston, Massachusetts.

(iii) Our model provides a very convenient setting for classifying enumeration schemes and analyzing their complexity. Our relevant results include a *formal definition of dynamic programming* and theorems showing dynamic programming to be *optimally efficient* for a large class of problems.

Since the model is based on a formalism that is accessible to most algorithm designers, it should be useful as an algorithm design tool as well as being theoretically interesting.

2. The model. In this section the model is formally presented. First a *problem structure* is defined, and then the connection is made between problem structures and *optimization problems.* We also view three well-known problems—the traveling salesperson problem, the matrix product chain problem, and the optimal alphabetic encoding problem—in the model's setting. The latter two are "parenthesization problems" that can naturally be solved using ordered binary trees. Dynamic programming solutions to both these problems have long been known [4], [7], but neither fits the DDP/SDP model.

The partition of a problem into two components (problem structure and optimization problem) formalizes the intuitive fact that two problems may have identical structure, though the computations required might be quite different. In our model, problems which require different computations may have identical problem structures and use identical enumerations in their solutions (e.g., the matrix product chain problem and the optimal alphabetic encoding problem). The difference in computation is captured in the statement of the optimization problem. In software terms, we can use the same main program and simply redefine a single function. We feel this unification to be a key contribution of the model.

2.1. The problem structure. Strings and regular expressions can be used to model enumerations of objects which are *associative* in nature. For example, consider paths through a graph with vertex set V. A single vertex v is a path, as is any string (i.e., element of V^*) of vertices $v_1 \cdot v_2 \cdot \cdots \cdot v_k$. If p and s are paths, then so is $(p \cdot s)$. Further, if p, s, and t are paths, then $((p \cdot s) \cdot t)$ is the *same* path as $(p \cdot (s \cdot t))$. In this sense, paths are associative in nature.

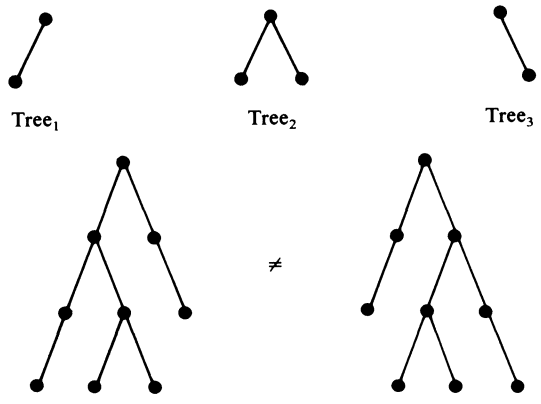
Suppose S and T are sets of paths. The *product* of the sets is defined to be

$$ST \triangleq \{(s \cdot t) | s \in S, t \in T\}.$$

Each of the sets S and T may correspond to a problem, e.g., find the shortest path in the set. Under certain conditions, it may be true that the solution to the product of the sets S and T is equal to the product of their solutions, i.e., it may be true that Solution(ST) is Solution(S) · Solution(T). The multiplication of sets S and T is a scheme for *implicitly enumerating* the paths in ST. Since $|ST|$ may be as large as $|S| * |T|$, this is a valuable solution strategy.

While many problems are associative in nature, many are not. Trees, for instance, are not. See Fig. 1.

A nonassociative analogue of strings is used to broaden the scope of our model. A *conjunct* is a finite length expression over a set A of "atoms" and $\odot$, a nonassociative analogue of the concatenation operator of strings. The conjuncts correspond to the objects enumerated for a given problem (e.g., paths through a graph, binary trees). The difference between conjuncts and strings is that the string concatenation operator must associate. As a result, conjuncts often allow for a far more natural problem statement than do strings. (A table at the end summarizes the notation introduced in this paper.)



$((\text{Tree}_1.\text{Tree}_2).\text{Tree}_3) \neq (\text{Tree}_1.(\text{Tree}_2.\text{Tree}_3))$

FIG. 1.

Formally, a *problem structure* is

$$P = \langle A, \cong, Q, \tilde{} \rangle,$$

where

A: is a finite set of *atoms.*

Every atom is a *conjunct,* and if X and Y are conjuncts so too is $(X \odot Y)$. The set of all conjuncts is denoted CONJ.

Notice that a conjunct is, technically, always fully parenthesized. We shall, however, write a conjunct as $(a_1 \odot a_2 \odot \cdots \odot a_k)$ when we do not wish to refer to a specific parenthesization.

$\cong$: is an equivalence relation on CONJ, called the *equality relation.* $\cong$ captures that distinct conjuncts may represent the same application object, e.g., if conjuncts represent paths, then for all conjuncts X, Y, Z

$$(X \odot (Y \odot Z)) \cong ((X \odot Y) \odot Z).$$

$[X]_{\cong}$ denotes the $\cong$-equivalence class containing X, i.e., $\{Y \mid X \cong Y\}$.

AXIOM PS1. $\cong$ *is preserved by* $\odot$. *That is, for all conjuncts* W, X, Y, Z

$$\langle W \cong X, Y \cong Z \rangle \Rightarrow \langle W \odot Y \cong X \odot Z \rangle.$$

We define a star free *nonassociative regular expression* (NRE) to represent a set of conjuncts as follows:

For $x \in A$, define an NRE (also denoted x) to represent the set $[x]_{\cong}$. x is called an *atomic expression.*

If Ψ and Υ are NREs representing the sets S and T, then $(\Psi + \Upsilon)$ is an NRE representing $S \cup T$.

If Ψ and Υ are NREs representing the sets S and T, then $(\Psi \odot \Upsilon)$ is an NRE representing the product of the sets S and T. The product of two sets of conjuncts is defined analogously to the product of two sets of strings. If X is a conjunct in S and Y is a conjunct in T, then the conjunct $(X \odot Y)$ is in the product. In addition, the product includes all conjuncts which are $\cong$-equivalent to such conjuncts $(X \odot Y)$. Formally, the product of the sets of conjuncts S and T, denotes $S \otimes T$, is defined as

$$S \otimes T \triangleq \bigcup_{\substack{X \in S \\ Y \in T}} [(X \odot Y)]_{\cong}.$$

We will on occasion use Rep (Ψ) to denote the set represented by Ψ, and speak of Ψ as enumerating the conjuncts in this set. If Rep (Ψ) = Rep (Υ) we say the two NREs are congruent and write $\Psi \equiv \Upsilon$.

Every NRE is a *subexpression* of itself. In addition,

$$\text{subexp}(\Psi + \Upsilon) = (\{(\Psi + \Upsilon)\} \cup \text{subexp}(\Psi) \cup \text{subexp}(\Upsilon)) \text{ and}$$

$$\text{subexp}(\Psi \odot \Upsilon) = (\{(\Psi \odot \Upsilon)\} \cup \text{subexp}(\Psi) \cup \text{subexp}(\Upsilon)).$$

Notice that all conjuncts are NREs, i.e., $(a_1 \odot a_2 \odot \cdots \odot a_k)$ is both a conjunct and an NRE. The following lemma establishes that, under the above definition, a conjunct represents its $\cong$ class.

LEMMA 1. *If X is a conjunct, then* Rep $(X) = [X]_{\cong}$.

The proof appears in the appendix.

Q: is a *distinguished* NRE, representing the set of conjuncts which must be enumerated. Q is in "disjunctive normal form" i.e.,
$Q = \Psi_1 + \cdots + \Psi_N$, where each Ψ_i is a conjunct (i.e., contains only $\odot$ operators) and

$$\langle \Psi_i \cong \Psi_k \rangle \Rightarrow \langle i = k \rangle.$$

Q thus *explicitly enumerates* the conjuncts in Rep (Q).

$\tilde{}$: is an equivalence relation on CONJ, called the *comparability relation.* $\cong$ refines $\tilde{}$.

AXIOM PS2. $\tilde{}$ *is preserved by* $\odot$. *That is, for all conjuncts W, X, Y, Z*

$$\langle W \mathbin{\tilde{}} X, Y \mathbin{\tilde{}} Z \rangle \Rightarrow \langle W \odot Y \mathbin{\tilde{}} X \odot Z \rangle.$$

Distinguished NRE Q is assumed to represent an entire $\tilde{}$-class.
$[X]$ denotes the $\tilde{}$-equivalence class containing X, i.e., $\{Y | X \mathbin{\tilde{}} Y\}$.
If an NRE represents a set of $\tilde{}$-equivalent conjuncts, we say it is a $\tilde{}$-NRE.

2.2. Optimization problems. The comparability relation $\tilde{}$ generalizes Bellman's sense of comparability [1]. In this section we formalize how comparability can be exploited to solve discrete optimization problems.

Let $P = \langle A, \cong, Q, \tilde{} \rangle$ be a problem structure. A (*discrete*) *optimization problem* (with structure P) is

$$\theta = \langle P, \text{choice} \rangle$$

where choice is a function which maps a set of $\tilde{}$-comparable conjuncts into a conjunct of that set. The choice function is to be interpreted as choosing an optimal (e.g., one which minimizes an objective function) conjunct from the set. We extend choice to $\tilde{}$-NRE Ψ by choice (Ψ) = choice (Rep (Ψ)). Notice it follows from Lemma 1 that if X and Y are $\cong$-equivalent conjuncts, then choice (X) = choice (Y), and is a member of their $\cong$ class.

The *optimization task for θ* is to compute choice (Q).

The optimization task is interpreted as the selection of the optimal conjunct from those in the set represented by Q.

Let Δ_{choice} denote the operation of choosing between two $\tilde{}$-comparable conjuncts, i.e.,

$$X \underset{\text{choice}}{\Delta} Y = \text{choice}((X + Y)) \quad \text{for } X \mathbin{\tilde{}} Y.$$

The interpretation that the choice function selects the optimal conjunct from a set of $\tilde{}$-comparable conjuncts requires that Δ_{choice} totally orders any such set. Axiom C1 imposes the necessary structure on choice (see Lemma 2 in the appendix).

AXIOM C1. *If* $(\Psi+\Upsilon)$ *is a* $\tilde{}$-NRE *then* choice $((\Psi+\Upsilon)) = \text{choice}(\Psi)\ \Delta_{\text{choice}}$ choice (Υ). See Fig. 2.

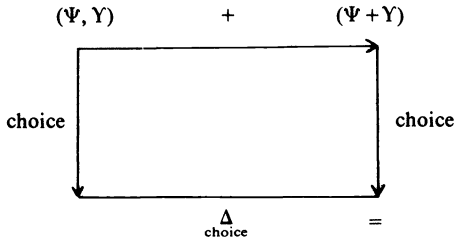


FIG. 2. *Commutativity diagram for Axiom* C1.

One solution to the optimization task is obtained by replacing in Q each occurrence of + by Δ_{choice}. The resulting expression is a computation Comp (Q) (computation is formally defined below), and Axiom C1 implies that Comp (Q) produces choice (Q). This computation is the brute force enumeration of all the conjuncts in Q and, in general, is infeasible. The ability to compute choice (Q) by implicit enumeration of the conjuncts is derived from the notion of comparability, as captured in the second axiom governing choice. This axiom is our abstraction of *Bellman's principle of optimality.*

AXIOM C2. *If* Ψ *and* Υ *are each* $\tilde{}$-NRE*s then* choice $((\Psi \odot \Upsilon)) \cong \text{choice}(\Psi) \odot$ choice (Υ). See Fig. 3.

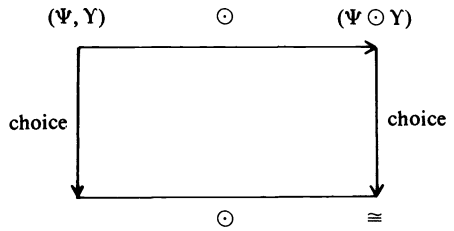


FIG. 3. *Commutativity diagram for Axiom* C2.

This axiom is the basis of *implicit enumeration schemes.* It states a sufficient condition for insuring that the optimal conjunct in the product of the sets (represented by) Ψ and Υ be the product of the optimal conjuncts in Ψ and Υ. We are thus provided with a means, which does not require the explicit enumeration of every conjunct in a set, of determining the optimal conjunct in that set.

We now formalize the notion of a *computation.*

If every subexpression of Ψ is a $\tilde{}$-NRE, we say that Ψ is *interpretable.* The replacement in Ψ of each occurrence of + by Δ_{choice} yields a *computation over* (CONJ, $\odot$, Δ_{choice}). Denote this computation by Comp (Ψ). The operators of Comp (Ψ) are $\odot$ and Δ_{choice}, and the operands and results are conjuncts. Notice that Ψ needs to be interpretable to guarantee that Δ_{choice} is defined for each pair of its operands.

The two choice axioms insure that, for interpretable NRE Ψ, Comp (Ψ) produces the conjunct (down to $\cong$-equality) which choice selects from among the conjuncts in Rep (Ψ).

LEMMA 3. *If* Ψ *is an interpretable* NRE, *then* Comp (Ψ) *produces a conjunct from the same* $\cong$*-class as* choice (Ψ).

Proof. The proof is a straightforward induction on the number of operations ($\odot$ and Δ_{choice}) in Comp (Ψ).

Since conjuncts from the same $\cong$-class represent the same application object, we shall say simply that Comp (Ψ) produces choice (Ψ) (if Comp (Ψ) produces conjunct X, then choice (X) is in fact choice (Ψ)).

Thus every interpretable NRE Ψ corresponds to a computation, Comp (Ψ), which begins with atoms from A and computes choice (Ψ). Since distinguished NRE Q is interpretable, Comp (Q) always produces a solution to the optimization task, but typically requires too many Δ_{choice} and $\odot$ operations. The algorithm designer must find an interpretable $Q' \equiv Q$ such that Comp (Q') is efficient. This problem is addressed in § 3.

Before presenting our examples, we point out that problems whose required computation is not an optimization can be handled by a generalized version of this model. In a forthcoming paper, we show that nonoptimization problems (e.g., context free language recognition) can be included in the theory by mapping enumerations into a generalized "computation domain". Analogues of Axioms C1 and C2 (Bellman's principle) allow the same enumerations which solve optimization problems to also solve nonoptimization problems. Thus, for example, the dynamic programming enumeration for the parenthesization problems presented in § 3 can be used to solve the context-free language recognition problem. In addition, the generalized model can be used for optimization problems, allowing a cost component to be included in the computation (i.e., choice also gives the cost of the optimal conjunct).

2.3. Examples of problem structures and optimization problems. We shall consider two problem structures: the first can be used for the traveling salesperson problem and its variants (and with minor modification for shortest path problems), the second for parenthesization problems. By use of appropriate choice functions we derive the traveling salesperson problem from the first problem structure, and the matrix product chain and optimal alphabetic encoding problems from the second. In § 3 dynamic programming solutions to these problems are presented.

Traveling salesperson (TS) *problem.* A problem structure which can be used for the TS problem (and its variants) over a directed graph $G = (V, E)$ with positive edge costs COST $(\langle v, w\rangle)$ is:

$$P = \langle A, \cong, Q, \tilde{} \rangle,$$

where

A is the set of vertices $V = \{v_0, v_1, \cdots, v_N\}$.

$\cong$ conjuncts which differ only by parenthesization represent the same application object, i.e.,

For every $X, Y, Z \in \text{CONJ}$, $(X \odot (Y \odot Z)) \cong ((X \odot Y) \odot Z)$.

We may therefore omit parentheses in conjuncts and view conjunct $v_1 \odot \cdots \odot v_k$ as the path $v_1, \cdots, v_k$.

Q $Q = \Psi_1 + \Psi_2 + \cdots + \Psi_{N!}$, where the Ψ_i are the tours through V which begin and terminate at v_0 (any other vertex in V would also suffice). For example, $\Psi_i = v_0 \odot v_{i_1} \odot \cdots \odot v_{i_N} \odot v_0$.

$\sim$ two paths are comparable if they pass through the same set of vertices and their terminal points correspond. Formally, if $X = x_1 \odot \cdots \odot x_j$ and $Y = y_1 \odot \cdots \odot y_k$ $\sim$
then $X \tilde{} Y$ iff $\langle x_1 = y_1$ and $x_j = y_k \rangle$
$\wedge \langle x_2, \cdots, x_{j-1}$ and $y_2, \cdots, y_{k-1}$ differ from each other by only a permutation (and thus $j = k) \rangle$.

The (standard) TS problem is obtained by defining choice (Ψ) to return a minimum cost path from those in Rep (Ψ), where cost is the sum of the edge costs on the path

$$\text{cost}(w_1 \odot \cdots \odot w_k) = \sum_{i=1}^{k-1} \text{COST}(\langle w_i, w_{i+1} \rangle).$$

Thus choice (Q) is a minimum cost tour through G.

In case of tie, a lexicographical ordering is used. Technically, choice must return a member of the $\cong$-class of the least cost path, i.e., some parenthesization of it. We arbitrarily define it to return the left to right parenthesization, $(((a \odot b) \odot c) \odot d)$.

Parenthesization problems. Many problems can be viewed as requiring the selection of a parenthesization tree for a string $z_1 z_2 \cdots z_N$. The following problem structure can be used for problems of this nature:

$$P = \langle A, \cong, Q, \tilde{} \rangle,$$

where

A is the alphabet $\{z_1, z_2, \cdots, z_N\}$.

$\cong$ is the identity, i.e., distinct conjuncts are never equivalent. We may thus view conjuncts as labeled binary trees: Atom z_i is the tree with a single node labeled z_i. Conjunct $(X \odot Y)$ is the labeled binary tree in which X is the left subtree and Y is the right subtree.

Q $Q = (z_1 \odot \cdots \odot z_N)_1 + \cdots + (z_1 \odot \cdots \odot z_N)_i + \cdots + (z_1 \odot \cdots \odot z_N)_K$, where the $(z_1 \odot \cdots \odot z_N)_i$ are all the possible parenthesizations of $z_1 \odot \cdots \odot z_N$. Q represents the set of all ordered binary trees over A (such that each internal node has two children).

$\tilde{}$ conjuncts which are different parenthesizations of the same string are comparable, i.e., if $X = (x_1 \odot \cdots \odot x_n)_1$ and $Y = (y_1 \odot \cdots \odot y_m)_2$ then $X \tilde{} Y$ iff $n = m$ and $x_i = y_i$, $1 \leqq i \leqq n$.

Two problems which use this problem structure are the matrix product chain problem and the optimal alphabetic encoding problem.

Matrix product chain problem. Let $z_1 X z_2 X \cdots X z_N$ be a chain of matrix products. The problem is to associate the matrices so as to minimize the number of multiplications required to compute the product. For $0 \leqq i \leqq N$, DIM (i) represents the dimensions of the matrices, i.e., z_i has DIM ($i-1$) rows and DIM (i) columns. For simplicity, assume the cost of multiplying an $N \times M$ matrix with an $M \times P$ matrix is NMP multiplications; no conceptual change is required if a technique such as Strassen multiplication is used. The problem is to construct a "multiplication tree," an ordered binary tree over $z_1, \cdots, z_N$, which corresponds to an optimal association for the matrix chain.

For example, if $N = 4$ and DIM (0) = 2, DIM (1) = 3, DIM (2) = 4, DIM (3) = 2, DIM (4) = 5, then a possible multiplication tree is shown in Fig. 4. Each internal node is labeled with the dimensions of the matrix it represents, along with the cost of obtaining that matrix under the association scheme. The cost of the tree is the cost of the root. To model this problem, choice bases its selection on the cost function described above, i.e., cost $((w_1 \odot \cdots \odot w_k))$ is the number of multiplications required to compute

the chain $w_1Xw_2X\cdots Xw_k$ under the association scheme given by the tree $(w_1 \odot \cdots \odot w_k)$. Thus choice (Q) is an ordered tree over $\{z_1, z_2, \cdots, z_N\}$ corresponding to an optimal association for the chain $z_1Xz_2X\cdots Xz_N$.

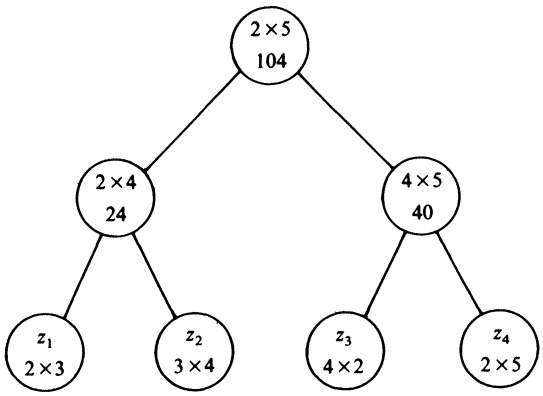


FIG. 4. *Matrix multiplication tree.*

Optimal alphabetic encoding. Given an alphabet $\{z_1, z_2, \cdots, z_N\}$ (with $z_1 < z_2 < \cdots < z_N$), let p_i denote the probability that character i will be chosen. The problem is to construct an order binary tree over $\{z_1, z_2, \cdots, z_N\}$ with minimum expected path length. An ordered binary tree corresponds to an encoding in the obvious way. The encoding has the required properties that no code is a prefix of another, and code $(z_i) <_{\text{lex}}$ code (z_j) if $i<j$. For example, let the alphabet be $\{A, B, C, D\}$. Then the encoding $\langle A=0, B=10, C=110, D=111\rangle$ is represented as in Fig. 5. To model this problem, choice bases its selection on the expected path length, i.e.,

$$\text{cost}((w_1 \odot \cdots \odot w_k)_i) = \sum_{i=1}^{k} \Pr\{w_i\} * (\text{depth of } w_i \text{ in the tree } ((w_1 \odot \cdots \odot w_k)_i)).$$

Thus choice (Q) is an ordered tree over $\{z_1, z_2, \cdots, z_N\}$ with minimum expected path length.

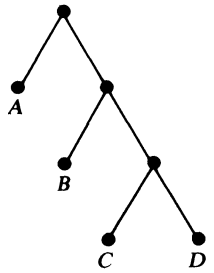


FIG. 5. *Encoding tree.*

3. Dynamic programming enumerations. In this section we formally define the class of dynamic programming enumerations, prove theorems demonstrating its optimality, and present dynamic programs for the two problem structures from § 2.3.

Any enumeration which is stated in the framework of the model is oblivious to problem specific data in the sense that the enumeration can be modeled as a circuit. This restriction immediately eliminates some powerful heuristics, e.g., branch and bound techniques. The complexity results which we obtain are relative to the class of

oblivious enumerations. In this sense the model captures the limits of oblivious computations; there are some well-known nonoblivious techniques [7] which can speed up the solutions, but which are beyond the power of computations which exploit only the axioms of our model.

3.1. The class of dynamic programming enumerations. The formalism of the model can capture many enumeration strategies. The definition of the class of dynamic programming enumerations is based on the notion that once a dynamic program starts work on a comparable set of conjuncts, it will continue on this comparability class until all conjuncts from the class have been enumerated [6], [11]. That is, it will $\odot$-multiply only NREs which have enumerated entire comparability classes. This definition is broad enough to include most of what the literature classifies as dynamic programming.

DEFINITION. An NRE of the form $(\Psi \odot \Upsilon)$ is called a *product.* A subexpression of an NRE which is a product is called a *subproduct.*

MAJOR DEFINITION. An interpretable NRE Φ (or its corresponding computation Comp (Φ)) is a *dynamic program* if for all subproducts $(\Phi_1 \odot \Phi_2)$ of Φ, Φ_1 and Φ_2 both represent *entire* $\tilde{}$-comparability classes. (To exclude redundant computations from our definition of dynamic programming, we add the additional constraint that a dynamic program *not contain* any subexpressions $(\Phi_1 + \Phi_2)$ where Rep (Φ_1) and Rep (Φ_2) are nested.)

Our definition of dynamic programming requires that if the "solution" to Φ_1 is to be combined with the $\odot$ operation, it must represent an entire $\tilde{}$-class. Since all subexpressions must be $\tilde{}$-NREs, this permits the $\odot$ combination of solutions to only *maximal* meaningful subproblems. This requirement corresponds to the notion of dynamic programming in the literature, and also makes good computational sense.

As a simple illustration of an enumeration excluded by our definition of dynamic programming, consider the problem structure of the traveling salesperson problem and the enumeration Q itself. Q explicitly enumerates all the tours. It is not a dynamic program (when $|V| > 4$) since it contains subexpressions of the form

$$(v_0 \odot v_1 \odot v_2 \odot v_3) \odot v_4,$$

and $(v_0 \odot v_1 \odot v_2 \odot v_3)$ represents a proper subset of the $\tilde{}$-class $\{(v_0 \odot v_1 \odot v_2 \odot v_3), (v_0 \odot v_2 \odot v_1 \odot v_3)\}$.

On the other hand, the expression

$$((v_0 \odot v_1 \odot v_2 \odot v_3) + (v_0 \odot v_2 \odot v_1 \odot v_3)) \odot v_4$$

would not violate the condition.

We construct as follows a dynamic programming enumeration Φ which solves the traveling salesperson problem.

Let $S = \{w, w_1, w_2, \cdots, w_m\}$ be any subset of $V - \{v_0\}$, and let S^w denote $S - \{w\}$. The $\tilde{}$-NRE $\Phi_{S,w}$ will represent the set of all paths which begin at v_0, visit each vertex w_i in S exactly once, and terminate at w. $\Phi_{S,w}$ is defined by

$$\Phi_{S,w} = \begin{cases} (v_0 \odot w) & \text{if } S = \{w\}, \\ ((\Phi_{S^w,w_1} \odot w) + \cdots + (\Phi_{S^w,w_m} \odot w)) & \text{if cardinality of } S \text{ is greater than 1.} \end{cases}$$

Let V' denote $V - \{v_0\}$. Then $\Phi_{V',v}$ represents the set of all paths starting at v_0, passing through all the vertices, and terminating at v. Since each tour must terminate at v_0, the final step is to define Φ by

$$\Phi = ((\Phi_{V',v_1} \odot v_0) + \cdots + (\Phi_{V',v_N} \odot v_0)).$$

The actual dynamic programming computation, Comp (Φ), is obtained by replacing in Φ each occurrence of + by the selection operator Δ_{choice}.

The identical dynamic programming enumeration solves both the matrix product chain and optimal alphabetic encoding problems.

A is the alphabet $\{z_1, z_2, \cdots, z_N\}$. The $\tilde{}$-NRE $\Phi_{i,k}$ will represent the set of all ordered binary trees over $\{z_i, z_{i+1}, \cdots z_k\}$. $\Phi_{i,k}$ is defined as follows.

For $1 \leqq i \leqq k \leqq N$,

$$\Phi_{i,i} = z_i$$

$$\Phi_{i,k} = ((\Phi_{i,i} \odot \Phi_{i+1,k}) + (\Phi_{i,i+1} \odot \Phi_{i+2,k}) + \cdots + (\Phi_{i,k-1} \odot \Phi_{k,k})).$$

The NRE $\Phi_{1,N}$ is a dynamic programming enumeration.

The actual dynamic programming computations, Comp (Φ), are obtained by replacing in Φ each occurrence of + by the selection operator Δ_{choice}. The differences in the computations for the two problems are captured by choice's selection criteria.

3.2. The uniqueness and optimality of dynamic programming. Our model allows us to state and prove the following theorem, which implies the uniqueness of the dynamic program for parenthesization problems.

THEOREM 1. *For each problem structure in which $\cong$ is the identity (i.e., distinct conjuncts are never equivalent), there exists a unique (down to the commutation and association of + operations) dynamic program.*

The proof appears in the appendix.

The straight-line complexity of any NRE [computation] is measured by the number of $\odot$ and $+[\Delta_{\text{choice}}]$ operations, with common subexpressions counted only once.

A strong optimality result has been obtained, applicable to any problem structure in which $\cong$ is the identity.

THEOREM 2. *If $\cong$ is the identity (i.e., distinct conjuncts are never equivalent), Ψ is an interpretable* NRE *with $\Psi \equiv Q$, and Φ is the unique dynamic program, then*

(1) *number of +'s in $\Phi \leqq$ number of +'s in Ψ.*

(2) *number of $\odot$'s in $\Phi \leqq$ number of $\odot$'s in Ψ.*

The proof appears in the appendix.

For example, the dynamic program presented for the parenthesization problems is optimal within the context of our model.

4. Future directions and summary. We see three areas for extending our results. The first is broadening the complexity results to problem structures in which $\cong$ includes more than $X \cong X$ (e.g., when $\odot$ is associative or commutative). We strongly suspect that under some very reasonable assumptions on $\cong$ and the comparability relation $\tilde{}$, dynamic programs can be shown to be optimal within the context of our model. (We do know that some restrictions on $\cong$ and $\tilde{}$ will be necessary, as we have been able to construct examples of problem structures for which dynamic programming is not optimal.)

All our optimality results are with respect to computations which can be stated in the model. A second area for future study are the classes of computations beyond the model's computational power. A major restriction on the computations stated in our model is that they be oblivious, that is to say the operations performed are not influenced by the data and can thus be modeled by a circuit. This rules out, for example, branch and bound enumerations. We are currently attempting to extend the model so as to be able to include these schemes, and thus provide a common formalism for the statement and analysis of dynamic programming and branch and bound enumerations.

Finally, the generalization of strings to conjuncts allows the modeling of parallel algorithms. Though parallel dynamic programming and branch and bound algorithms have appeared in the literature, they cannot be formalized in the setting of the DDP/SDP. In terms of the DDP/SDP, strings X and Y are comparable if they take the automaton from the initial state to the same state q. If the string X is the least cost string among the set S of comparable strings then, for any $a \in \Sigma$, $X \cdot a$ is the least cost string among $\{Y \cdot a \mid Y \in S\}$. This provides a means of comparing only policy initial segments. It allows only for serial, one step at a time combinations.

Our abstraction of the principle of optimality allows nonsequential combinations of objects. In the context of optimization problems, if X is the least cost conjunct in Rep (Ψ_1) and Y is the least cost conjunct in Rep (Ψ_2) (assuming the Ψ_1 are $\tilde{}$-NREs), then $X \odot Y$ is the least cost conjunct in Rep $(\Psi_1 \odot \Psi_2)$.

This allows Ψ_1 and Ψ_2 to be "solved" in parallel, and then for their solutions to be combined. We will explore extentions to our model that would permit such computations to be formally stated and analyzed.

Summary. The model makes three fundamental contributions:

1. Neither dynamic programming nor our formalism requires the associativity of string concatenation. Nonassociative "parenthesization problems" are easily modeled by an operation which grafts trees together.

2. We model implicit enumerations of conjuncts in one algebra, and evaluate conjuncts in another. This allows us to formally identify the commonality between algorithms which make identical assumptions about problem structure and use the same enumeration scheme, but perform different evaluations. The formalism developed in this paper allows for the statement of only optimization problems, i.e., problems which require the selection of a conjunct in Rep (Q). In a forthcoming paper, we show that the same "implicit enumeration" strategy, based on a generalization of Axiom C2, can apply even when the underlying computation is not a minimization (e.g., context free language recognition, probability calculations). The problem structure is unchanged, and the computation is performed in a new algebra. Theorems 1 and 2 directly generalize to this new class of problems.

3. The model provides a structure for studying implicit enumeration schemes and for analyzing their complexity. Our relevant results include a formal definition of dynamic programming, and a theorem showing that for an enormous class of problems, dynamic programming is optimally efficient within the context of our model.

Appendix—Proofs of theorems and lemmas.

LEMMA 1. *If X is a conjunct, then* Rep $(X) = [X]_{\cong}$.

Proof. The proof is by induction on k = the number of $\odot$'s in X.

$k = 0$. Then X is an atom and Rep $(X) = [X]_{\cong}$ by definition.

Assume true for $k < n$.

Show true for $k = n$. Let X be a conjunct containing n $\odot$'s. Then there exist conjuncts A and B, each containing fewer than n $\odot$'s, such that $X = (A \odot B)$. Thus we must show

$$\bigcup_{\substack{W \in \text{Rep}(A) \\ Y \in \text{Rep}(B)}} [(W \odot Y)]_{\cong} = [A \odot B]_{\cong}.$$

By the inductive hypothesis, Rep $(A) = [A]_{\cong}$ and Rep $(B) = [B]_{\cong}$. Thus we must show

$$\bigcup_{\substack{W \in [A]_{\cong} \\ Y \in [B]_{\cong}}} [(W \odot Y)]_{\cong} = [A \odot B]_{\cong}.$$

That the LHS contains the RHS is obvious. To show containment in the opposite direction, let

$$Z \in \bigcup_{\substack{W \in [A]_{\cong} \\ Y \in [B]_{\cong}}} [(W \odot Y)]_{\cong}.$$

Then $\exists W \cong A$, $\exists Y \cong B$ such that $W \odot Y \cong Z$. But since $\odot$ preserves $\cong$ (Axiom PS1), we have that $W \odot Y \cong A \odot B$. Since $\cong$ is an equivalence relation, $Z \cong A \odot B$. □

Define $\ll$ as:

$$[X]_{\cong} \ll [Y]_{\cong} \quad \text{iff} \quad X \tilde{} Y \text{ and choice } ((X + Y)) \cong X.$$

That $\ll$ is well defined is a consequence of the fact that $\cong$ refines $\tilde{}$ and that the value of choice (Ψ) depends only on the set Ψ represents.

LEMMA 2. *$\ll$ is a partial order on the $\cong$-classes of* CONJ. *Two $\cong$-classes are ordered iff their conjuncts are $\tilde{}$-comparable.*

Proof. The only nontrivial part of the proof is to establish that $\ll$ is transitive. Suppose $[X]_{\cong} \ll [Y]_{\cong}$ and $[Y]_{\cong} \ll [Z]_{\cong}$. Then by definition of $\ll$,

$$\text{choice } ((X + Y)) \cong X \quad \text{and} \quad \text{choice } ((Y + Z)) \cong Y.$$

But then we have choice $(((X + Y) + Z)) = \text{choice } ((X + Z))$ by Axiom C1 and also choice $(((X + Y) + Z)) = \text{choice } ((X + (Y + Z))) \cong X$. Therefore choice $((X + Z)) \cong X$ and thus $[X]_{\cong} \ll [Z]_{\cong}$.

Proofs of the theorems. Before Theorems 1 and 2 can be proved, a few intermediate results must be obtained.

DEFINITION. Conjunct X is a *factor* of conjunct Y if there exists atoms $a_1, a_2, \cdots, a_L, a_{L+1}, \cdots, a_R$ and a parenthesization i of

$$a_1 \odot a_2 \odot \cdots \odot a_L \odot (X) \odot a_{L+1} \odot \cdots \odot a_R$$

such that $(a_1 \odot a_2 \odot \cdots \odot a_L \odot (X) \odot a_{L+1} \odot \cdots \odot a_R)_i = Y$. X is a *proper factor* of Y if it is a factor and $X \neq Y$.

Notice that X "is a factor of" Y is a partial order on CONJ.

DEFINITION. $\tilde{}$-class C' is a [*proper*] *descendent* of $\tilde{}$-class C if there exist $X \in C'$ and $Y \in C$ such that X is a [proper] factor of Y.

Notice that, since $\odot$ preserves $\tilde{}$, if some $X \in C'$ is a proper factor of some $Y \in C$, then each element of C' is a proper factor of some element of C.

An NRE can represent a set of conjuncts from only finitely many different $\cong$-classes. Hence, when $\cong$ is the identity, an NRE (in particular Q) can represent a set containing only a finite number of conjuncts. In what follows, we are interested only in $\tilde{}$-classes which are descendents of Rep (Q). Since [Rep (Q) is finite] implies [all descendents of Rep (Q) are finite], when $\cong$ is the identity we restrict attention to finite $\tilde{}$-classes.

LEMMA APX1. *Let $\cong$ be the identity. Then [$C' < C$ iff C' is a descendent of C] is a partial order.*

Proof. Reflexive is trivial.

Anti-symmetric. Suppose $C' < C$, $C < C'$, and $C' \neq C$. Since $C' < C$, $\exists X \in C'$ and atoms $a_1, a_2, \cdots, a_L, a_{L+1}, \cdots, a_R$ ($R > 0$ since $C' \neq C$) and a parenthesization such that $(a_1 \odot a_2 \odot \cdots \odot a_L \odot (X) \odot a_{L+1} \odot \cdots \odot a_R) \in C$. (Abbreviate this to $A_L X A_R \in C$.) But since $\odot$ preserves $\tilde{}$, for $\forall X_i \in C'$, $A_L X_i A_R \in C$.

Similarly, $\exists A'_{L'}, A'_{R'}$ (using the notation from above) such that for $\forall Y_j \in C$, $A'_{L'} Y_j A'_{R'} \in C'$. But then we have

$$(A_L X A_R) \in C$$

$$(A'_{L'}(A_L X A_R) A'_{R'}) \in C'$$

$$(A_L(A'_{L'}(A_L X A_R) A'_{R'}) A_R) \in C$$

$$\vdots$$

violating the fact that classes are finite when $\cong$ is the identity.

Transitive. Follows from the fact that "is a factor of" is transitive. □

The *rank* of a class is well defined when $\cong$ is the identity, since in this case $<$ is a partial order.

DEFINITION. A class C is *minimal* if $C' < C \rightarrow C' = C$.

Notice that if C is minimal, then all of its conjuncts are atoms.

DEFINITION. The rank of C is defined recursively:

If C is minimal, then Rank $(C) = 1$.

If C is not minimal, then Rank $(C) = 1 + \max \{\text{Rank}(C') \mid C' \text{ is a proper descendent of } C\}$.

Notice that $C \neq C'$ and $C' < C \rightarrow \text{Rank}(C') < \text{Rank}(C)$.

THEOREM 1. *For each problem structure in which* $\cong$ *is the identity* (*i.e., distinct conjuncts are never equivalent*), *there exists a unique* (*down to the commutation and association of* + *operations*) *dynamic program.*

Proof. We prove that if C is any $\tilde{}$-class then there is a unique dynamic program which represents C. The proof will be by induction on $r = \text{Rank}(C)$.

$r = 1$. Then $C = \{a_1, a_2, \cdots, a_n\}$ and $a_1 + a_2 + \cdots + a_n$ is a dynamic program which represents C. Any NRE whose terms are not exactly these same atoms (unless it contains redundant atoms and is thus not a dynamic program) must represent a different set (since $\cong$ is the identity), and thus this dynamic program is unique down to order of +'s.

Assume true for all $r < k$.

Show true for $r = k$. Let Rank $(C) = k$. Partition C into conjuncts which are atoms and conjuncts which are not atoms, i.e., $C = \{(L_1 \odot R_1), (L_2 \odot R_2), \cdots, (L_k \odot R_k)\} \cup \{a_1, a_2, \cdots, a_m\}$. It follows from the fact that $\odot$ preserves $\tilde{}$ and that C is an entire class that $\cup_{i=1}^{k} (L_i \odot R_i) = \cup_{i=1}^{k} ([L_i] \otimes [R_i])$. Renumbering the L's and R's to eliminate redundant pairs of classes, we have $C = \cup_{i=1}^{q} ([L_i] \otimes [R_i]) \cup \{a_1, \cdots, a_m\}$. Since Rank $([L_i])$ and Rank $([R_i])$ are both less than k, the inductive hypothesis implies that there are unique dynamic programs Φ_{L_i} and Φ_{R_i} representing $[L_i]$ and $[R_i]$ respectively $(1 \leqq i \leqq q)$. Therefore

$$\Phi = ((\Phi_{L_1} \odot \Phi_{R_1}) + \cdots + (\Phi_{L_q} \odot \Phi_{R_q}) + a_1 + \cdots + a_m)$$

is a dynamic program which represents C.

Further, suppose Φ' is another dynamic program representing C, with $\Phi' = ((\Phi'_{L_1} \odot \Phi'_{R_1}) + \cdots + (\Phi'_{L_{q'}} \odot \Phi'_{R_{q'}}) + a'_1 + \cdots + a'_{m'})$. Suppose that Φ and Φ' differ by more than order of + operations. There are two possibilities. The first is that they differ on their atoms, i.e., $a_1 + a_2 + \cdots + a_m$ and $a'_1 + a'_2 + \cdots + a'_{m'}$ differ by more than order of + operations. WLOG, assume a_i is in the first sum, but not the second. Since $\cong$ is

the identity, $a_i \notin \text{Rep}(a'_j)(1 \leqq j \leqq m')$, and (since a_i is an atom) $a_i \notin \text{Rep}((\Phi'_{L_j} \odot \Phi'_{R_j}))$ $(1 \leqq j \leqq q')$. Therefore $a_i \in \text{Rep}(\Phi) - \text{Rep}(\Phi')$, and thus Φ and Φ' cannot both represent C.

If they do not differ on their atoms, then they differ on their products. WLOG, suppose $(\Phi_{L_i} \odot \Phi_{R_i})$ is a product in Φ but not Φ', i.e., for all $1 \leqq j \leqq q'(\Phi_{L_i} \odot \Phi_{R_i})$ differs from $(\Phi'_{L_j} \odot \Phi'_{R_j})$ by more than order of + operations. We will show that $\text{Rep}(\Phi') \cap \text{Rep}((\Phi_{L_i} \odot \Phi_{R_i})) = \varnothing$, and hence Φ and Φ' cannot both represent C.

For all $1 \leqq j \leqq m'$, $\text{Rep}(a'_j)$ and $\text{Rep}((\Phi_{L_i} \odot \Phi_{R_i}))$ are disjoint since $\cong$ is the identity and a'_j is an atom. Consider $(\Phi'_{L_j} \odot \Phi'_{R_j})$, for any $1 \leqq j \leqq q'$. WLOG, suppose Φ_{L_i} differs from Φ'_{L_j} by more than the order of + operations. Since Φ and Φ' are dynamic programs, so too must be Φ_{L_i} and Φ'_{L_j}. Further, they both represent classes of rank less than k (since both classes are proper descendents of C), and thus by the inductive hypothesis they cannot represent the same $\tilde{}$-class, i.e., they must represent disjoint sets of conjuncts. It then follows from the "Sub-lemma" below that $\text{Rep}((\Phi_{L_i} \odot \Phi_{R_i}))$ and $\text{Rep}((\Phi'_{L_j} \odot \Phi'_{R_j}))$ are disjoint, giving us the desired result.

SUB-LEMMA. *Let $\cong$ be the identity, and suppose that Υ_1 and Υ_2 represent disjoint sets. Then for all Υ'_1 and Υ'_2, $(\Upsilon_1 \odot \Upsilon'_1)$ and $(\Upsilon_2 \odot \Upsilon'_2)$ represent disjoint sets, and similarly so do $(\Upsilon'_1 \odot \Upsilon_1)$ and $(\Upsilon'_2 \odot \Upsilon_2)$.*

Proof. Simply observe that when $\cong$ is the identity $[(W \odot X) \cong (Y \odot Z)$ iff $W = Y$ and $X = Z]$. □

Before proving Theorem 2, we need the following simple definition and lemma.

DEFINITION. A term is an NRE which is either an atomic expression or a product.

LEMMA APX2. *Let $\cong$ be the identity and Ψ and interpretable* NRE *representing the $\tilde{}$-class C, with $\Psi = ((\Psi_1 \odot \Psi'_1) + \cdots + (\Psi_k \odot \Psi'_k) + a_1 + \cdots + a_m)$. Let $X \in \text{Rep}(\psi_i)$, $Y \in \text{Rep}(\Psi'_i)$ for some $1 \leqq i \leqq k$. Then*

$$[W \tilde{} X \text{ and } Z \tilde{} Y] \rightarrow [\text{there exists } j\ (1 \leqq j \leqq k) \text{ such that } W \in \text{Rep}(\Psi_j),\ Z \in \text{Rep}(\Psi'_j)].$$

Proof. Since $\odot$ preserves $\tilde{}$ and C is an entire class, $(W \odot Z) \in \text{Rep}(\Psi) = C$. Thus for some term T in $\{(\Psi_1 \odot \Psi'_1), \cdots, (\Psi_k \odot \Psi'_k), a_1, \cdots, a_m\}$, $(W \odot Z) \in \text{Rep}(T)$. But since $\cong$ is the identity, T must be a product $(\Psi_j \odot \Psi'_j)(1 \leqq j \leqq k)$ with $W \in \text{Rep}(\Psi_j)$ and $Z \in \text{Rep}(\Psi'_j)$. □

THEOREM 2. *If $\cong$ is the identity (i.e., distinct conjuncts are never equivalent), Ψ is an interpretable* NRE *with $\Psi \equiv Q$, and Φ is the unique dynamic program, then*

(1) *number of +'s in $\Phi \leqq$ number of +'s in Ψ.*

(2) *number of $\odot$'s in $\Phi \leqq$ number of $\odot$'s in Ψ.*

Proof. We will show the result holds if Ψ represents any class C. The proof is by induction on $c =$ number of $\odot$'s contained in Ψ.

$c = 0$. Then Ψ must be $a_1 + a_2 + \cdots + a_k$, where $a_i \in A$. To obtain the required dynamic program Φ, choose a subset of the $\{a_1, a_2, \cdots, a_k\}$ such that there is exactly one a_i from each of the $\cong$-classes $[a_1]_{\cong}, \cdots, [a_k]_{\cong}$.

Assume true for all $c < k$.

Show true for $c = k$. Let

$$\begin{aligned}\Psi = &[(\Psi_{1_1} \odot \Psi'_{1_1}) + \cdots + (\Psi_{1_{k_1}} \odot \Psi'_{1_{k_1}})] + \cdots \\ &+ [(\Psi_{i_1} \odot \Psi'_{i_1}) + \cdots + (\Psi_{i_{k_i}} \odot \Psi'_{i_{k_i}})] + \cdots \\ &+ [(\Psi_{n_1} \odot \Psi'_{n_1}) + \cdots + (\Psi_{n_{k_n}} \odot \Psi'_{n_{k_n}})] + [a_1 + \cdots + a_m],\end{aligned}$$

where each group $[(\Psi_{i_1} \odot \Psi'_{i_1}) + \cdots + (\Psi_{i_{k_i}} \odot \Psi'_{i_{k_i}})]$ contains exactly those terms in which the left members Ψ_i represent subsets of the class C_i and the right member Ψ'_i

TABLE 1
Summary of terms and symbols.

Term or symbol	Brief definition
A	finite set of "atoms"
$\odot$	nonassociative analogue of string concatenation
conjunct	expression over A and $\odot$
CONJ	set of all conjuncts
$\cong$	equality relation on CONJ. $X \cong Y$ denotes that X and Y represent the same application object
$[X]_{\cong}$	$\cong$-equivalence class containing X
NRE	nonassociative regular expression
$+$	the union operator for NREs
$\otimes$	the product operator for sets of conjuncts
Q	NRE representing the set of conjuncts which are to be enumerated
$\tilde{}$	the comparability relation on CONJ
$[X]$	$\tilde{}$-equivalence class containing X
$\tilde{}$-NRE	an NRE which represents a set of $\tilde{}$-comparable conjuncts
interpretable	an NRE whose subexpressions are all $\tilde{}$-NREs
choice	function which chooses the optimal conjunct from a set of $\tilde{}$-comparable conjuncts
Δ_{choice}	operation which chooses between two $\tilde{}$-comparable conjuncts
Comp (Ψ)	computation derived from NRE Ψ by replacing each $+$ with a Δ_{choice}
Rep	Rep (Ψ) is the set of conjuncts represented by the NRE Ψ
$\equiv$	$\Psi \equiv \Upsilon$ means Rep (Ψ) = Rep (Υ)
subexp (Ψ)	is the set of subexpressions of the NRE Ψ
product	NRE of the form ($\Psi \odot \Upsilon$)
term	NRE which is either a product or single atom

represent subsets of the class C_i'. (Obviously the terms of any interpretable NRE can always be so grouped without changing the set represented nor the number of operations. Also assume that the a_i are from m distinct $\cong$ classes; if not, remove redundancy as in the basis.)

Consider any one of these groups of terms, $[(\Psi_1 \odot \Psi_1') + \cdots + (\Psi_k \odot \Psi_k')]$. (We have dropped one of the subscripts as a notational convenience.) Let q be the largest integer such that Rep (Ψ_q) contains a conjunct not in any Rep (Ψ_r), $1 \leqq r < q$. It then follows from Lemma Apx2 that

$$\bigcup_{j=1}^{q} \text{Rep}(\Psi_j) = C_i \quad \text{and} \quad \bigcup_{j=q}^{k} \text{Rep}(\Psi_j') = C_j'.$$

(The second equality follows because there is a conjunct $X \in C_i$ which is not in any Rep (Ψ_j) for $j < q$. Lemma Apx2 implies that every $Y \in C_i'$ must appear in some Rep (Ψ_r') for which $X \in$ Rep (Ψ_r). Since X appears only when $r \geqq q$, each $Y \in C'$, must appear in some Rep (Ψ_r'), $q \leqq r \leqq k$.)

Thus the group of terms $[(\Psi_1 \odot \Psi_1') + \cdots + (\Psi_k \odot \Psi_k')]$ can be replaced by the single term $(\Psi_1 + \cdots + \Psi_q) \odot (\Psi_q' + \cdots + \Psi_k')$. This term has exactly the same number of $+$'s, $k - 1 = (q - 1) + (k - q)$, as the original group, and only one $\odot$. This replacement is completely "localized" within Ψ since Lemma Apx2 insures the set represented is not altered, and the way the terms of Ψ were originally arranged insures no term $(\Psi_j \odot \Psi_j')$ appears outside of the collection. $(\Psi_1 + \cdots + \Psi_q)$(and $(\Psi_q' + \cdots + \Psi_k')$) is an interpretable NRE because

(i) each Ψ_i is interpretable, since Ψ is interpretable, and

(ii) the Rep (Ψ_i) are all subsets of the same $\tilde{}$-class.

We replace each group of terms in Ψ in a similar manner, obtaining

$$\Psi^* = [(\Psi_{1_1} + \cdots + \Psi_{1_{q_1}}) \odot (\Psi'_{1_{q_1}} + \cdots + \Psi'_{1_{k_1}})] + \cdots$$
$$+ [(\Psi_{i_1} + \cdots + \Psi_{i_{q_i}}) \odot (\Psi'_{i_{q_i}} + \cdots + \Psi'_{i_{k_i}})] + \cdots$$
$$+ [(\Psi_{n_1} + \cdots + \Psi_{n_{q_n}}) \odot (\Psi'_{n_{q_n}} + \cdots + \Psi'_{n_{k_n}})] + [(a_1 + \cdots + a_m)].$$

For each $1 \leqq i \leqq n$, $(\Psi_{i_1} + \cdots + \Psi_{i_{q_i}})$ and $(\Psi'_{i_{q_1}} + \cdots + \Psi'_{i_{k_i}})$ are interpretable NREs, each containing fewer than k $\odot$'s, representing the classes C_i and C'_i. Thus, by the inductive hypothesis, each pair $(\Psi_{i_1} + \cdots + \Psi_{i_{qi}})$ and $(\Psi'_{i_{q_i}} + \cdots + \Psi'_{i_{k_i}})$ can be replaced by the dynamic programs Φ_i and Φ'_i (which represent C_i and C'_i) without increasing the number of + or $\odot$ operations. It is also the case that the transformation does not increase the overall cost when common subexpressions are counted only once. It is easy to see that, at each inductive step, the replacement of $(\Psi_{i_1} + \cdots + \Psi_{i_{q_i}})$ by Φ_i either leaves each subexpression of $(\Psi_{i_1} + \cdots + \Psi_{i_{q_i}})$ available for use elsewhere, or replaces it with a subexpression which can be used anywhere in its place, i.e., by a product which represents the $\tilde{}$-class containing the set represented by the original subexpression.

Therefore the dynamic program

$$\Phi = (\Phi_1 \odot \Phi'_1) + \cdots + (\Phi_k \odot \Phi'_k) + a_1 + \cdots + a_m$$

represents C with no more +'s or $\odot$'s than Ψ. □

Acknowledgments. While developing the principles later realized in this model, A. Rosenthal was partially supported by the Horace Rackham Graduate School, University of Michigan, and by National Science Foundation grant MCS77-01753.

REFERENCES

[1] R. BELLMAN, *Dynamic Programming*, Princeton Univ. Press, Princeton, NJ, 1957.
[2] U. BERTELE AND F. BRIOSCHI, *Nonserial Dynamic Programming*, Academic Press, New York, 1972.
[3] P. BONZON, *Necessary and sufficient conditions for dynamic programming of the combinatorial type*, J. Assoc. Comput. Mach., 17 (1970), pp. 675-682.
[4] E. GILBERT AND E. MOORE, *Variable length encodings*, Bell System Tech. J., 38 (1959), pp. 933-967.
[5] S. GNESI, U. MONTANARI AND A. MARTELLI, *Dynamic programming as graph searching: an algebraic approach*, J. Assoc. Comput. Mach., 28 (1981), pp. 737-751.
[6] P. HELMAN, *A new theory of dynamic programming*, Ph.D thesis, Dept. Computer and Communications Sciences, Univ. Michigan, Ann Arbor, 1982.
[7] E. HOROWITZ AND S. SAHNI, *Fundamentals of Computer Algorithms*, Computer Science Press, Potomac, MD, 1978.
[8] T. IBARAKI, *Minimal representations of some classes of dynamic programming*, Inform. Con., 27 (1975), pp. 289-328.
[9] R. M. KARP AND M. HELD, *Finite state processes and dynamic programming*, SIAM J. Appl. Math., 15 (1967), pp. 693-718.
[10] L. G. MITTEN AND G. L. NEMHAUSER, *Multistage optimization*, Chemical Engineering Progress, 59 (1963), pp. 52-60.
[11] A. ROSENTHAL, *Dynamic programming is optimal for nonserial optimization problems*, SIAM J. Comput., 11 (1982), pp. 47-59.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 2, April 1985

018

ON DISCRETE SEARCH FOR A MULTIPLE NUMBER OF OBJECTS*

MARK CIANCUTTI†

Abstract. In this paper we discuss discrete search for a number of objects distributed among a number of cells. A cell is chosen during each inspection period and objects removed. The initial number of objects in each cell has a discrete probability density independent of the number in another cell. Under a wide variety of conditions, it is shown that in order to maximize the expected discounted value of objects discovered in a fixed and infinite number of inspection periods, the myopic rule of selecting the cell at each stage which has largest expected value of objects is optimal.

We prove this in the first part of the paper when objects are binomially distributed in each cell and later in the second part show that the myopic rule is optimal for a wide variety of distributions using the same proofs presented in the first part of the paper.

AMS(MOS) subject classifications. 62L99, 68E05, 68E10

1. Introduction. The problem of searching for a single object in a number of cells has been discussed extensively in the literature. (See Chew, Black, Blackwell, Matula, Kadane, and DeGroot.) Enslow has even compiled a bibliography on the research which has some of its roots in the work by Bellman. Some of the aspects considered have been various costs for searching the cells and overlook probabilities for finding objects in attempting to find an optimal procedure to minimize the expected cost of finding the object. These aspects are included in the beginning portion of the paper with Theorem 1 and used to motivate the generalizations in the latter portions.

Theorem 1 is new in the sense that a multiple number of objects may be in the cells whereas the literature previously cited deals with the search for a single object. Another paper by Kimeldorf and Smith deals with an optimal search procedure for a random number of multinomially hidden objects, but by assuming the number of objects in a cell to be distributed independently of those in another cell the present paper develops an allocation rule under wider conditions of cost and overlook considerations. To do this, the present paper uses the techniques of dynamic programming to develop a simple rule for searching through the cells in a fixed number of periods.

Of mathematical interest is the way stochastic dynamic programming is applied to problems of this type with a fixed number of searches and the generalization to the case involving an infinite number of search periods. The results are of such a general nature that they may be construed as the theoretical justification for employing the myopic rule of sequentially selecting the most valuable cell in each search period.

2. Discrete search for a multiple number of objects. Consider the model for discrete search where there are a multiple number of objects distributed among G cells and discrete time period inspections of the cells conducted to discover and remove objects. Specifically, let X_i represent the random variable corresponding to the number of objects in cell i, $i=1,\cdots,G$. Initially, let X_i, $i=1,\cdots,G$ be independently distributed such that X_i has a binomial distribution with parameters N_i and s_i. Here the N_i and s_i are known. We also have known overlook probabilities p_i associated

* Received by the editors July 6, 1983 and in revised form February 8, 1984. This work was presented at the SIAM Second Conference on the Applications of Discrete Mathematics held at Massachusetts Institute of Technology, Cambridge, Massachusetts, June 27–29, 1983.

† Robert Morris College, Pittsburgh, Pennsylvania 15219.

with each cell such that

$$\Pr(\text{overlooking exactly } k \text{ objects in cell } i \mid b_i \text{ objects in cell } i) = \binom{b_i}{k} p_i^k (1-p_i)^{b_i-k}.$$

Given the above condition, the number of objects remaining in cell i after m searches of cell i is distributed as a binomial with parameters $N_i - d_m$ and $s_i^{(m)}$, where d_m is the number of objects found in the m searches of cell i and $s_i^{(m)} = T_i^{(m)}(s_i)$. Here $T_i^{(m)}$ is the mth-fold composition of the operator T_i, where $T_i(x) = p_i x/1 + p_i x - x$.

To prove this last statement, let B be the random variable denoting the number of objects left behind after the first search and let A be the random variable denoting the number found in the first search. We have

$$\Pr(B = k \mid A = j) = c\binom{N_i}{k+j} s_i^{k+j}(1-s_i)^{N_i-(k+j)}\binom{k+j}{k} p_i^k (1-p_i)^j$$

where $c = 1/\Pr(A = j)$, which does not involve k. Now

$$\Pr(B = k \mid A = j) = c'\binom{N_i}{k+j}\binom{k+j}{k}\left(\frac{s_i p_i}{1+s_i p_i - s_i}\right)^k \left(\frac{1-s_i}{1+s_i p_i - s_i}\right)^{(N_i-j)-k},$$

where c' is a constant not involving k. This latter expression is proportional to the probability that a binomial random variable with parameters $N_i - j$ and $s_i p_i/1 + s_i p_i - s_i$ is equal to k. Therefore, distribution of B given $A = j$ is binomial with these parameters. By using the above argument inductively on the number of searches of cell i, it is easy to see that if we continue to search cell i, there is a binomial distribution for the remaining objects whose parameters are $N_i - d_m$ and $T_i^{(m)}(s_i)$.

In addition to the above assumptions for the model, there is a cost c_i associated with each search of cell i and a value V_i for each object found in cell i. We are going to search through the cells in n stages in order to maximize the expected discounted net value, $E(\sum_{i=1}^n \beta_i v_i)$. Here v_i is a random variable defined to be $v_i = fV_l - C_l$ if we find f objects in cell l which we have chosen in the ith search. The β_i are discount factors such that $\beta_i > \beta_{i+1}$, $0 < \beta_i < \infty$.

Now all search strategies through the cells in n stages may be viewed as sequential decision procedures. If the system is in state $\mathbf{X}_i$ (a vector of binomial random variables as discussed above) at the ith stage, a procedure designates which cell is to be searched at the ith stage as a function of the system at the ith stage. The state of the system is completely described by the vector of random variables which represent the distribution for the undiscovered objects in the system.

THEOREM 1. *The optimal policy for the above model selects the cell with the highest expected net value of objects available at each stage. That is, the optimal rule is the myopic rule.*

Proof. We are going to use backward induction, from stage n, describing the optimal strategy and eliminating all search procedures which do not fit the optimal policy. Any procedure which at the nth stage selects a cell which does not have the highest expected net value cannot be optimal since we can replace this procedure with one which selects, at stage n, a cell which has highest expected net value and get a policy which has a higher $E(\sum_{i=1}^n \beta_i v_i)$.

Now we are ready for the induction step. It is assumed that at the $(i+1)$st stage only procedures which, given state vector $\mathbf{X}_{(i+1)}$ select a cell which has highest expected net value can be considered optimal. We intend to show that a procedure from this

collection which at the ith stage with state vector $\mathbf{X}_i$ selects a cell which does not have highest expected net value cannot be optimal.

Suppose we have just such a procedure and that at stage i with state vector $\mathbf{X}_i$ it selects a cell that does not belong to the subset B of cells b which attain the highest expected net value at this stage. By the induction hypothesis, the procedure must select a cell from B at the $(i+1)$st stage—no matter what we find in a—since these same cells will again have highest expected net value for the stage vector $\mathbf{X}_{i+1}$. Suppose the procedure has followed the strategy $\boldsymbol{\sigma}$ which has led to the state vector $\mathbf{X}_i$ and that after a cell in B is selected at the $(i+1)$st stage the procedure continues with policy $\mathscr{P}$ on the remaining state vectors until the nth stage. We now examine an expression for the expected net worth of the procedure $\{\boldsymbol{\sigma}, a, b, \mathscr{P}\}$ which is defined to be the strategy which follows $\boldsymbol{\sigma}$, then selects cell a at stage i, then selects cell b at stage $i+1$, and then continues with policy $\mathscr{P}$ for the remaining stages, where $\boldsymbol{\sigma}$, a, b, and $\mathscr{P}$ are as given above.

To do this, we will employ the following notation. Let X_r be the random variable designating the number of undiscovered objects in cell r after having followed $\boldsymbol{\sigma}$ to the ith stage for $r = a, b, \cdots, G$. We have $X_a \sim \text{Bin}\,(N_a^{\boldsymbol{\sigma}}, s_a^{\boldsymbol{\sigma}})$ (i.e. X_a is a random variable distributed binomially with parameters $N_a^{\boldsymbol{\sigma}}$ and $s_a^{\boldsymbol{\sigma}}$) where $N_a^{\boldsymbol{\sigma}}$ equals N_a minus the number of objects found in cell a during $\boldsymbol{\sigma}$ and $s_a^{\boldsymbol{\sigma}} = s_a^{(k)}$ where k is the number of times cell a was investigated during $\boldsymbol{\sigma}$. Let a_f be the probability of finding f objects in cell a at stage i, $X_a(f) \sim \text{Bin}\,(N_a^{\boldsymbol{\sigma}} - f, s_a^{(k+1)})$. Let $\mathscr{W}_{\boldsymbol{\sigma}}$ denote the expected net worth already gotten from $\boldsymbol{\sigma}$. Then, if $\mathscr{W}_{\{\boldsymbol{\sigma},a,b,\mathscr{P}\}}(\mathbf{X}_i)$ is the expected net worth of the procedure under investigation and $\mathscr{W}'_{\{b,\mathscr{P}\}}$ the expected net worth of its argument under the procedure $\{b, \mathscr{P}\}$ we have:

$$\begin{aligned}\mathscr{W}_{\{\boldsymbol{\sigma},a,b,\mathscr{P}\}}(\mathbf{X}_i) &= \mathscr{W}_{\{\boldsymbol{\sigma},a,b,\mathscr{P}\}}(X_a, X_b, \cdots, X_G) \\ &= \sum_{f=0}^{N_a^{\boldsymbol{\sigma}}} a_f\,(\beta_i(fV_a - \mathscr{C}_a) + \mathscr{W}'_{\{b,\mathscr{P}\}}(X_a(f), X_b, \cdots, X_G)) + \mathscr{W}_{\{\boldsymbol{\sigma}\}}.\end{aligned}$$

Now let b_g denote the probability of finding g objects in cell b at stage $i+1$, $X_b(g) \sim \text{Bin}\,(N_b^{\boldsymbol{\sigma}} - g, s_b^{(k'+1)})$, where k' is the number of times cell b is visited during $\boldsymbol{\sigma}$, and let $\mathscr{W}''_{\mathscr{P}}$ be the expected net worth of its argument under the procedure $\mathscr{P}$. We may expand the above expression and get:

$$\begin{aligned}\mathscr{W}_{\{\boldsymbol{\sigma},a,b,\mathscr{P}\}}(\mathbf{X}_i) &= \sum_{f=0}^{N_a^{\boldsymbol{\sigma}}} a_f(\beta_i(fV_a - \mathscr{C}_a)) + \sum_{g=0}^{N_\beta^{\boldsymbol{\sigma}}} b_g(\beta_{i+1}(gV_b - \mathscr{C}_b)) \\ &\quad + \sum_{f=0}^{N_a^{\boldsymbol{\sigma}}} \sum_{g=0}^{N_b^{\boldsymbol{\sigma}}} a_f b_g \mathscr{W}''_{\{\mathscr{P}\}}(X_a(f), X_b(g), \cdots, X_G) + \mathscr{W}_{\{\boldsymbol{\sigma}\}}.\end{aligned}$$

Now we are going to construct another procedure, not necessarily optimal itself, but one with higher expected net worth than $\{\boldsymbol{\sigma}, a, b, \mathscr{P}\}$. Suppose that $\mathscr{P}$ applies a certain policy to the state vector $(X_a(f), X_b(g), \cdots, X_G)$ at the $(i+2)$nd step. We construct a new procedure $\mathscr{P}'$ which applies that same policy to the same state vector which has now resulted from the procedure $\{\boldsymbol{\sigma}, b, a\}$. We now investigate the expected net worth of $\{\boldsymbol{\sigma}, b, a, \mathscr{P}'\}$.

We have:

$$\begin{aligned}\mathscr{W}_{\{\boldsymbol{\sigma},b,a,\mathscr{P}'\}}(X_a, X_b, \cdots, X_G) &= \sum_{g=0}^{N_b^{\boldsymbol{\sigma}}} b_g(\beta_i(gV_b - \mathscr{C}_b)) + \sum_{f=0}^{N_a^{\boldsymbol{\sigma}}} a_f(\beta_{i+1}(fV_a - \mathscr{C}_a)) \\ &\quad + \sum_{g=0}^{N_b^{\boldsymbol{\sigma}}} \sum_{f=0}^{N_a^{\boldsymbol{\sigma}}} b_g a_f \mathscr{W}''_{\{\mathscr{P}'\}}(X_a(f), X_b(g), \cdots, X_G) + \mathscr{W}_{\{\boldsymbol{\sigma}\}}.\end{aligned}$$

If we take $\mathcal{W}_{\{\sigma,b,a,\mathcal{P}'\}} - \mathcal{W}_{\{\sigma,a,b,\mathcal{P}\}}$, the $\mathcal{W}$-terms cancel out because $\mathcal{W}'_{\{\mathcal{P}'\}}(X_a(f), X_b(g), \cdots, X_G) = \mathcal{W}'_{\{\mathcal{P}\}}(X_a(f), X_b(g), \cdots, G)$. Hence $\mathcal{W}_{\{\sigma,b,a,\mathcal{P}'\}} - \mathcal{W}_{\{\sigma,a,b,\mathcal{P}\}} > 0$ since $N_b^{\sigma}(1-p_b)s_b^{\sigma}V_b - \mathcal{C}_b > N_a^{\sigma}(1-p_a)s_a^{\sigma}V_a - \mathcal{C}_a$ and therefore $\sigma, a, b, \mathcal{P}$ is not optimal. By the induction hypothesis, then, a procedure which at some stage does not select the cell which has highest expected net value is not optimal. Since the only procedure left for consideration is the procedure σ_m which selects at each stage the cell with highest expected net value, it must be optimal. □

We now let $n \to \infty$.

THEOREM 2. *Under the conditions of Theorem* 1, *if* $\sum_{i=1}^{\infty} \beta_i < \infty$ *then the procedure* σ'_m *which selects a cell with largest expected net value maximizes* $E(\sum_{i=1}^{\infty} \beta_i v_i)$.

Proof. Again the proof is by contradiction. Since the expectations are finite, suppose that there exists a procedure σ' not the same as σ'_m such that

$$E_{\sigma'}\left(\sum_{i=1}^{\infty} \beta_i v_i\right) = E_{\sigma'_m}\left(\sum_{i=1}^{\infty} \beta_i v_i\right) + \varepsilon$$

for some $\varepsilon > 0$. Now since $\sum_{i=1}^{\infty} \beta_i$ is convergent and since the v_i have bounded expectation, we know there exists an n such that for all $\sigma \in S'$ where S' is the set of all infinite stage sequential decision procedures,

$$\left|E_{\sigma}\left(\sum_{i=1}^{\infty} \beta_i v_i\right) - E_{\sigma}\left(\sum_{i=1}^{n} \beta_i v_i\right)\right| < \frac{\varepsilon}{4},$$

for any $\varepsilon > 0$. By the choice of n,

$$\left|E_{\sigma'}\left(\sum_{i=1}^{\infty} \beta_i v_i\right) - E_{\sigma'}\left(\sum_{i=1}^{n} \beta_i v_i\right)\right| < \frac{\varepsilon}{4}$$

and

$$\left|E_{\sigma'_m}\left(\sum_{i=1}^{\infty} \beta_i v_i\right) - E_{\sigma'}\left(\sum_{i=1}^{n} \beta_i v_i\right)\right| < \frac{\varepsilon}{4}.$$

Hence, $E_{\sigma'}(\sum_{i=1}^{n} \beta_i N_i) > E_{\sigma'_m}(\sum_{i=1}^{n} \beta_i v_i)$ which is a contradiction to the optimality of choosing the cell with highest net value for the n stage system. Hence σ'_m optimizes $E(\sum_{i=1}^{\infty} \beta_i v_i)$ in S'. □

3. A subcase and generalization of the model. As a special case of this model, consider the instance where $N_i = 1$ for $i = 1, \cdots, G$. That is, there is at most one object per cell. In order to associate these results with the Fermi–Dirac distribution described in Theorem 1, the joint distribution for the number of objects in each of the cells must be equal to the product of the marginal distributions for the number of objects in each cell. Under these conditions, the only possible distribution for the total number of objects in the model of Theorem 1 is Bin (G, p_0) for some p_0.

To see this first note that if the total number of objects in the system has a binomial distribution, Bin (G, p_0), it can be represented as the sum of G independent Bernoulli random variables with parameter p_0, one for each cell. Thus we satisfy the condition that the joint distribution for the number of objects in each of the cells must be equal to the product of the marginal distributions for the number of objects in each cell. Furthermore, for any fixed number of objects, any arrangement among the cells will be equally likely, thus satisfying the Fermi–Dirac distribution criterion.

Conversely, suppose that the conditions of marginal independence for the number of objects in each cell are satisfied with at most one object per cell. Then the distribution

for the total number of objects in the cells must be the sum of G independent Bernoulli random variables with cell i having parameter p_i. However, the configurations $(1,0,0,\cdots,0), (0,1,0,\cdots,0),\cdots,(0,0,\cdots,1)$ must have the same probability by the Fermi–Dirac distribution. That is, $p_1\prod_{j\neq 1}(1-p_j)=p_2\prod_{j\neq 2}(1-p_j)=p_3\prod_{j\neq 3}(1-p_j)=\cdots=p_k\prod_{j\neq k}(1-p_j)$. Hence the parameters p_i are equal and the total number of objects in the system has a binomial distribution.

Noting the vital points in the proof of Theorems 1 and 2, we can make appropriate generalizations of the model for these theorems and still arrive at the same conclusions. Let $X_1,\cdots,X_G$ be independent, discrete random variables. Let X_i represent the number of objects in cell i. Let $X_i^{(k)}$ be a random variable representing the number of objects remaining in cell i after the kth search; $i=1,\cdots,G$; $k=0,1,\cdots$. We use the notation $X_i^{(0)}=X_i$. Also we have functions f_i; $i=1,\cdots,G$. Here $f_i(X_i^{(k)})$ is the random variable representing the number of objects found in the $(k+1)$st search of cell i. We also have functions V_i, $i=1,\cdots,G$ representing the value of objects found in a search of cell i. We require $-\infty<E(V_i(f_i(X_i^{(k+1)})))\leqq E(V_i(f_i(X_i^{(k)})))<\infty$ for $i=1,\cdots,G$ and $k=0,1,2,\cdots$. In this model we wish to maximize $E(\sum_{i=1}^{n}\beta_i v_i)$ in a search through the cells in n stages where the β_i are discount factors as in Theorem 1 and v_i is a random variable denoting the value of objects found in the cell chosen for the ith search. Under these conditions, the optimal policy is to choose at each stage the cell with highest expected value.

The proof of this statement follows exactly the lines of Theorem 1 except that now we have appropriately generalized the initial conditions and valuation functions. Following the proof of Theorem 1 step by step, let $\tilde{X}_a(f)$ be the random variable for the number of objects in cell a after finding f at stage i, $\tilde{X}_b(g)$ be the random variable for the number of objects in cell b after finding g at stage $i+1$, and $\tilde{X}_j$ be the random variable for the number of objects in cell $j\neq a$ or b at stage i. We have:

$$\mathcal{W}_{\{\sigma,a,b,\mathcal{P}\}}=\sum_f a_f\beta_i[V_a(f)]+\sum_g b_g\beta_{i+1}[V_b(g)]$$
$$+\sum_f\sum_g a_f b_g\mathcal{W}''_{\{\mathcal{P}\}}(\tilde{X}_a(f),\tilde{X}_b(g),\cdots,\tilde{X}_G)+\mathcal{W}_{\{\sigma\}}.$$

Similarly, we have

$$\mathcal{W}_{\{\sigma,b,a,\mathcal{P}'\}}=\sum_g b_g\beta_i[V_b(g)]+\sum_f a_f\beta_{i+1}[V_a(f)]$$
$$+\sum_g\sum_f b_g a_f\mathcal{W}''_{\{\mathcal{P}'\}}(\tilde{X}_b(g),\tilde{X}_a(f),\cdots,\tilde{X}_G)+\mathcal{W}_{\{\sigma\}}.$$

Thus $\mathcal{W}_{\{\sigma,b,a,\mathcal{P}'\}}-\mathcal{W}_{\{\sigma,a,b,\mathcal{P}\}}>0$ since $E(V_b(f_b(\tilde{X}_b)))>E(V_a(f_a(\tilde{X}_a)))$. This last inequality turns out to be the optimal rule. Hence the myopic rule is optimal for this generalization.

Furthermore, if we assume that there exists a number M such that for all i and k $|E(V_i(f_i(X_i^{(k)})))|\leqq M<\infty$; and if $\sum_{i=1}^{\infty}\beta_i<\infty$, then the myopic rule holds in the limit as the number of steps n approaches infinity. The proof of this follows exactly the lines of Theorem 2. Hence with the restriction of bounded expected values, we have an analogue of Theorem 2 for this generalization of Theorem 1.

Acknowledgments. This paper was taken from part of the author's Ph.D. thesis. He would like to express his gratitude to Professor Morris DeGroot (thesis advisor) and Professor Joseph Kadane, both of the Department of Statistics of Carnegie-Mellon University, for their help and suggestions in the completion of this effort.

REFERENCES

R. BELLMAN (1957), *Dynamic Programming*, Princeton Univ. Press, Princeton, NJ.

W. L. BLACK (1965), *Discrete sequential search*, Inform. Control, 8, pp. 156–162.

D. BLACKWELL (1962), *Notes on dynamic programming*, unpublished notes, Univ. California, Berkeley, 1962.

M. C. CHEW JR., *A sequential search procedure*, Ann. Math. Stat., 38 (1967), pp. 494–502.

M. H. DEGROOT (1970), *Optimal Statistical Decisions*, McGraw-Hill, New York.

P. ENSLOW, JR. (1966), *A bibliography of search theory and reconnaissance theory literature*, Naval Res. Logist. Quarterly, 13, pp. 177–202.

J. B. KADANE (1968), *Discrete search and the Neyman–Pearson lemma*, J. Math. Anal. Appl., 22, pp. 156–171.

G. KIMELDORF AND F. H. SMITH (1979), *Discrete search for a random number of multinomially hidden objects*, Management Sci., 25, pp. 1115–1126.

D. MATULA (1964), *A periodic optimal search*, Amer. Math. Monthly, 71, pp. 15–21.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 2, April 1985

019

SOLUTION OF THE DISCRETE LYAPUNOV EQUATION*

SHEAU-WEI FU† AND MAHMOUD E. SAWAN‡

Abstract. The lower bounds for the geometric means of the eigenvalues of the positive definite solution to the discrete Lyapunov equation are investigated. The results obtained by different methods are compared in search of an optimal value.

1. Introduction. The numerical solutions to the discrete Lyapunov matrix equation are frequently needed in the control system design and analysis. The computation involves the proper selection of an initial guess of the solution matrix to assure the convergence of the solution. For large scale real-time system application, it is particularly important to know a proper initial guess of the solution matrix that provides fast convergence. Thus an estimate of the size of the solution matrix will be useful. In this paper we try to establish the optimal lower bounds of the geometric means of the eigenvalues of the positive definite solution to the discrete Lyapunov matrix equation. The proof of Theorem 1 was presented by Tran and Sawan [4] and the proofs of Theorem 2 and Theorem 3 were presented by Mori and Fukuma [5].

In the following, the notations x^T, $\lambda_i(X)$, tr (X), $|x|$ and $\rho(X)$ denote the transpose, eigenvalue, trace, determinant and spectral radius of the matrix x, respectively. The discrete Lyapunov equation is given as

$$P = A^T P A + Q, \qquad Q = Q^T > 0, \tag{1}$$

where A, P, $Q \in R^{n\times n}$ and $>$ denotes positive definiteness.

Assuming $\rho(A) < 1$, i.e. the matrix A is stable, the solution $P = P^T > 0$ uniquely exists. The geometric mean of the eigenvalues of P is defined as

$$m_g(P) \triangleq \left(\prod_{i=1}^{n} \lambda_i(P) \right)^{1/n} = |P|^{1/n}.$$

For our later derivation, we will make use of the following results [1, p. 70], [2, p. 225], [3].

(i) For matrices L and H, $L > 0$, L, $H \in R^{n\times n}$

$$\text{tr}\,(L^{-1}HLH^T) \geqq \sum_{i=1}^{n} |\lambda_i(H)|^2 \geqq \frac{1}{n}[(\text{tr}\,(H))]^2. \tag{2}$$

(ii) For matrices X and Y, $X = X^T > 0$ and $Y = Y^T > 0$, X, $Y \in R^{n\times n}$

$$|X|^{1/n} = \min_{|Y|=1} \frac{\text{tr}\,(XY)}{n}, \tag{3}$$

$$|X+Y|^{1/n} \geqq |X|^{1/n} + |Y|^{1/n} \quad \text{(Minkowski inequality for determinants)}, \tag{4}$$

$$|X+Y| \geqq |X| + |Y|. \tag{5}$$

2. Inequalities satisfied by matrix *P*.

THEOREM 1. *The solution matrix P to* (1) *satisfies the following inequality*:

$$m_g(P) \geqq \frac{n m_g(Q)}{n - \sum_{i=1}^{n} |\lambda_i(A)|^2} \quad \text{if } \rho(A) < 1. \tag{6}$$

* Received by the editors December 12, 1983.
† Kansas Gas & Electric Company, P.O. Box 208, Wichita, Kansas 67201.
‡ Electrical Engineering Department, Wichita State University, Wichita, Kansas 67208.

Proof. The proof of this theorem follows (2) and (3) by letting $L=P$, $H=A^T$, $X=P^{-1}$ and $Y=Q/|Q|^{1/n}$ [4].

Premultiplying (1) by P^{-1} and computing the traces of both sides yields

$$\text{tr}\,(P^{-1}Q)=n-\text{tr}\,(P^{-1}A^TPA), \tag{7}$$

$$\text{tr}\,(P^{-1}A^TPA)\geqq\sum_{i=1}^{n}|\lambda_i(A^T)|^2=\sum_{i=1}^{n}|\lambda_i(A)|^2. \tag{8}$$

From (7) and (8)

$$\text{tr}\,(P^{-1}Q)\leqq n-\sum_{i=1}^{n}|\lambda_i(A)|^2. \tag{9}$$

From (3)

$$|P^{-1}|^{1/n}\leqq\frac{\text{tr}\,(P^{-1}Q)}{n|Q|^{1/n}}\leqq\frac{n-\sum_{i=1}^{n}|\lambda_i(A)|^2}{n|Q|^{1/n}}. \tag{10}$$

Hence

$$m_g(P)=|P|^{1/n}\geqq\frac{nm_g(Q)}{n-\sum_{i=1}^{n}|\lambda_i(A)|^2}. \qquad \text{Q.E.D.}$$

THEOREM 2. *The solution matrix to* (1) *satisfies the following inequality*:

$$m_g(P)\geqq\frac{m_g(Q)}{1-m_g^2(A)}\quad \text{if } \rho(A)<1. \tag{11}$$

Proof. The proof of this theorem follows the inequality (4) by letting $X=A^TPA$, $Y=Q$ [5].

$$|P|^{1/n}=|A^TPA+Q|^{1/n}\geqq|P|^{1/n}|A|^{2/n}+|Q|^{1/n}. \tag{12}$$

Since $|A|^{2/n}<1$ because of the assumption $\rho(A)<1$,

$$m_g(P)=|P|^{1/n}\geqq\frac{m_g(Q)}{1-m_g^2(A)}. \qquad \text{Q.E.D.}$$

THEOREM 3. *The solution matrix P to* (1) *satisfies the following inequality*:

$$m_g(P)\geqq\frac{m_g(Q)}{(1-m_g^{2n}(A))^{1/n}}\quad \text{if } \rho(A)<1. \tag{13}$$

Proof. The proof of this theorem follows the inequality (5) by letting $X=A^TPA$, $Y=Q$ [5].

$$|P|=|A^TPA+Q|\geqq|P||A|^2+|Q|. \tag{14}$$

Since $|A|^2<1$ because of the assumption $\rho(A)<1$,

$$|P|\geqq\frac{|Q|}{1-|A|^2}$$

which yields (13). Q.E.D.

3. Comparison of results. Our purpose is to determine the optimal (maximum) lower bounds of the geometric mean of the eigenvalues of matrix P. A comparison of the lower bounds in (6), (11) and (13) is given below.

For n positive numbers, the arithmetic mean is always greater than or equal to the geometric mean; thus

$$\frac{1}{n}\sum_{i=1}^{n}|\lambda_i(A)|^2 \geqq \left(\prod_{i=1}^{n}|\lambda_i(A)|^2\right)^{1/n} = \left(\prod_{i=1}^{n}|\lambda_i(A)|\right)^{2/n}. \tag{15}$$

Since $\rho(A)<1$,

$$\frac{1}{n}\sum_{i=1}^{n}|\lambda_i(A)|^2 < 1 \tag{16}$$

and

$$\prod_{i=1}^{n}|\lambda_i(A)| < 1. \tag{17}$$

After simple manipulation (15), (16) and (17) yield

$$\frac{1}{1-1/n\sum_{i=1}^{n}|\lambda_i(A)|^2} \geqq \frac{1}{1-(\prod_{i=1}^{n}\lambda_i(A))^{2/n}}, \tag{18}$$

$$\frac{nm_g(Q)}{n-\sum_{i=1}^{n}|\lambda_i(A)|^2} \geqq \frac{m_g(Q)}{1-(\prod_{i=1}^{n}\lambda_i(A))^{2/n}}, \tag{19}$$

which shows that the lower bound given in (6) is superior to that given in (11). From (17) it can be shown that

$$\left(\prod_{i=1}^{n}|\lambda_i(A)|\right)^{2/n} \geqq \left(\prod_{i=1}^{n}|\lambda_i(A)|\right)^{2} \tag{20}$$

and

$$1-\left(\prod_{i=1}^{n}\lambda_i(A)\right)^{2/n} \leqq 1-\prod_{i=1}^{n}\lambda_i^2(A) \leqq \left(1-\prod_{i=1}^{n}\lambda_i(A)\right)^{1/n}. \tag{21}$$

From (19) and (21)

$$\frac{nm_g(Q)}{n-\sum_{i=1}^{n}|\lambda_i(A)|^2} \geqq \frac{m_g(Q)}{1-[\prod_{i=1}^{n}\lambda_i(A)]^{2/n}} \geqq \frac{m_g(Q)}{(1-\prod_{i=1}^{n}\lambda_i^2(A))^{1/n}}. \tag{22}$$

Hence

$$\frac{nm_g(Q)}{n-\sum_{i=1}^{n}|\lambda_i(A)|^2} \geqq \frac{m_g(Q)}{(1-m_g^{2n}(A))^{1/n}},$$

which shows that the lower bound given in (6) is superior to that given in (13).

4. Conclusion. The inequality (6) provides a good estimate of the size of the solution matrix P to the discrete Lyapunov equation. We have proved that the lower bound given in (6) is superior to the lower bounds given in (11) and (13). However, the computation of the eigenvalues of matrix A generally requires more effort than the computation of the determinant of matrix A. For practical applications, if the eigenvalues of matrix A can be computed without much difficulty or if matrix A will be used repeatedly in (1) to solve for different matrix P, inequality (6) can be used to estimate an initial guess of the solution matrix to yield fast convergence. Otherwise, inequality (11), which yields lower bounds that are superior to inequality (13), can still be used to estimate the initial guess of the solution matrix. The additional computational time due to inferior initial guess may be offset by the ease of computing the determinant.

REFERENCES

[1] E. F. BECHENBACH AND R. BELLMAN, *Inequalities*, Springer-Verlag, Berlin, 1965.
[2] R. V. PATEL AND M. TODA, *Modeling error analysis of stationary linear discrete-time filters*, NASA Ames Research Center, Moffett Field, CA, TM X-73, Feb. 1977.
[3] M. MARCUS AND H. MINC, *A Survey of Matrix Theory and Inequalities*, Allyn and Bacon, Boston, 1964.
[4] M. T. TRAN AND M. E. SAWAN, *On the discrete Lyapunov and Riccati matrix equations*, Pi Mu Epsilon J., 8 (1983), pp. 574–581.
[5] T. MORI, N. FUKUMA AND M. KUWAHARA, *On the discrete Lyapunov matrix equation*, IEEE Trans. Automat. Contr., AC-27 (1982), pp. 463–464.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 3, July 1985

COVERING MULTIGRAPHS BY SIMPLE CIRCUITS*

N. ALON†‡ AND M. TARSI†

Abstract. Answering a question raised in [SIAM J. Comput., 10 (1981), pp. 746–750], we show that every bridgeless multigraph with v vertices and e edges can be covered by simple circuits whose total length is at most $\min(\frac{5}{3}e, e+\frac{7}{3}v-\frac{7}{3})$. Our proof supplies an efficient algorithm for finding such a cover.

Key words. bridgeless multigraphs, Eulerian subgraphs, graph algorithms, nowhere-zero flow

AMS (MOS) subject classification. 05

1. The main results. Let $G=(V, E)$ be an undirected bridgeless multigraph (i.e., a multigraph with no isthmus) and put $v=|V|$, $e=|E|$. A family $C_1, \cdots, C_m$ of simple circuits (=cycles) in G is a *cover* of G if every edge of G is in at least one of the circuits (2-cycles are allowed if they contain different edges of G). The *size* of such a cover is the sum of the lengths of the circuits $C_1, \cdots, C_m$. We are interested in the problem of finding covers of minimum size.

Itai, Lipton, Papadimitriou and Rodeh considered this problem in [ILPR]. Their main result is that every bridgeless multigraph G with $v \geqq 2$ vertices and $e \geqq 4$ edges has a cover of size at most

$$\min(3e-6, e+6v-7),$$

and that such a cover can be found in $O(e+v^2)$ time. (Note that since G is a multigraph, it is possible that $e \gg v^2$.) This improves a result of Itai and Rodeh in [IR].

The authors of [ILPR] ask if the multiplicative constants in their bound can be improved. In § 5 we settle this question in the affirmative by proving the following.

THEOREM 5.1. *Every bridgeless multigraph G with v vertices and e edges has a cover of size at most*

$$\min(\tfrac{5}{3}e, e+\tfrac{7}{3}v-\tfrac{7}{3}).$$

Such a cover can be found in polynomial time.

For planar multigraphs we have a better (and, in a sense, best possible) result:

THEOREM 4.2. *Every bridgeless planar multigraph with v vertices and e edges has a cover of size at most*

$$\min(\tfrac{4}{3}e, e+\tfrac{5}{3}v-\tfrac{5}{3}).$$

For a bridgeless multigraph G, let $s(G)$ denote the minimum size of a cover of G. One can easily show that if G is cubic then $s(G) \geqq \frac{4}{3}e$. Therefore, Theorem 4.2 gives the best possible upper bound for every cubic planar multigraph. In fact, $s(G)=\frac{4}{3}e$ for every cubic planar multigraph G.

One can also show (see [ILPR]) that if P is the Petersen graph (with 15 edges), then $s(G)=21$. This implies that if G is a graph obtained by substituting a path of length k for every edge of P, then $s(G)/e(G)=7/5$, where $e(G)=15k$ is the number of edges of G. Therefore, the coefficient $\frac{5}{3}$ in Theorem 5.1 cannot be replaced by any constant smaller than $\frac{7}{5}$.

* Received by the editors April 5, 1983, and in final form January 16, 1984.

† School of Mathematical Sciences, Tel Aviv University, Ramat Aviv, Israel.

‡ Present address: Department of Mathematics, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139.

In order to prove our results we use some known results about nowhere-zero flows in multigraphs. In the next section we state these results. In § 3 we develop a general method of constructing covers of small size from covers by Eulerian subgraphs. In § 4 we combine this method with the fact that every bridgeless multigraph has a nowhere-zero 8-flow and obtain a slightly weaker version of Theorem 5.1. We also prove Theorem 4.2 in this section. In § 5 we finally use nowhere-zero 6-flow to prove Theorem 5.1.

During the completion of this manuscript we were notified that our main result (Theorem 5.1) was recently proved independently by Bermond, Jackson and Jaeger [BJJ], with a different method.

2. Nowhere-zero flows. If $G=(V, E)$ is a directed multigraph and $v \in V$, then $A^+(v)$ is the set of nonloop edges with tail v and $A^-(v)$ the set with head v. If K is any Abelian group (with additive notation), a *K-flow* in G is a function $f: E \to K$ such that for every $v \in V$,

$$\sum \{f(e): e \in A^+(v)\} = \sum \{f(e): e \in A^-(v)\}.$$

If $f(e) \neq 0$ for all $e \in E$, f is called a *nowhere-zero K-flow*. For $k > 1$, f is called a *nowhere-zero k-flow* in G if f is a nowhere-zero Z-flow in G such that $-k < f(e) < k$ for all $e \in E$. (Here Z denotes the set of all integers.)

It is easy to see that if G has a nowhere-zero k-flow (K-flow) under some orientation of its edges, then it has one under every orientation, and thus the existence of such a flow depends only on the underlying undirected multigraph.

Tutte [Tu] conjectured that every bridgeless multigraph has a nowhere-zero 5-flow. Jaeger [J1], [J2] proved:

PROPOSITION 2.1 (Jaeger). *Every bridgeless multigraph has a nowhere-zero 8-flow.*

Seymour [Se] improved this result by showing:

PROPOSITION 2.2 (Seymour). *Every bridgeless multigraph has a nowhere-zero 6-flow.*

An *Eulerian multigraph* is a multigraph (not necessarily connected) in which every vertex has an even degree. Equivalently, as is well known, an Eulerian multigraph is an edge disjoint union of cycles. Thus the problem of covering a multigraph by a family of cycles of minimum total size is equivalent to that of covering the multigraph by a family of Eulerian subgraphs of minimum total size. The existence of nowhere-zero flows in a multigraph is closely related to the minimum number of Eulerian subgraphs that cover it. This is shown in the following known results.

PROPOSITION 2.3 (Jaeger [J2]). *Let G be a bridgeless multigraph. The following conditions are equivalent for every $k \geq 2$:*

(i) *There exists a nowhere-zero Z_k-flow in G.*

(ii) *For every Abelien group K of order k there exists a nowhere-zero K-flow in G.*

(iii) *There exists a nowhere-zero k-flow in G.*

PROPOSITION 2.4 (Mathews [Ma]). *Let G be a bridgeless multigraph. For every $k \geq 1$, G can be covered by k Eulerian subgraphs iff it has a nowhere-zero Z_2^k-flow.*

In §§ 4, 5, we combine Propositions 2.1–2.4 in order to obtain for every bridgeless multigraph G a cover by Eulerian subgraphs. From this cover we obtain a cover of small size of G using the method we develop in § 3.

Our results showing the connection between nowhere-zero flows and short cycle covers are summarized in Table 1.

3. Generating covers of small size from covers by Eulerian subgraphs. Our main result in this section is the following:

PROPOSITION 3.1. *Let $G=(V, E)$ be a bridgeless multigraph, and let $C=(C_1, C_2, \cdots, C_k)$ be a given cover of G by k Eulerian subgraphs. Then there exists a*

TABLE 1

If $G=(V, E)$ has a nowhere-zero k-flow for $k=$	then G has a cycle cover of length at most:
2	$\|E\|$
4	$\frac{4}{3}\|E\|$
6	$\frac{5}{3}\|E\|$
8	$\frac{12}{7}\|E\|$

cover of G of size at most

$$s=\frac{k\cdot 2^{k-1}\cdot |E|}{2^k-1}. \tag{3.1}$$

Such a cover can be found in $O(2^{k^2}|E|)$ time.

Proof. Identify each C_i with the corresponding element of the cycle space of G, i.e., with the characteristic function of C_i, regarded as a function from E to $GF(2)$. For every binary vector $u=(u_1, u_2, \cdots, u_k)$ define $C(u)=\bigoplus_{i=1}^{k} u_iC_i$. (Here $\bigoplus$ denotes the sum over $GF(2)$.) Obviously $C(u)$ is an Eulerian subgraph of G. For every edge $f\in E$, let $v(f)=(v_1, \cdots, v_k)$ be a binary vector in which $v_i=1$ iff $f\in C_i$. One can easily check that for every vector $u=(u_1, \cdots u_k)$, $f\in C(u)$ iff $\langle v(f), u\rangle=\bigoplus_{i=1}^{k} v_iu_i=1$. This implies the following:

Fact 1. For every edge $f\in E$, the number of vectors u such that $f\in C(u)$ is precisely 2^{k-1}.

Let $u^{(1)}, u^{(2)}, \cdots, u^{(k)}$ be a basis of $(GF(2))^k$. If $f\in E$ then $v(f)\neq 0$, and thus $\langle v(f), u^{(i)}\rangle\neq 0$ (i.e., $f\in C(u^{(i)})$), for at least one index $1\leqq i\leqq k$. Therefore:

Fact 2. For every basis $u^{(1)}, u^{(2)}, \cdots, u^{(k)}$ of $(GF(2))^k$, $C(u^{(1)}), \cdots, C(u^{(k)})$ is a cover of G by Eulerian subgraphs.

Let B be the set of all bases of $(GF(2))^k$, and put $b=|B|$. By Fact 2, every element of B induces a cover of G. We now compute the sum of the sizes of these b covers. By symmetry, every nonzero vector $u\in(GF(2))^k$ belongs to exactly $b\cdot k/(2^k-1)$ bases. Combining this with Fact 1, we conclude that every edge $f\in E$ is covered precisely $(b\cdot k/(2^k-1))\cdot 2^{k-1}$ times by the collection of all the b covers associated with the elements of B. Therefore the sum of sizes of these covers is $b\cdot k\cdot 2^{k-1}\cdot|E|/(2^k-1)$, and the average size is just the number s given in (3.1). Thus, there exists a cover of G corresponding to an element of B of size at most s. This establishes the first part of Proposition 3.1. The time bound follows from the fact that

$$b=\frac{1}{k!}\prod_{i=0}^{k-1}(2^k-2^i)\leqq\frac{1}{k!}2^{k^2}. \qquad \square$$

Next we prove the following proposition, which is an improvement of [ILPR, Cor. 1]. (There is a misprint in this corollary. $|E|+s(|V|, 2|V|-2)$ should read $|E|+2|V|-2+s(|V|, 2|V|-2)$.)

PROPOSITION 3.2. *Suppose that one can find in time $O(e^2)$ a cover of size $\leqq d\cdot e (d>1)$ for any bridgeless multigraph G with v vertices and e edges. Then we can find a cover of size $\leqq\min(de, e+(2d-1)(v-1))$ in time $O(e+v^2)$.*

In order to prove Proposition 3.2 we need the following result of [ILPR]:

LEMMA 3.3. (i) *Let $T=(V, E_T)$ be a spanning tree of a multigraph $G=(V, E)$. Then there exists an Eulerian subgraph $C=(V, E_c)$ of G with $E_c\supseteq E-E_T$. C can be found in $O(|E|)$ time.*

(ii) *Let $T=(V, E_T)$ be a spanning tree of a bridgeless multigraph G. Then there exists a bridgeless subgraph $H=(V, E_H)$ of G such that $E_H \supseteq E_T$ and $|E_H| \leqq 2|V|-2$. Such T, H can be found in $O(|E|)$ time.*

Proof of Proposition 3.2. Let $G=(V, E)$ be a bridgeless multigraph. Put $v=|V|$ and $e=|E|$. If $de \leqq e+(2d-1)(v-1)$, then $e=O(v)$ and there is nothing to show. Otherwise we argue as follows. Clearly we may assume that G is connected; otherwise apply the theorem to each of its connected components. Let $T=(V, E_T)$ be a spanning *DFS* tree of G. By Lemma 3.3(ii), there exists a bridgeless subgraph $H=(V, E_H)$ of G such that $E_H \supseteq E_T$ and $|E_H| \leqq 2v-2$. Define $\bar{G}=(V, \bar{E})$ where $\bar{E}=E-(E_H-E_T)$. Clearly T is a spanning tree of $\bar{G}$. By Lemma 3.3(i), there exists an Eulerian subgraph $C=(V, E_c)$ of $\bar{G}$ with $E_c \supseteq \bar{E}-E_T$. By assumption there exists a cover of H of size at most $d|E_H|$. This cover together with C forms a cover of G of size at most

$$d|E_H|+|E_C| \leqq d|E_H|+|\bar{E}| = d|E_H|+|E|-|E_H|+|E_T|$$
$$= e+(d-1)|E_H|+v-1 \leqq e+(d-1)(2v-2)+v-1 = e+(2d-1)(v-1).$$

T, H and C can be found in $O(e)$ time. The cover of H can be found in $O(|E_H|^2)=O(v^2)$ time. Therefore, the total time bound is $O(e+v^2)$. This completes the proof of Proposition 3.2. □

4. Consequences of nowhere-zero 8-flow and nowhere-zero 4-flow. Combining Proposition 2.1 with Propositions 2.3 and 2.4, one can easily deduce the following result of Jaeger [J2]. His result appears also in [Ma]. The proof in [ILPR] supplies the algorithms and the time bound.

LEMMA 4.1. *Every bridgeless multigraph with e edges can be covered by three Eulerian subgraphs. These subgraphs can be found in $O(e^2)$ time.*

Combining this lemma with Proposition 3.1 (with $k=3$) and Proposition 3.2, we obtain the following weaker version of Theorem 5.1:

Every bridgeless multigraph G with v vertices and e edges has a cover of size at most

$$\min\left(\tfrac{12}{7}e,\ e+\tfrac{17}{7}v-\tfrac{17}{7}\right).$$

Such a cover can be found in $O(e+v^2)$ time.

The following result [Ma] (see also J2]) is equivalent to the four color theorem:

Every planar bridgeless multigraph can be covered by two Eulerian subgraphs.

Combining this with Proposition 3.1 with $k=2$ and Proposition 3.2, we obtain:

THEOREM 4.2. *Every bridgeless planar multigraph with v vertices and e edges has a cover of size at most*

$$\min\left(\tfrac{4}{3}e,\ e+\tfrac{5}{3}v-\tfrac{5}{3}\right).$$

Similarly, Jaeger's result [J2] that every 4-edge-connected multigraph has a nowhere-zero 4-flow implies:

THEOREM 4.3. *Every 4-edge-connected multigraph with v vertices and e edges has a cover of size at most*

$$\min\left(\tfrac{4}{3}e,\ e+\tfrac{5}{3}v-\tfrac{5}{3}\right).$$

Such a cover can be found in $O(e+v^2)$ time.

5. A consequence of nowhere-zero 6-flow. In this section we prove our main result.

THEOREM 5.1. *Every bridgeless multigraph G with v vertices and e edges has a cover*

of size at most

$$\min\left(\tfrac{5}{3}e,\ e+\tfrac{7}{3}v-\tfrac{7}{3}\right).$$

Such a cover can be found in polynomial time.

Proof. By Proposition 3.2 it is enough to show that G has a cover of size $\leqq\frac{5}{3}e$. Let $G_1=(V,E)$ be an orientation of G. By Propositions 2.2 and 2.3, G_1 has a nowhere-zero $Z_2\times Z_3$-flow f_1, i.e., for every $e\in E$, $f_1(e)\in\{(1,0),(1,1),(1,2),(0,1),(0,2)\}$. For any K-flow f and $g\in K$ define

$$E(f,g)=\{e\in E: f(e)=g\}.$$

Put

$$E_1=E(f_1,(1,0))\cup E(f_1,(1,1))\cup E(f_1,(1,2)).$$

Clearly E_1 is an Eulerian subgraph of G. Let G_2 be an orientation of G in which E_1 is a directed Eulerian circuit, and let f_2 be the $Z_2\times Z_3$-flow obtained from f_1 by defining $f_2(e)=f_1(e)$ if the directions of e in G_1 and G_2 coincide, and $f_2(e)=-f_1(e)$ otherwise. Clearly there exists an i, $0\leqq i\leqq 2$, such that

$$|E(f_2,(1,i))|\geqq\tfrac{1}{3}|E_1|=\tfrac{1}{3}(|E(f_2,(1,0))|+|E(f_2,(1,1))|+|E(f_2,(1,2))|).$$

Let f_3 be the flow obtained from f_2 by letting $f_3(e)=f_2(e)$ if $e\notin E_1$ and $f_3(e)=f_2(e)-(0,i)$ if $e\in E_1$. Obviously

$$|E(f_3,(1,0))|=|E(f_2,(1,i))|\geqq\tfrac{1}{3}|E_1|. \tag{5.1}$$

Put $E_3=E(f_3,(1,0))$, $E_2=E\setminus E_3$. The second coordinate of f_3 is a nowhere-zero Z_3-flow in E_2. By Proposition 2.3 there exists a nowhere-zero 3-flow in E_2, which is, of course, also a 4-flow. By Proposition 2.3, E_2 has a Z_4-flow, and by Proposition 2.4, E_2 can be covered by two Eulerian subgraphs C_2 and C_3. By Proposition 3.1 with $k=2$, E_2 has a cover C of size at most $\frac{4}{3}|E_2|=\frac{4}{3}(|E|-|E_3|)$. In order to obtain a cover of G, we add to C an Eulerian subgraph D of G that contains E_3. There are four possibilities to such a subgraph: E_1, $E_1\oplus C_2$, $E_1\oplus C_3$ and $E_1\oplus C_2\oplus C_3$. Let D be that of smallest size. One can easily check that

$$|E_1|+|E_1\oplus C_2|+|E_1\oplus C_3|+|E_1\oplus C_2\oplus C_3|=4|E_3|+2(|E|-|E_3|)=2(|E|+|E_3|).$$

Therefore

$$|D|\leqq\frac{|E|+|E_3|}{2}.$$

Since $|D|\leqq|E_1|$, (5.1) implies

$$|E_3|\geqq\tfrac{1}{3}|D|.$$

C together with D is a cover of G of size at most

$$\begin{aligned}|D|+\tfrac{4}{3}|E|-\tfrac{4}{3}|E_3|&=\tfrac{1}{3}|D|-|E_3|+\tfrac{2}{3}|D|-\tfrac{1}{3}|E_3|+\tfrac{4}{3}|E|\\&\leqq\tfrac{2}{3}|D|-\tfrac{1}{3}|E_3|+\tfrac{4}{3}|E|\leqq\tfrac{1}{3}|E|+\tfrac{1}{3}|E_3|-\tfrac{1}{3}|E_3|+\tfrac{4}{3}|E|=\tfrac{5}{3}|E|.\end{aligned}$$

This establishes the existence of the desired cover.

We now briefly sketch an evaluation of the complexity of the construction. The constructions which are explicitly described in the proof can clearly be executed in $O(e^2)$ time.

However, the proof uses the following two existence theorems:

1. the existence of a nowhere-zero $Z_2 \times Z_3$ flow for every bridgeless multigraph;

2. the fact that one can obtain a $(Z_2)^2$ nowhere-zero flow from a given Z_3 nowhere-zero flow.

In [Yo] Younger shows that the needed $Z_2 \times Z_3$ flow can be formed in $O(v \cdot e)$ time. Statement 2 can be settled by means of maximal matching algorithms, and thus the time complexity certainly does not exceed $O(e^2)$. Thus the total time bound is at most $O(e^2)$, which can be reduced, by Proposition 3.2, to $O(e+v^2)$. □

REFERENCES

[ILPR] A. ITAI, R. J. LIPTON, C. H. PAPADIMITRIOU AND M. RODEH, *Covering graphs by simple circuits*, SIAM J. Comput., 10 (1981), pp. 746-750.

[IR] A. ITAI AND M. RODEH, *Covering a graph by circuits*, Proc. ICALP Conf., Udine, 1978.

[J1] F. JAEGER, *Flows and generalized coloring theorems in graphs*, J. Combin. Theory B, 26 (1979), pp. 205-216.

[J2] ———, *On nowhere-zero flows in multigraphs*, in Proc. Fifth British Combinatorial Conference, Aberdeen, 1975, Utilitas Mathematica, pp. 373-378.

[Ma] K. R. MATTHEWS, *On the Eulericity of a graph*, J. Graph Theory, 2 (1978), pp. 143-148.

[Se] P. D. SEYMOUR, *Nowhere-zero 6-flows*, J. Combin. Theory B, 30 (1981), pp. 130-135.

[Tu] W. T. TUTTE, *A contribution to the theory of chromatic polynomials*, Canad. J. Math., 6 (1954), pp. 80-91.

[BJJ] J. C. BERMOND, B. JACKSON AND F. JAEGER, *Shortest coverings of graphs with cycles*, preprint.

[Yo] D. H. YOUNGER, *Integer flows*, J. Graph Theory, 7 (1983), pp. 349-357.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 3, July 1985

002

CONVEX SETS OF HERMITIAN MATRICES WITH CONSTANT INERTIA*

CHARLES R. JOHNSON† AND LEIBA RODMAN‡

Abstract. For n-by-n Hermitian matrices $A_1, \cdots, A_m$, the situation in which all matrices in the convex hull of $A_1, \cdots, A_m$ have the same inertia is studied. It is shown, for example, that if $m=2$ *or* $n=2$ and the matrices are nonsingular, then they are simultaneously congruent to matrices of a special form in which the upper left principal submatrix is positive definite and its complementary principal submatrix is negative definite. The singular case is also studied, and the nonsingular case for $m>2$, $n>2$ remains open.

AMS(MOS) subject classifications. 15-A21, 15-A57, 15-A63

1. Introduction For an n-by-n Hermitian matrix A, the inertia of A is the triple

$$i(A)=(i_+(A), i_-(A), i_0(A)),$$

in which $i_+(A)$ (respectively, $i_-(A)$, $i_0(A)$) is the number of positive (respectively, negative, zero) eigenvalues of A, counting multiplicities. It is straightforward, for example using the interlacing inequalities, to make the following observation.

Observation (1). If $\hat{A}$ is a principal submatrix of the n-by-n Hermitian matrix A, then

$$i_+(A)\geqq i_+(\hat{A}) \quad \text{and} \quad i_-(A)\geqq i_-(\hat{A}).$$

It follows directly from (1) that if A has a k-by-k positive definite principal submatrix, then $i_+(A)\geqq k$. In some cases the inertia of the n-by-n Hermitian matrix A may be fully determined by such observations. For example, it also follows from (1) that if A may be partitioned as

$$A=\begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{12}^* & A_{22} & A_{23} \\ A_{13}^* & A_{23}^* & A_{33} \end{bmatrix}, \tag{2}$$

in which A_{jj} is k_j-by-k_j ($j=1,2,3$; $k_1+k_2+k_3=n$), so that A_{11} is positive definite, A_{22} is negative definite and $A_{j3}=0$, $j=1, 2, 3$, then clearly,

$$i(A)=(k_1, k_2, k_3).$$

(If $k_j=0$, the blocks involving the index j are, of course, empty.) We call an Hermitian matrix *inertia explicit* if it can be partitioned as in (2).

Two n-by-n matrices A, B are said to be *congruent* if there is a nonsingular n-by-n matrix C such that $B=C^*AC$. Since convex combinations of inertia explicit (Hermitian) matrices (same k_1, k_2, k_3) are inertia explicit, and since congruence preserves inertia of Hermitian matrices, we may make a second observation which motivates the current study.

Observation (3). Suppose that $A_1, A_2, \cdots, A_m$ are n-by-n Hermitian matrices, $i(A_j)=(k_1, k_2, k_3)$, $j=1, \cdots, m$, and suppose that there is a nonsingular, n-by-n matrix

* Received by the editors July 13, 1983, and in revised form March 15, 1984.

† Mathematical Sciences Department, Clemson University, Clemson, South Carolina 29631. The work of this author was supported by the National Science Foundation under grant MCS 80-01611 and by the Air Force Wright Aeronautical Laboratory under contract F-33615-81-K-3224, and was carried out while he was a visitor at Tel Aviv University.

‡ School of Mathematical Sciences, Tel Aviv University, Ramat Aviv, Tel Aviv 69978, Israel.

C such that

$$C^*A_jC \text{ is inertia explicit}, \qquad j=1,\cdots,m.$$

Then every convex combination

$$\alpha_1A_1+\cdots+\alpha_mA_m, \qquad \sum_{j=1}^{m}\alpha_j=1, \quad \alpha_j\geqq 0, \quad j=1,\cdots,m$$

of the A_j has the same inertia (k_1, k_2, k_3). In the situation of (3), we say that $A_1,\cdots,A_m$ are *simultaneously inertia explicit.*

For an Hermitian matrix A, put

$$S_+^F(A)=\{x\in F^n | x^*Ax>0\}, \qquad S_-^F(A)=\{x\in F^n | x^*Ax<0\},$$

where $F=\mathbb{C}$ or $F=\mathbb{R}$. Also put $S_0^F(A)=\{x\in F^n | x^*Ax=0\}$. It is straightforward to make the following observation.

Observation (4). Hermitian matrices $A_1,\cdots,A_m$ are simultaneously inertia explicit if there exist subspaces M_+, M_-, $M_0\subset\mathbb{C}^n$ such that $M_+\dotplus M_-\dotplus M_0=\mathbb{C}^n$ and

$$M_\pm\subset S_\pm^{\mathbb{C}}(A_j)\cup\{0\}, \qquad M_0\subset S_0^{\mathbb{C}}(A_j), \qquad j=1,\cdots,m.$$

The converse holds if $A_1,\cdots,A_m$ are nonsingular.

An analogous observation holds also for real symmetric matrices $A_1,\cdots,A_m$ which are simultaneously inertia explicit with a real congruence matrix (in this case one has to replace $\mathbb{C}$ by $\mathbb{R}$ in Observation (4)).

The conclusion of (3) is that matrices in the convex hull of simultaneously inertia explicit matrices have constant inertia. Our goal here is to investigate the extent to which the converse of Observation (3) holds. In particular, we show that if $n=2$, *or* if $m=2$, and one of the components of inertia is zero, then the converse does hold. An example shows that the restriction that one of the components of inertia be zero is, in general, necessary.

2. Main result; two matrices. In this section we present and demonstrate the extent to which the converse of (3) holds in case $m=2$, with dimension, n, arbitrary.

THEOREM (5). *Let A_1 and A_2 be n-by-n Hermitian matrices. Suppose that*

$$i(\alpha A_1+(1-\alpha)A_2)=i(A_1), \qquad 0\leqq\alpha\leqq 1 \tag{5a}$$

and that at least one of the numbers

$$i_+(A_1),\ i_-(A_1),\ i_0(A_1)$$

is zero. Then A_1 and A_2 are simultaneously inertia explicit; i.e. there is a nonsingular n-by-n matrix C such that C^*A_1C and C^*A_2C are inertia explicit. Furthermore, in case the matrices A_1 and A_2 are real, then the matrix C can be chosen to be real as well.

An example which indicates that (5) is not in general valid if all components of $i(A_1)$ are positive is the following.

Example (6). Let

$$A_1=\begin{bmatrix}0&1&0\\1&1&0\\0&0&0\end{bmatrix} \quad \text{and} \quad A_2=\begin{bmatrix}0&0&0\\0&1&1\\0&1&0\end{bmatrix}.$$

Then $i(A_1)=i(A_2)=i(\alpha A_1+(1-\alpha)A_2)=(1,1,1)$, $0\leqq\alpha\leqq 1$. If there were a nonsingular 3-by-3 matrix C with C^*A_1C and C^*A_2C simultaneously inertia explicit, then

the vector

$$v = C\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

would be orthogonal to the range of both A_1 and A_2. As these two ranges together span all of $\mathbb{C}^3$, the only possibility for v is the zero vector, which contradicts the required nonsingularity of C.

We make two further observations prior to demonstrating (5).

Observation (7). Let A_1, A_2 be n-by-n Hermitian matrices with A_1 nonsingular. The following statements are then equivalent:

(a) $i(\alpha A_1 + (1-\alpha)A_2)$ is constant, $0 \leqq \alpha \leqq 1$;

(b) $\det(\alpha A_1 + (1-\alpha)A_2) \neq 0$, $0 \leqq \alpha \leqq 1$;

(c) $A_1^{-1}A_2$ has no nonpositive real eigenvalues.

Proof. Statement (a) implies (b) since $\det A_1 \neq 0$, and (a) implies that $\det(\alpha A_1 + (1-\alpha)A_2)$ has the same sign as $\det A_1$; multiplication of $\alpha A_1 + (1-\alpha)A_2$ on the left by A_1^{-1} implies $\det((\alpha/(1-\alpha))I + A_1^{-1}A_2) \neq 0$, $0 \leqq \alpha < 1$, so that (b) implies (c). Converses of both implications are similar. □

So, in case $i_0(A_1) = 0$, the condition (5a) in Theorem (5) can be replaced by either of the conditions (7b) or (7c).

In the case of m nonsingular Hermitian matrices $A_1, A_2, \cdots, A_m$, it is clear that the obvious analogues of (7a) are equivalent. That is, all matrices in $\mathrm{Co}(\{A_1, \cdots, A_m\})$ have the same inertia if and only if every matrix in $\mathrm{Co}(\{A_1, \cdots, A_m\})$ is nonsingular. Here, $\mathrm{Co}(\cdot)$ denotes the (closed) convex hull of a set. Thus, study of convex sets of Hermitian matrices with constant inertia includes the study of the structure of convex sets of nonsingular Hermitian matrices. Observation (3), for example, implies that if $A_1, \cdots, A_m$ are nonsingular and simultaneously inertia explicit, then $\mathrm{Co}(\{A_1, \cdots, A_m\})$ includes only nonsingular matrices.

Observation (8). For n-by-n positive semidefinite Hermitian matrices A_1, A_2, the following statements are equivalent:

(a) $i(\alpha A_1 + (1-\alpha)A_2)$ is constant, $0 \leqq \alpha \leqq 1$;

and

(b) $\mathrm{Ker}\, A_1 = \mathrm{Ker}\, A_2$.

Proof. Straightforwardly, (b) implies (a). Conversely, assume (a) holds. Then, for any $x \neq 0$ in the orthogonal complement of $\mathrm{Ker}\, A_1 \cap \mathrm{Ker}\, A_2$, at least one of the numbers x^*A_1x or x^*A_2x is positive, and then

$$x^*(\alpha A_1 + (1-\alpha)A_2x > 0, \qquad 0 < \alpha < 1.$$

Since the inertia of $\alpha A_1 + (1-\alpha)A_2$ is the same as that of A_1, it follows that $\dim(\mathrm{Ker}\, A_1 \cap \mathrm{Ker}\, A_2)^{\perp} \leqq \dim(\mathrm{Ker}\, A_1)^{\perp}$. This means that $\mathrm{Ker}\, A_1 \subseteq \mathrm{Ker}\, A_2$. Similarly, $\mathrm{Ker}\, A_2 \subseteq \mathrm{Ker}\, A_1$, and (b) follows. □

Note that (8) holds in the real case as well as in the complex case. (In the real case, the matrices A_1 and A_2 are considered as linear transformations on $\mathbb{R}^n$.)

For the proof of Theorem (5), we adopt the following notation. We denote by P_k, the k-by-k "backward identity" permutation matrix,

$$P_k = \begin{bmatrix} 0 & \cdots & 0 & 1 \\ \vdots & & \iddots & 0 \\ 0 & \iddots & & \vdots \\ 1 & 0 & \cdots & 0 \end{bmatrix},$$

and by Q_k, the k-by-k 0, 1 matrix with 1's only one stripe down from those in P_k,

$$\begin{bmatrix} 0 & \cdots & 0 \\ & \iddots & 1 \\ \vdots & \iddots & 0 \\ & \iddots & \vdots \\ 0 & 1 \cdots & 0 \end{bmatrix}.$$

The special matrix

$$\lambda P_k + Q_k$$

is denoted by $J_k(\lambda)$; note that $P_k J_k(\lambda) = P_k^{-1} J_k(\lambda)$ is just a basic Jordan block. The k-by-k identity matrix is denoted by I_k, where it is helpful to specify the dimension.

Proof of Theorem (5). It suffices to consider two cases: (I) $i_-(A_1) = 0$; and (II) $i_0(A_1) = 0$. The remaining a priori case, $i_+(A_1) = 0$, is trivially equivalent to case I by replacing A_j with $-A_j$, $j = 1, 2$.

Consider the first case. By (8), $\operatorname{Ker} A_1 = \operatorname{Ker} A_2$. If $r = \dim \operatorname{Ker} A_1$, then we may construct a nonsingular C so that the vectors $Ce_{n-r+1}, \cdots, Ce_n$ form a basis of $\operatorname{Ker} A_1$. Here, e_i is the ith vector in the standard basis, with a 1 in the ith position and zeros elsewhere. If A_1 and A_2 are real, then C can clearly be chosen to be real. In any event, $C^* A_1 C$ and $C^* A_2 C$ now have the necessary form.

Now consider the second case, which is equivalent to $\det(\alpha A_1 + (1-\alpha)A_2) \neq 0$, $0 \leqq \alpha \leqq 1$. (The sign of $\det(\alpha A_1 + (1-\alpha)A_2)$ is that of $(-1)^{i_-(A_1)}$ throughout the interval $0 \leqq \alpha \leqq 1$.) We employ a version of the *canonical pair form* for two Hermitian matrices, see e.g., [2, Chap. S.5] or [4], [5]. There is a nonsingular matrix T such that

$$T^* A_i T = A_{i1} \oplus \cdots \oplus A_{ir}, \qquad i = 1, 2,$$

in which A_{ij} is k_j-by-k_j, $j = 1, \cdots, r$, and the pair A_{1j}, A_{2j} has one of the following two forms:

$$A_{1j} = P_{k_j} \quad \text{and} \quad A_{2j} = \begin{bmatrix} 0 & J_{k_{j/2}}(\lambda) \\ J_{k_{j/2}}(\bar{\lambda}) & 0 \end{bmatrix}, \tag{9}$$

in which $\operatorname{Im}(\lambda) < 0$ (here k_j is, of course, even); *or*

$$A_{1j} = \varepsilon P_{k_j}, \quad \varepsilon = \pm 1, \quad \text{and} \quad A_{2j} = J_{k_j}(\lambda), \tag{10}$$

in which $\lambda \neq 0$ is real. Evidently, via rearrangement with permutation congruence, it suffices to consider A_{1j}, A_{2j} given in the special form (9) or (10), in place of A_1 and A_2. In so doing, we shall suppress the subscripts j. (Note that the product of blocks in the canonical pair form, one from A_1^{-1} and one from A_2, is just a basic Jordan block, or pair of basic Jordan blocks, in the Jordan canonical form of $A_1^{-1}A_2$.) Furthermore, since we may replace A_2 with βA_2, $\beta > 0$, without loss of generality, we may suppose that $|\lambda|$, in either (9) or (10), is arbitrarily large or small.

Consider first the situation in which (10) holds. Since $\det(\alpha A_1 + (1-\alpha)A_2) \neq 0$, $0 \leqq \alpha \leqq 1$, it follows that $\lambda > 0$ if $\varepsilon = 1$, and $\lambda < 0$ if $\varepsilon = -1$. Suppose, for example, that $\varepsilon = 1$ and $\lambda > 0$. Note that λ is the only eigenvalue of $A_1 A_2$. Let

$$S_n = \begin{cases} \begin{bmatrix} I_{n/2} & P_{n/2} \\ P_{n/2} & -I_{n/2} \end{bmatrix} & \text{if } n \text{ is even,} \\ \begin{bmatrix} I_k & 0 & P_k \\ 0 & 1 & 0 \\ P_k & 0 & -I_k \end{bmatrix} & \text{if } n \text{ is odd,} \quad k = \dfrac{n-1}{2}. \end{cases}$$

Thus, $S_n = S_n^*$ is an n-by-n matrix, as are A_1 and A_2. A computation then verifies that

$$S_n A_1 S_n = 2\begin{bmatrix} I_p & 0 \\ 0 & -I_q \end{bmatrix}, \qquad p = \left[\frac{n+1}{2}\right], \qquad q = n - p,$$

and

$$S_n A_2 S_n = 2\lambda \begin{bmatrix} I_p & 0 \\ 0 & -I_q \end{bmatrix} + S_n J_n(0) S_n.$$

For $\lambda > 0$ sufficiently large (without loss of generality), both matrices $S_n^* A_i S_n = S_n A_i S_n$, $i = 1, 2$, are inertia explicit, as was to be shown.

Consider, then, the situation in which (9) holds and the dimension n is even. Define

$$C_t = \begin{bmatrix} (t-i)I_k & tP_k \\ tP_k & (-t-i)I_k \end{bmatrix},$$

in which $k = n/2$ and t is a real parameter. A computation then verifies that

$$C_t^* A_1 C_t = 2t^2 \begin{bmatrix} I_k & * \\ * & -I_k \end{bmatrix}$$

and

$$C_t^* A_2 C_t = 2t \begin{bmatrix} (-\operatorname{Im}(\lambda) + t \operatorname{Re}(\lambda))I_k + \frac{1}{2}[([-iE]+[-iE]^*) + t(E+E^T)] & * \\ * & (\operatorname{Im}(\lambda) - t \operatorname{Re}(\lambda))I_k + \frac{1}{2}[([-iE]+[-iE]^*) - t(E+E^T)] \end{bmatrix},$$

in which $E = PQ$. Since $\operatorname{Im}(\lambda) < 0$, $C_t^* A_j C_t$, $j = 1, 2$, are inertia explicit for $|\lambda|$ sufficiently large and $t > 0$ sufficiently small. The corresponding choice of C_t completes the proof in the situation of (9).

We have proven (5) in case A_1 and A_2 are complex Hermitian. We next suppose that A_1 and A_2 are real symmetric and wish to show that the simultaneous congruence can be achieved with a real matrix. The breakdown into cases is as before, and the proof in case I has already been noted (exactly as before using the real version of (8)).

Assume then, case II, that $\det(\alpha A_1 + (1-\alpha)A_2) \neq 0$, $0 \leqq \alpha \leqq 1$. The strategy is as before using the *real* canonical pair form for two symmetric matrices (see, e.g. [1], [3]), which is to the real Jordan canonical form as the (complex) canonical pair form was to the (complex) Jordan canonical form. There is a real nonsingular matrix T such that

$$T^* A_i T = A_{i1} \oplus \cdots \oplus A_{ir}, \qquad i = 1, 2,$$

in which A_{ij} is k_j-by-k_j, $j = 1, \cdots, r$ and the pair A_{1j}, A_{2j} has one of the following two forms:

$$(9') \qquad A_{1j} = P_{k_j} \quad \text{and} \quad A_{2j} = \begin{bmatrix} 0 & & & \Sigma \\ & & \Sigma & I_2 \\ & \iddots & \iddots & \\ \Sigma & I_2 & & 0 \end{bmatrix}, \quad \text{where} \quad \Sigma = \begin{bmatrix} -\tau & \sigma \\ \sigma & \tau \end{bmatrix},$$

with σ real and $\tau < 0$ and k_j, of course, even; *or* (10), as before.

Again, it suffices to prove the theorem for pairs A_{1j}, A_{2j} of the form (9′) or (10), and we suppress the subscripts j. Also, we may, without loss of generality, replace Σ with a positive scalar mutiple of it. In case A_1, A_2 is of the form (10), the proof is

exactly as it was before, since the matrix S_n was real. In case of the form (9'), we now use

$$C_t = \operatorname{diag}\left(\begin{bmatrix} t & 1 \\ -1 & t \end{bmatrix}, \cdots, \begin{bmatrix} t & 1 \\ -1 & t \end{bmatrix}, \begin{bmatrix} t & -1 \\ 1 & t \end{bmatrix}, \cdots, \begin{bmatrix} t & -1 \\ 1 & t \end{bmatrix}\right) + P_n,$$

if $n/2$ is even, in which there are $n/4$ blocks of the first kind and $n/4$ of the second kind and t is a real parameter, *or*

$$C_t = \operatorname{diag}\left(\begin{bmatrix} t & 1 \\ -1 & t \end{bmatrix}, \cdots, \begin{bmatrix} t & 1 \\ -1 & t \end{bmatrix}, \begin{bmatrix} a & b \\ c & d \end{bmatrix}, \begin{bmatrix} t & -1 \\ 1 & t \end{bmatrix}, \cdots, \begin{bmatrix} t & -1 \\ 1 & t \end{bmatrix}\right) + P_n,$$

if $n/2$ is odd, in which there are $(n-2)/4$ blocks of the first kind, one $\left[\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right]$, and $(n-2)/4$ blocks of the second kind and t is a real parameter. In case $n/2$ is odd, $\left[\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right]$ is chosen so that $\left[\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right]^T \Sigma \left[\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right]$ is of the form $\left[\begin{smallmatrix} u & 0 \\ 0 & -u \end{smallmatrix}\right]$ and $u > 0$ is as large as necessary. As before, calculations again show that

$$C_t^T A_i C_t, \qquad i = 1, 2,$$

are inertia explicit for appropriate values of $(\sigma^2 + \tau^2)^{1/2}$, t and u (in case $n/2$ is odd). This completes the proof of Theorem (5). □

3. The case of many matrices ($m \geqq 3$). In the case of arbitrarily many (complex Hermitian or real symmetric) n-by-n matrices $A_1, \cdots, A_m$, it is also clear, using (8), that the converse of (3) holds if the constant inertia includes $i_+ = 0$ or $i_- = 0$.

Observation (11). Suppose that $A_1, \cdots, A_m$ are n-by-n Hermitian matrices and that $i(\sum_{j=1}^m \alpha_j A_j) = (k_1, k_2, k_3)$, $\alpha_j \geqq 0$, $j = 1, \cdots, m$ and $\sum_{j=1}^m \alpha_j = 1$. If $k_1 = 0$ or if $k_2 = 0$, then $A_1, \cdots, A_m$ are simultaneously inertia explicit. Furthermore, if $A_1, \cdots, A_m$ are real, the simultaneous congruence matrix may be taken to be real.

Proof. Using (8) and appropriate convex coefficients, $\operatorname{Ker}(A_{j_1}) = \operatorname{Ker}(A_{j_2})$ for each pair j_1, j_2. Thus $\operatorname{Ker}(A_1) = \operatorname{Ker}(A_2) = \cdots = \operatorname{Ker}(A_m)$, and the construction used in the proof of (5) yields the desired result. □

COROLLARY (12). *Suppose that $A_1, \cdots, A_m$ are n-by-n Hermitian matrices. If the smallest (respectively, the largest) eigenvalue of $\sum_{j=1}^m \alpha_j A_j = \lambda_{\min}$ (respectively, $\lambda_{\max}$) with multiplicity k for all $\alpha_j \geqq 0$, $\sum_{j=1}^m \alpha_j = 1$, then all the A_j, $j = 1, \cdots, m$, have a common k-dimensional eigenspace corresponding to $\lambda_{\min}$ (respectively, $\lambda_{\max}$).*

Proof. This follows directly from Observation 11 by replacing each A_j with $A_j - \lambda_{\min} I$ (respectively, $A_j - \lambda_{\max} I$), $j = 1, \cdots, m$. □

The question of the converse of (3) remains open in general when the number m of matrices $A_1, \cdots, A_m$ is greater than two, there is constant inertia in the convex hull $\operatorname{Co}(\{A_1, \cdots, A_m\})$, and $i_0(A_j) = 0$, $j = 1, \cdots, m$. However, we are able to prove the converse of (3) for 2-by-2 real matrices.

4. The case $n = 2$ (2-by-2 matrices). In this case we note that the converse of (3) is also valid when $n = 2$, m is arbitrary and $A_1, A_2, \cdots A_m$ are real.

THEOREM (13). *Suppose that $A_1, \cdots, A_m$ are real symmetric 2-by-2 matrices. If $i(\sum_{j=1}^m \alpha_j A_j) = (k_1, k_2, k_3)$, $\sum \alpha_j = 1$, $\alpha_j \geqq 0$, $j = 1, \cdots, m$, then $A_1, \cdots, A_m$ are simultaneously inertia explicit.*

In order to prove (13), it suffices, in view of (11), to suppose $i(\Sigma \alpha_j A_j) = (1, 1, 0)$. Now S_+^F, S_-^F and S_0^F are defined relative to $F = \mathbb{R}$. The vectors in $S_0^{\mathbb{R}}(A)$ are called *isotropic vectors* for the real matrix A, and the one-dimensional subspaces of $\mathbb{R}^n$ in $S_0^{\mathbb{R}}(A)$ are called *isotropic lines*. We call $S_+^{\mathbb{R}}(A)$ the "plus set" for A and $S_-^{\mathbb{R}}(A)$ the "minus set" for A.

If A is a 2-by-2 real symmetric matrix, $i(A) = (1, 1, 0)$, then $\mathbb{R}^2$ may be partitioned as two isotropic lines for A with the plus and minus sets in between (see Fig. 1).

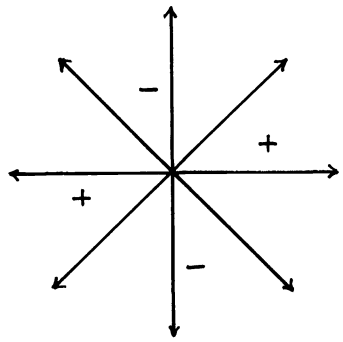


FIG. 1

In fact, the isotropic lines and the plus set determine A up to a positive scalar multiple, so that there is a one-to-one correspondence between iostropic lines and plus set pairs on the one hand, and scaled indefinite matrices on the other.

Proof of Theorem (13). It suffices to suppose that $k_1 = k_2 = 1$, and then to show that there is a real nonsingular C such that

$$C^T A_j C = \begin{bmatrix} + & * \\ * & - \end{bmatrix}, \qquad j = 1, \cdots, m.$$

Equivalently, we may show that

$$\bigcap_{j=1}^{m} S_+^{\mathbb{R}}(A_j) \neq \varnothing \quad \text{and} \quad \bigcap_{j=1}^{m} S_-^{\mathbb{R}}(A_j) \neq \varnothing.$$

Because of (5), we know that the constant inertia condition implies that for each pair the plus sets (respectively, the minus sets) must intersect.

Suppose $m = 3$. (It will turn out that this case is sufficient.) For (13) to be false, it would have to be that

(14a) $\quad S_+(A_1) \cap S_+(A_2) \neq \varnothing, \quad S_+(A_2) \cap S_+(A_3) \neq \varnothing, \quad S_+(A_1) \cap S_+(A_3) \neq \varnothing.$

and

(14b) $\quad S_-(A_1) \cap S_-(A_2) \neq \varnothing, \quad S_-(A_2) \cap S_-(A_3) \neq \varnothing, \quad S_-(A_1) \cap S_-(A_3) \neq \varnothing,$

but

(14c) $$S_+(A_1) \cap S_+(A_2) \cap S_+(A_3) = \varnothing$$

or

(14d) $$S_-(A_1) \cap S_-(A_2) \cap S_-(A_3) = \varnothing.$$

Interpreting vectors from $\mathbb{R}^2$ as complex numbers, we see that

$$S_+^{\mathbb{R}}(A_i) = \{z \in \mathbb{C} \mid \theta_j < \operatorname{Arg} z < \tau_j \text{ or } \theta_j + \pi < \operatorname{Arg} z < \tau_j + \pi\}$$

for some numbers θ_j, τ_j such that $0 \leqq \theta_j < \pi$ and $0 < \tau_j - \theta_j < \pi$. Also,

$$S_-^{\mathbb{R}}(A_j) = \{z \in \mathbb{C} \mid \tau_j < \operatorname{Arg} z < \theta_j + \pi \text{ or } \tau_j + \pi < \operatorname{Arg} z < \theta_j + 2\pi\}.$$

So the statements (14a) and (14b) are equivalent, as are (14c) and (14d). It is not

difficult to see that statement (14a) together with (14c) can happen only if

$$\theta_2 < \theta_1 < \tau_2 < \theta_3 < \tau_1 < \theta_2 + \pi < \tau_3$$

(possibly after a permutation of indices $\{1, 2, 3, \}$). But then for any set of signs ε_1, ε_2, $\varepsilon_3 = \pm 1$ such that exactly one of them is -1, we have

$$S_+(\varepsilon_1 A_1) \cap S_+(\varepsilon_2 A_2) \cap S_+(\varepsilon_3 A_3) \neq \varnothing$$

and

$$S_-(\varepsilon_1 A_1) \cap S_-(\varepsilon_2 A_2) \cap S_-(\varepsilon_3 A_3) \neq \varnothing.$$

Hence, using Observation 3, we obtain that if (13) is false, then

$$\det(\alpha_1 A_1 + \alpha_2 A_2 + \alpha_3 A_3) \neq 0$$

as long as any two of the real coefficients α_i are positive (or any two are negative). We shall obtain a contradiction (and thereby prove Theorem (13)) by showing that there are coefficients α_1, α_2, α_3, two of which are of the same nonzero sign, such that the matrix $\alpha_1 A_1 + \alpha_2 A_2 + \alpha_3 A_3$ is singular.

Since a simultaneous congruence does not change anything relevant to the problem, we may assume one of the matrices, say A_3, is

$$\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}.$$

Let

$$A_1 = \begin{pmatrix} a_{11} & a_{12} \\ a_{12} & a_{13} \end{pmatrix} \quad \text{and} \quad A_2 = \begin{pmatrix} a_{21} & a_{22} \\ a_{22} & a_{23} \end{pmatrix}.$$

Write

$$A = A_1 - \frac{a_{13}}{a_{23}} A_2 + \left(\frac{a_{13} a_{22}}{a_{23}} - a_{12} \right) A_3.$$

Then A is diagonal and has a zero entry in the lower right. Similarly we may try

$$A = A_1 - \frac{a_{11}}{a_{21}} A_2 + \left(\frac{a_{11} a_{22}}{a_{21}} - a_{12} \right) A_3$$

or

$$A = A_2 - \frac{a_{21}}{a_{11}} A_1 + \left(\frac{a_{21} a_{12}}{a_{11}} - a_{22} \right) A_3$$

or

$$A = A_2 - \frac{a_{23}}{a_{13}} A_1 + \left(\frac{a_{23} a_{12}}{a_{13}} - a_{22} \right) A_3.$$

Each of these constructions produces a matrix which is singular by virtue of being diagonal with a zero diagonal entry. There are two possible ways in which all of them could fail to produce a singular linear combination of A_1, A_2 and A_3 with at least two coefficients of the same nonzero sign:

(i) $$a_{11} = a_{13} = a_{21} = a_{23} = 0, \; a_{12}, a_{22} \neq 0$$

(none of the constructions could be carried out because all the denominators would

be zero); and

(ii) $$a_{12}=\frac{a_{11}a_{22}}{a_{21}}=\frac{a_{13}a_{22}}{a_{23}}, \qquad \frac{a_{11}}{a_{21}}=\frac{a_{13}}{a_{23}}>0$$

(in each case one coefficient is zero and the others are opposite in sign).

Note that these cases have been presented so that they are mutually exclusive. In case (i), both A_1 and A_2 are nonzero multiples of A_3, and it is clear how to produce the 0 matrix with all coefficients nonzero. In case (ii), A_1 and A_2 must be the same matrix, up to a positive factor of scale, and in this event the assumed configuration of plus sets as in (14) could not occur.

If $m \geqq 4$, for (13) to be false, it would have to happen that each $(m-1)$ of $S_+(A_1), \cdots, S_+(A_m)$ intersect nontrivially while all m of them not have a common intersection. Since it is clear geometrically that this cannot occur (for $m \geqq 4$), the proof of the theorem is complete. □

5. An example: $n=m=3$, $F=\mathbb{R}$. We close with an example of three 3-by-3 nonsingular real-entried matrices which are not simultaneously inertia explicit (over the reals), but whose convex hull has constant inertia (2, 1, 0). This example was constructed by Steve Pierce, and we do not know if these matrices are simultaneously inertia explicit under complex congruence. Let

$$A_1=\begin{bmatrix}-1 & -1 & -1\\ -1 & 1 & 0\\ -1 & 0 & 1\end{bmatrix}, \quad A_2=\begin{bmatrix}-1 & -1 & 1\\ -1 & 1 & 0\\ 1 & 0 & 1\end{bmatrix}, \quad \text{and} \quad A_3=\begin{bmatrix}1 & 0 & 0\\ 0 & -1 & 0\\ 0 & 0 & 1\end{bmatrix}.$$

Then, a computation verifies that $\det(\alpha_1 A_1+\alpha_2 A_2+\alpha_3 A_3)<0$ for $\alpha_1, \alpha_2, \alpha_3 \geqq 0$, $\alpha_1+\alpha_2+\alpha_3=1$, so that the inertia of all such matrices is (2, 1, 0). However, there is no two-dimensional subspace of R^3 on which all of A_1, A_2 and A_3 are positive definite. (Of course, there is such a two-dimensional subspace for each pair, and there are one-dimensional subspaces on which all three are positive definite and on which all three are negative definite.) This observation follows from graphing two-dimensional cross sections of the isotropic sets for a fixed first coordinate.

Acknowledgment. Some of the questions addressed here were raised by Steve Pierce. They were of interest to him in connection with the geometric study of linear maps on matrices which preserve certain matricial properties, in particular the determination of those linear transformations on Hermitian matrices which preserve inertia.

REFERENCES

[1] P. A. FUHRMANN, *On symmetric rational transfer functions*, Linear Alg. and Appl., to appear.
[2] I. GOHBERG, P. LANCASTER AND L. RODMAN, *Matrix Polynomials*, Academic Press, New York, 1982.
[3] ———, *Matrices and Indefinite Scalar Products*, Birkhäuser-Verlag, Boston, 1983.
[4] R. C. THOMPSON, *Notes on the canonical pair form for Hermitian matrices*, unpublished.
[5] ———, *The characteristic polynomial of a principal subpencil of a Hermitian matrix pencil*, Linear Alg. and Appl., 14 (1976), pp. 135-177.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 3, July 1985

DOUBLY-PERIODIC SEQUENCES AND TWO-DIMENSIONAL RECURRENCES*

STEVEN HOMER† AND JERRY GOLDMAN‡

Abstract. In this paper we study doubly-periodic sequences, which are a natural generalization of (singly) periodic sequences, a class which has proven to have wide-ranging applications. We present two-dimensional recurrence relations characterizing doubly-periodic sequences over finite rings and derive a number of properties of these double sequences using power series rings in two variables. Conditions for "factoring" such sequences as tensor products over fields are given and some results of Zierler on linear recurring sequences are extended. Applications are made to automata and product codes and conditions sufficient for a set of doubly periodic sequences to be a field with respect to certain operations are given.

1. Introduction. Let R be a commutative ring with identity and let N be the set of nonnegative integers. Define the double sequence (infinite matrix) $s = (s_{ij}): N \times N \to R$ to be *doubly-periodic* if there exist positive integers p and q such that

$$s_{i+p,j} = s_{ij} = s_{i,j+q}$$

for all i, j in N. The class of such doubly-periodic sequences is closely related to the work of Nerode [12] in linear automata and to that of the authors in quadratic automata [4]. A number of authors have studied doubly-periodic sequences for application to algebraic coding theory [11], [13], [14], [15] and in particular have concentrated upon sequences with maximal periods for coding purposes. Our results generalize some power series ring results of Zierler [18] for arbitrary sequences and our tensor product result clarifies the construction of some two-dimensional arrays in [11] and [13]. Moreover we can apply the tensor product result to view doubly-periodic sequence spaces as fields under certain conditions. In this paper we apply our results to automata and also connect double-periodicity to direct product codes. Finally we mention that doubly-periodic sequences have been applied to classical tomography [16] and to the geometry of fabrics [5].

Doubly-periodic sequences are natural generalizations of periodic sequences which, together with their recurrence relations, have a long history and an enormous range of applications [1], [2], [3], [6], [7], [18]. Recall that a sequence $u = (u_i): N \to R$ is *periodic* if there is a positive integer p such that $u_{i+p} = u_i$ for all i in N. Call the least such p the *period* of the sequence (u_i). The theory of periodic sequences, often called linear recurring sequences, and their recurrence relations is well known and we rely particularly upon the basic results of Zierler [18], an exposition of which may be found in [1].

Recurrence relations with two indices have been studied in the literature [10], [17], with the relation between binomial coefficients indicating the construction of the Pascal triangle being the most famous example. Unlike the previous literature on the general theory of doubly-periodic sequences of which we are aware [11], [13], [14], we begin our study with recurrence relations over a finite commutative ring.

* Received by the editors November 9, 1981, and in final revised form April 26, 1984.

† Department of Computer Science, Boston University, Boston, Massachusetts 02215. The research of this author was partially supported by the National Science Foundation under grant MCS 82-18383.

‡ Department of Mathematical Sciences, DePaul University, Chicago, Illinois 60614. The research of this author was partially supported by a DePaul University College of Liberal Arts and Sciences faculty research grant.

2. Recurrence relations and an application to automata. For integers m and n, each $\geqq 1$, choose an element $c=(c_0, c_1, \cdots, c_m)\in R^{m+1}$ and an element $d=(d_0, \cdots, d_n)\in R^{n+1}$ with c_0 and d_0 invertible in R. Given mn arbitrary elements s_{ij}, $i=0, \cdots, m-1$ and $j=0, \cdots, n-1$ of R, recursively define a double sequence (s_{ij}) of elements of R using the following two recurrence relations on two indices:

$$(1)\qquad \sum_{k=0}^{m} c_k s_{i-k,t}=0 \quad \text{for all integers } i\geqq m \text{ and integers } t=0, 1, \cdots, n-1$$

and

$$(2)\qquad \sum_{t=0}^{n} d_t s_{k,j-t}=0 \quad \text{for all integers } j\geqq n \text{ and all integers } k\geqq 0.$$

Denote the set of all double sequences constructed in the fashion above satisfying (1) and (2) by $S(c, d)$. Observe that the sum of two such sequences as well as the product of such a sequence by an element of R (both the addition and R-multiplication operations taken componentwise) also satisfy (1) and (2). Consequently $S(c, d)$, the set of solutions to (1) and (2), is an R-module.

Now suppose (s_{ij}) is doubly-periodic in the sense of § 1. For given s_{ij}, $i=0, \cdots, p-1$ and $j=0, \cdots, q-1$, it is clear from the definition that for $m=p$, $n=q$, $c=(1, 0, \cdots, 0, -1)$, $d=(1, 0, \cdots, 0, -1)$, we have $(s_{ij})\in S(c, d)$. That is, every doubly-periodic sequence satisfies recurrences of the form (1) and (2). Our first theorem establishes the converse for a finite ring.

THEOREM 1. *Let R be a finite commutative ring with identity. For fixed $c=(c_0, \cdots, c_m)\in R^{m+1}$ and $d=(d_0, \cdots, d_n)\in R^{n+1}$ with c_0, c_m, d_0, d_n invertible, let $S(c, d)$ be the set of double sequences satisfying* (1) *and* (2). *If $(s_{ij})\in S(c, d)$ then (s_{ij}) is doubly-periodic.*

Proof. We apply the results of Zierler [18, Lemma 2] (or cf. [1, p. 372, Thm. 1]. Since (s_{ij}) satisfies (1) and c_m is invertible, for each $t=0, 1, \cdots, n-1$ the sequence $(s_{it})_{i=0}^{\infty}$ is periodic with period p_t. Set $p=$ the least common multiple of $p_0, p_1, \cdots, p_{n-1}$. Then for any $t=0, \cdots, n-1$, we have $s_{it}=s_{i+p,t}$ for any $i\in N$.

We now claim that for any $t\in N$ we have $s_{it}=s_{i+p,t}$ for each $i\in N$. The proof of this claim proceeds by induction on t. It has been established above that the claim is true for $t=0, 1, \cdots, n-1$. As induction hypothesis assume that for any $t<e$ where $e\geqq n$, we have $s_{it}=s_{i+p,t}$ for each $i\in N$. If we show that $s_{ie}=s_{i+p,e}$ for each $i\in N$ then the claim will be established. Apply (2), then the induction hypothesis to obtain, for $i\in N$,

$$s_{ie}=\frac{-(d_1 s_{i,e-1}+d_2 s_{i,e-2}+\cdots+d_n s_{i,e-n})}{d_0}$$

$$=\frac{-(d_1 s_{i+p,e-1}+d_2 s_{i+p,e-2}+\cdots+d_n s_{i+p,e-n})}{d_0}$$

$$=-s_{i+p,e} \quad \text{after another application of (2).}$$

Consequently every column of the infinite matrix (s_{ij}) is periodic with period p and the claim is proved.

Again using the results on linear recurring sequences and (2), together with d_n invertible, we have that for each $k=0, 1, \cdots, p-1$, $(s_{kj})_{j=0}^{\infty}$ is periodic with period q_k. Set $q=$ the least common multiple of $q_0, \cdots, q_{p-1}$. Then for any k, say $k=fp+r$ where

$0 \leqq r < p$, we have for any t,

$$\begin{aligned} s_{k,t+q} = s_{fp+r,t+q} &= s_{r,t+q} && \text{(using the claim proved above)} \\ &= s_{rt} && \text{(by definition of } q\text{)} \\ &= s_{kt} && \text{(by the claim again).} \end{aligned}$$

Thus q is a period of every row and (s_{ij}) is doubly-periodic. Theorem 1 is proved.

Denote the cardinality of a set A by $|A|$. As we remarked above, $S(c, d)$ is an R-module and the proof of the next theorem is immediate.

THEOREM 2. *If $R = F$, a field, and $S(c, d)$ is the set of solutions to* (1) *and* (2) *as defined at the outset in* § 2, *then $S(c, d)$ is a vector space of dimension mn over F. If, in addition, F is finite, then $|S(c, d)| = |F|^{mn}$.*

We can apply our results thus far to count the number of linear and quadratic automaton transformations over a finite field. Only one definition is necessary to state the result and we refer the reader to [4] and [12] for all other definitions. Define a double sequence $(u_{ij}): N \times N \to R$ to be *eventually doubly-periodic* if there exist positive integers N_1, N_2, p, and q such that $u_{ij} = u_{i+p,j}$ for all $i \geqq N_1$, for all $j \in N$ and $u_{ij} = u_{i,j+q}$ for all $j \geqq N_2$, for all $i \in N$. It is shown in Nerode [12] that there is a one-to-one correspondence between linear automaton transformations and eventually doubly-periodic matrices. In the event that $\frac{1}{2}$ is in the base field, the same correspondence holds for quadratic automaton transformations [4].

THEOREM 3. *The number of linear automaton transformations over a finite field F (and of quadratic automaton transformations when $\frac{1}{2} \in F$) determined by an eventually doubly-periodic sequence (u_{ij}) for given N_1, N_2, p, and q, where p and q are (least) periods as in the definition above, is*

$$|F|^{(N_1+p)(N_2+q)}.$$

Proof. It follows directly from the definition and Theorem 1 that the eventually doubly-periodic matrix (u_{ij}) corresponds to a unique doubly-periodic matrix determined by $(N_1+p)(N_2+q)$ elements u_{ij} of F with $0 \leqq i \leqq N_1+(p-1), 0 \leqq j \leqq N_2+(q-1)$ satisfying the recursions (1) and (2) with $m = N_1+p$, $n = N_2+q$. Consequently the conclusion of Theorem 3 follows directly from Theorem 2.

3. Characterizations of solutions and an application to codes. We now introduce several notions for use in the sequel. Let $R[[x, y]]$ be the ring of formal power series in two variables x and y over the ring R. Consider the polynomial rings $R[x]$, $R[y]$, and $R[x, y]$ to be subsets of $R[[x, y]]$, where we identify $a_0 + a_1x + \cdots + a_tx^t$ with the power series $a_0 + a_1x + \cdots + a_tx^t + 0 + \cdots + 0 \cdots$, etc. Each double sequence (s_{ij}) has associated with it the *generating function*, $\sum_{i=0}^{\infty} \sum_{j=0}^{\infty} s_{ij}x^iy^j$, a formal power series in $R[[x, y]]$.

Define two operators X^{-1} and Y^{-1} on double sequences in the following manner. For any double sequence (s_{ij}), let

$$X^{-1}(s_{ij}) = (s_{i+1,j}) \quad \text{and} \quad Y^{-1}(s_{ij}) = (s_{i,j+1}).$$

The operators X^{-1} and Y^{-1} commute; moreover, we interpret their powers, sums of them, and R-multiples of them in the usual fashion. Thus, one has available the operators which are polynomials in X^{-1} and Y^{-1}: i.e., for any polynomial $p(x, y) \in R[x, y]$, $p(X^{-1}, Y^{-1})$ is unambiguously defined.

Now let $s=(s_{ij})\in S(c, d)$ as in § 2. If $k=0, 1, \cdots, m, j\geqq n$, and $i\geqq k$, multiplying (2) by c_k yields

$$c_k \sum_{t=0}^{n} d_t s_{i-k,j-t}=0.$$

Summing this last relation over k produces

$$\sum_{k=0}^{m}\sum_{t=0}^{n} c_k d_t s_{i-k,j-t}=0 \quad \text{for all } i\geqq m, j\geqq n. \tag{3}$$

Therefore if we set

$$\bar{f}(x, y)=\sum_{k=0}^{m}\sum_{t=0}^{n} c_k d_t x^{m-k}y^{n-t}\in R[x, y],$$

then (3) is equivalent to the operator relation

$$\bar{f}(X^{-1}, Y^{-1})(s)=0, \tag{4}$$

for any $s\in S(c, d)$. We call the polynomial

$$f(x, y)=\sum_{k=0}^{m}\sum_{t=0}^{n} c_k d_t x^k y^t$$

the *characteristic polynomial* of (1) and (2) and call $\bar{f}$ the *reciprocal polynomial* of f. Observe that we have the formal relation $f(x, y)=x^m y^n \bar{f}(1/x, 1/y)$, which can be made rigorous in the ambient ring of formal Laurent series in two variables.

Finally, for use in the next lemma, note that $f(x, y)$ is invertible in $R[[x, y]]$ since its constant term $c_0 d_0$ is invertible in R.

LEMMA 4. *Let $s(x, y)$ be the generating function of a double sequence solution $s\in S(c, d)$ of* (1) *and* (2). *There exists a unique $a(x, y)\in R[x, y]$ of degree in $x\leqq m-1$ and degree in $y\leqq n-1$ such that*

$$s(x, y)=\frac{a(x, y)}{f(x, y)},$$

where $f(x, y)$ is the characteristic polynomial of (1) *and* (2).

Proof. The coefficient of $x^i y^j$ in $s(x, y)\cdot f(x, y)$ is

$$\sum_{k=0}^{i}\sum_{t=0}^{j} c_k d_t s_{i-k,j-t}. \tag{5}$$

But $k>m$ implies $c_k=0$ or $t>n$ implies $d_t=0$. Furthermore, (3) implies that the coefficient of $x^m y^n$ in $s(x, y)\cdot f(x, y)=0$. Thus,

$$s(x, y)\cdot f(x, y)=a(x, y)=\sum_{i=0}^{g}\sum_{j=0}^{h} a_{ij}x^i y^j,$$

where $g\leqq m-1$ and $h\leqq n-1$.

THEOREM 5. *If $R=F$, a field, $S(c, d)$ is the set of solutions to* (1) *and* (2) *as in* § 2, *and $f(x, y)$ is the characteristic polynomial of* (1) *and* (2), *then the vector subspace in $F[[x, y]]$ of generating functions of solutions in $S(c, d)$ is equal to*

$$Q_f=\left\{\frac{a(x, y)}{f(x, y)}\in F[[x, y]]\;\middle|\; x \text{ degree of } a(x, y)\leqq m-1,\ y \text{ degree of } a(x, y)\leqq n-1\right\}.$$

Proof. Lemma 4 has established containment in one direction. We now show that $x^i y^j / f(x, y) \in Q_f$ is the generating function $s(x, y)$ of a solution $s \in S(c, d)$ for any $0 \leqq i \leqq m-1, 0 \leqq j \leqq n-1$. The expression (5) computed in the proof of Lemma 4 shows that this amounts to solving the mn linear equations

$$\sum_{k=0}^{g} \sum_{t=0}^{h} c_k d_t s_{g-k,h-t} = a_{gh},$$

where $a_{ij} = 1$ and $a_{gh} = 0$ for all $0 \leqq g \leqq m-1$, $0 \leqq h \leqq n-1$, $(g, h) \neq (i, j)$, for the mn "unknowns" $s_{\alpha\beta}$.

Rewrite these equations in standard form by ordering the unknowns in (ascending) lexicographic order from the left on their indices to obtain the $mn \times mn$ matrix system

$$\begin{bmatrix} c_0d_0 & 0 & 0 & \cdots & 0 & 0 \\ c_0d_1 & c_0d_0 & 0 & \cdots & 0 & 0 \\ * & * & c_0d_0 & \cdots & 0 & 0 \\ \vdots & & & & \vdots & \\ * & * & & & c_0d_0 & 0 \\ * & * & & & * & c_0d_0 \end{bmatrix} \begin{bmatrix} s_{00} \\ s_{01} \\ \vdots \\ s_{0,n-1} \\ s_{1,0} \\ \vdots \\ s_{m-1,n-1} \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

Note that the 1 on the right-hand side occurs in the $(in+j+1)$st row. The coefficient matrix is nonsingular since its determinant is $(c_0d_0)^{mn} \neq 0$, so solutions $s_{\alpha\beta}$ exist. Extend these $s_{\alpha\beta}$ to $s \in S(c, d)$ using (1) and (2). It is now apparent that $s(x, y) \cdot f(x, y) = x^i y^j$. Thus

$$B_f = \left\{ \frac{x^i y^j}{f(x, y)} \,\middle|\, 0 \leqq i \leqq m-1, 0 \leqq j \leqq n-1 \right\}$$

is contained in the set of generating functions of solutions in $S(c, d)$.

It is easy to show that B_f is a linearly independent subset of Q_f and that the subspace it generates is Q_f of dimension mn. Since, by Theorem 2, $S(c, d)$ has dimension mn, the equality asserted in Theorem 5 is proved.

We make several observations before stating our next result. Let $c(x) = \sum_{k=0}^{m} c_k x^k$ and $d(y) = \sum_{t=0}^{n} d_t y^t$ for $c \in R^{m+1}$, $d \in R^{n+1}$ chosen as in § 2. Then $f(x, y) = c(x) \cdot d(y)$, where $f(x, y)$ is the characteristic polynomial of (1) and (2). It follows from the theory of linear recurrence relations in one index [18] that if $c(x)$ is the characteristic polynomial of the recurrence $\sum_{k=0}^{m} c_k u_{i-k} = 0$, $i \geqq m$, then the set Q_c of generating functions of solutions to this recurrence is of the form $Q_c = \{v(x)/c(x) \in F[[x]] \,|\, \text{degree } v(x) \leqq m-1\}$. In a similar manner, $d(y)$ is the characteristic polynomial of a recurrence relation $\sum_{t=0}^{n} d_t w_{j-t} = 0$, $j \geqq n$, with analogous set $Q_d = \{z(y)/d(y) \in F[[y]] \,|\, \text{degree } z(y) \leqq n-1\}$. The proof of the next corollary is now immediate from Theorem 5.

COROLLARY 6. *If $f(x, y)$ is the characteristic polynomial of* (1) *and* (2), *then $f(x, y) = c(x)d(y)$, the product of the characteristic polynomials of the associated single-index recurrences. Moreover, if Q_f, Q_c, and Q_d are the vector spaces of Theorem* 5 *and above, then $Q_f = Q_c Q_d$, the set of sums of products of generating functions of single-index recurrences.*

We now introduce polynomials which are reciprocal respectively to $c(x)$ and $d(y)$ defined prior to Corollary 6 by setting $\bar{c}(x) = \sum_{k=0}^{m} c_k x^{m-k}$ and $\bar{d}(y) = \sum_{t=0}^{n} d_t y^{n-t}$. We have $\bar{f}(x, y) = \bar{c}(x)\bar{d}(y)$ for $\bar{f}(x, y)$ the reciprocal polynomial of the characteristic polynomial $f(x, y)$ of (1) and (2). We are now in a position to state the next result.

LEMMA 7. *Let $R = F$, a field, let $\bar{c}(x)$ and $\bar{d}(y)$ be defined as above, and suppose α and β are nonzero elements of F. Set $s_{ij} = \alpha^i \beta^j$ for $i, j \in N$. Then (s_{ij}) satisfies* (1) *and* (2) *if and only if $\bar{c}(\alpha) = \bar{d}(\beta) = 0$.*

Proof. Substitution for $s_{i-k,t}$ yields

$$\sum_{k=0}^{m} c_k s_{i-k,t} = \sum_{k=0}^{m} c_k \alpha^{i-k} \beta^t = \alpha^{i-m} \beta^t \sum_{k=0}^{m} c_k \alpha^{m-k} = \alpha^{i-m} \beta^t \bar{c}(\alpha).$$

Hence (s_{ij}) satisfies (1) if and only if $\bar{c}(\alpha) = 0$. Similarly, (s_{ij}) satisfies (2) if $\bar{d}(\beta) = 0$. This proves Lemma 7.

The next theorem succinctly characterizes solutions of (1) and (2) in the event that F contains the splitting fields of $\bar{c}(x)$ and $\bar{d}(y)$. First, we informally motivate the result by displaying a notation suitable for expressing these ideas.

Assume $\rho = (r_0, r_1, r_2, \cdots)$ and $\sigma = (\sigma_0, \sigma_1, \sigma_2, \cdots)$ are infinite sequences of elements of R. Consider ρ and σ as (infinite) row vectors. Thus, ρ^t, the transpose of ρ, is an (infinite) column vector. Extend the notion of Kronecker product (or direct product) to this setting by defining

$$\rho^t \otimes \sigma = \begin{bmatrix} r_0\sigma_0 & r_0\sigma_1 & \cdots \\ r_1\sigma_0 & r_1\sigma_1 & \cdots \\ \vdots & & \end{bmatrix};$$

that is, $\rho^t \otimes \sigma$ is an (infinite) matrix whose row i is the row vector $r_i\sigma$ for $i = 0, 1, 2, \cdots$.

It is a consequence of the one index recurrence relation theory that α is a root of $\bar{c}(x)$ if and only if the sequence $a = (1, \alpha, \alpha^2, \alpha^3, \cdots)$ is a solution to the linear recurrence associated with $c(x)$ [1, p. 374]. In a like manner, the sequence $b = (1, \beta, \beta^2, \beta^3, \cdots)$ is a solution to the recurrence of the characteristic polynomial $c(y)$ iff $\bar{c}(\beta) = 0$. Lemma 7 and the above informal definition of Kronecker product show that $\bar{c}(\alpha) = 0 = \bar{d}(\beta)$ imply that $a^t \otimes b$ is a solution to (1) and (2). We show more carefully below that all solutions are linear combinations of these in the event $\bar{c}(x)$ and $\bar{d}(y)$ split into distinct linear factors in that $S(c, d)$ is isomorphic to the tensor product of $S(c)$ with $S(d)$.

THEOREM 8. *Let R be a field F. For fixed $c = (c_0, \cdots, c_m) \in F^{m+1}$ and $d = (d_0, \cdots, d_n) \in F^{n+1}$ with c_0, c_m, d_0, and d_n nonzero, let $S(c, d)$ be the set of double sequences satisfying* (1) *and* (2). *Suppose $\alpha_0, \alpha_1, \cdots, \alpha_{m-1}$ are distinct roots of $\bar{c}(x)$ and $\beta_0, \beta_1, \cdots, \beta_{n-1}$ are distinct roots of $\bar{d}(y)$ in F. Define the sequences a_i, $i = 0, 1, \cdots, m-1$ and b_j, $j = 0, 1, \cdots, n-1$ by $a_i = (1, \alpha_i, \alpha_i^2, \cdots)$ and $b_j = (1, \beta_j, \beta_j^2, \cdots)$. Then*

$$S(c, d) \cong S(c) \otimes_F S(d)$$

as tensor products of vector spaces, where $S(c)$ and $S(d)$ are the spaces of solutions to the recurrences associated with $c(x)$ and $d(y)$.

Proof. First note that all α_i and β_j are nonzero since both c_m and d_n are nonzero; consequently Lemma 7 applies here. Furthermore, it follows from Lemma 14 of [18] that $\{a_0, \cdots, a_{m-1}\}$ and $\{b_0, \cdots, b_{n-1}\}$ are linearly independent sets of solutions to their respective recurrences and that these sets are bases for the vector spaces $S(c)$ and $S(d)$. For the pair of sequences (a_i, b_j) define a map $(a_i, b_j) \mapsto D_{ij}$, from $S(c) \times S(d) \to S(c, d)$ by defining the double sequence D_{ij} to have the entry $\alpha_i^k \beta_j^t$ in its row k, column t for $k = 0, 1, 2, \cdots$ and $t = 0, 1, 2, \cdots$. It follows from Lemma 7 and the hypotheses that $D_{ij} \in S(c, d)$ for $i = 0, \cdots, m-1$ and $j = 0, \cdots, n-1$. Since the map above is a bilinear map out of the Cartesian product, $S(c) \times S(d)$, it follows from the

universal mapping property of a tensor product that there is an F-linear map ψ out of the tensor product $S(c)\otimes S(d)$ into $S(c, d)$ such that $\psi(a_i\otimes b_j)=D_{ij}$. Suppose

$$e=\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}\varepsilon_{ij}a_i\otimes b_j$$

is an element of the kernel of ψ. Then $\psi(e)=0=\sum_{i,j}\varepsilon_{ij}D_{ij}$. However, for $k=0, 1, \cdots, m-1$, row k of the right-hand side of this last relation must equal 0, which yields

$$\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}\varepsilon_{ij}\alpha_i^k b_j=\sum_j\left(\sum_i\varepsilon_{ij}\alpha_i^k\right)b_j=0.$$

But $\{b_j\}$ is linearly independent, so

$$\sum_{i=0}^{m-1}\varepsilon_{ij}\alpha_i^k=0 \quad \text{for all } j=0, \cdots, n-1. \tag{6}$$

For each fixed j the system (6) of linear equations in the unknowns $\varepsilon_{0j}, \varepsilon_{1j}, \cdots, \varepsilon_{m-1,j}$ has coefficient matrix given by

$$\begin{pmatrix}1 & \alpha_0 & \alpha_0^2 & \cdots & \alpha_0^{m-1}\\ \vdots & & & & \\ 1 & \alpha_{m-1} & \alpha_{m-1}^2 & \cdots & \alpha_{m-1}^{m-1}\end{pmatrix},$$

whose determinant is the Vandermonde determinant $\prod_{i<j}(\alpha_i-\alpha_j)\neq 0$. Thus, all $\varepsilon_{ij}=0$, therefore $e=0$ and ψ is 1-1. $S(c)\otimes S(d)$ is a vector space of dimension mn, which implies that ψ is an isomorphism and Theorem 8 is proved.

Our last theorem has an immediate application to difference codes [1], with the codewords in this case being doubly-periodic sequences. When F is a finite field there are $|F|^{mn}$ such codewords (using our Theorem 2) which satisfy relations (1) and (2).

COROLLARY 9. *With the same hypotheses as Theorem* 8 *and assuming, in addition, that F is finite, then the linear difference code* $S(c, d)$ *is the direct product of the linear difference codes* $S(c)$ *and* $S(d)$.

Proof. It follows directly from our construction that the rows of a matrix in $S(c, d)$ are codewords of $S(d)$ and that the columns are transposes of codewords of $S(c)$. But this is exactly the definition of direct product codes given in van Lint [9].

It is pointed out in [9] that when $(m, n)=1$, there is a transmission method which is effective in combatting burst errors.

4. Periods of doubly-periodic sequences. Recall from Corollary 6 that the characteristic polynomial of the relations (1) and (2) factors as $f(x, y)=c(x)\,d(y)$, where $c(x)$ and $d(y)$ are the characteristic polynomials of the associated single-index recurrences. Within this section we will often denote the set $S(c, d)$ of solutions to (1) and (2) by $S(f(x, y))$. Furthermore, we will often identify $s\in S(f(x, y))$ with its generating function $s(x, y)$. Note that any $f^*(x, y)$ in $F[x, y]$ for which the variables can be separated in the sense that $f^*(x, y)=c^*(x)\,d^*(y)$ defines associated recurrence relations of the form (1) and (2).

LEMMA 10. *Suppose* $S(c, d)$ *and* $S(c^*, d^*)$ *are solution sets of recurrences of the form* (1) *and* (2) *with associated characteristic polynomials* $f(x, y)$ *and* $f^*(x, y)$ *respectively. Then* $S(c, d)\subseteq S(c^*, d^*)$ *if and only if* $f(x, y)$ *divides* $f^*(x, y)$ *in* $F[x, y]$.

Proof. Assume that $S(c, d)\subseteq S(c^*, d^*)$. Since by Theorem 5, $1/f(x, y)$ is the generating function of a solution in $S(c, d)$, the assumed containment implies the

existence of some $a(x, y)$ such that $1/f(x, y) = a(x, y)/f^*(x, y)$, where Theorem 5 is used again. Thus $f^*(x, y) = a(x, y)f(x, y)$ or $f|f^*$.

Conversely, suppose $f(x, y) = c(x)d(y)$ and $f^*(x, y) = c^*(x)d^*(y)$ with $f|f^*$ in $F[x, y]$. It follows from the fact that $gcd(c^*(x), d^*(y)) = 1$ in $F[x, y]$ that $c(x)|c^*(x)$ in $F[x]$ and $d(y)|d^*(y)$ in $F[y]$. Thus, $c^*(x) = c_1(x)c(x)$ and $d^*(y) = d_1(y)d(y)$, yielding $f^*(x, y) = c_1(x)d_1(y)f(x, y)$. Now consider any generator of the vector space $S(c, d)$ of the form $x^i y^j/f(x, y)$. We must have

$$\frac{x^i y^j}{f(x, y)} = \frac{c_1(x)x^i d_1(y)y^j}{f^*(x, y)} \in S(c^*, d^*)$$

since the degrees in x and y of the final numerator satisfy the proper conditions. This completes the proof of Lemma 10.

THEOREM 11. *Let $f(x, y)$ and $f^*(x, y)$ be the characteristic polynomials of two-dimensional recurrence relations of the form* (1) *and* (2) *with $R = F$, a field. Then*

$$S(f(x, y)f^*(x, y)) = S(f(x, y))S(f^*(x, y))$$

where $S(f(x, y))S(f^*(x, y))$ is the set of all finite sums $\sum ss^*$ for $s \in S(f(x, y))$ and $s^* \in S(f^*(x, y))$, taken in $F[[x, y]]$.

Proof. From Lemma 10, each of $S(f(x, y))$ and $S(f^*(x, y))$ is contained in $S(f(x, y)f^*(x, y))$. It follows from Corollary 6 that all products ss^* are in $S(f(x, y)f^*(x, y))$ as well. Since the latter is an F-vector space, $S(f)S(f^*)$ is contained in $S(ff^*)$. A comparison of dimensions, using the representation of Theorem 5, finishes the proof of Theorem 11.

If $s = (s_{ij})$ is a doubly-periodic sequence, define the functions p_1 and p_2 by $p_1(s) = p$ and $p_2(s) = q$ where p and q are the least positive integers satisfying $s_{ij} = s_{i+p,j} = s_{i,j+q}$ for all i, j. If $f(x, y)$ is the characteristic polynomial of relations (1) and (2) then write $f(x, y) = c(x)d(y)$. It follows from the theory of linear recurrence relations that there is an $r > 0$ such that $c(x)$ divides $x^r - 1$ [18]. Define $\text{ord}_1(f(x, y))$ to be the smallest positive integer e such that $c(x)$ divides $x^e - 1$. Similarly, define $\text{ord}_2(f(x, y))$ to be the smallest positive integer e such that $d(y)$ divides $y^e - 1$.

THEOREM 12. *If $f(x, y) = c(x)d(y)$ and $f^*(x, y) = c^*(x)d^*(y)$ are relatively prime characteristic polynomials of two-dimensional recurrence relations of the form* (1) *and* (2) *over a finite field F, then for $s \in S(f(x, y))$ and $s^* \in S(f^*(x, y))$,*

$$p_i(s + s^*) = \text{lcm}\,(p_i(s), p_i(s^*))$$

for $i = 1, 2$.

Proof. Both s and s^* are doubly-periodic by Theorem 1. It follows that $(f(x, y), f^*(x, y)) = 1$ in $F[x, y]$ if and only if $(c(x), c^*(x)) = 1$ in $F[x]$ and $(d(y), d^*(y)) = 1$ in $F[y]$. Moreover, since the periods of a double sequence in each index are independent, we can now apply the one-dimensional theory [1], [18] to derive the conclusion of Theorem 12.

The following theorem is another immediate conclusion from the theory of linear recurrence relations.

THEOREM 13. *If $f(x, y) = c(x)d(y)$ is the characteristic polynomial of relations* (1) *and* (2) *over a finite field F and $c(x)$ and $d(y)$ are irreducible in $F[x]$ and $F[y]$ respectively, then for $0 \neq s \in S(f(x, y))$,*

$$p_i(s) = \text{ord}_i\,(f(x, y))$$

for $i = 1, 2$.

For reasons similar to those used in the previous two results, we can carry over a final conclusion regarding maximal period sequences from [1] and [18]. In fact, it follows immediately from the one-dimensional theory that if $s \in S(c, d)$ of (1) and (2) and F is finite, then $p_1(s) \leqq |F|^m - 1$ and $p_2(s) \leqq |F|^n - 1$ where m and n are the degrees of $c(x)$ and $d(y)$. Furthermore, the next theorem provides a sufficient condition for double sequences to have maximum periods (cf. [1, p. 387]).

THEOREM 14. *Let $f(x, y) = c(x)d(y)$ be the characteristic polynomial of* (1) *and* (2) *over a finite field F with* $\deg c(x) = m$ *and* $\deg d(y) = n$. *Suppose $c(x)$ divides $x^{|F|^m - 1} - 1$ but $c(x)$ divides no $x^r - 1$ for $r < |F|^m - 1$ and $d(y)$ divides $y^{|F|^n - 1} - 1$, but $d(y)$ divides no $y^r - 1$ for $r < |F|^n - 1$. If $0 \neq s \in S(f(x, y))$, then*

$$p_1(s) = |F|^m - 1 \quad \text{and} \quad p_2(s) = |F|^n - 1.$$

Sequences with maximal period are particularly important for applications in the one-dimensional case and we can use their connection with fields to obtain the following result.

THEOREM 15. *Let $f(x, y) = c(x)d(y)$ be the characteristic polynomial of* (1) *and* (2) *over a finite field F and suppose $c(x)$ and $d(y)$ are primitive and separable in $F[x]$ and $F[y]$ respectively with relatively prime degrees. Then $S(c)$ and $S(d)$ are fields, and $S(c, d)$ can be endowed with a field structure over F.*

Proof. As in [11], take $c(x)$ and $d(y)$ primitive to mean that $c(x)$ and $d(y)$ are irreducible and that $S(c)$ and $S(d)$ consist of sequences of maximal period. Furthermore, both $S(c)$ and $S(d)$ can be endowed with multiplications which give them the structure of finite fields [11, p. 1721] or [6] in which $c(x)$ and $d(y)$ respectively split. Let k be the composite of the fields $S(c)$ and $S(d)$. From Theorem 8, $S(c) \otimes_k S(d)$ and $S(c, d)$ are isomorphic as vector spaces over k. Consequently, since $S(c)$ and $S(d)$ are algebras over k, $S(c) \otimes_k S(d)$ is a k-algebra under the multiplication $(\sum a_i \otimes b_i) \cdot (\sum a_i' \otimes b_i') = \sum a_i a_i' \otimes b_i b_i'$. Take $S(c, d)$ to have the multiplication induced by this k-algebra tensor product. Since $c(x)$ is the minimum polynomial of a primitive element of $S(c)$, we can apply results from field theory [8, pp. 83-87] to conclude that $(c(x))$ is a maximal ideal of $S(d)[x]$ and that

$$S(c) \otimes_k S(d) \cong \frac{S(d)[x]}{(c(x))}.$$

Thus $S(c, d)$ is a field over $S(d)$. Since the degrees of $c(x)$ and $d(y)$ are relatively prime, $S(c, d)$ is a field over F.

5. Remarks and open problems. The carryover of one-dimensional results to their two-dimensional analogues in certain cases might lead the reader to the false conclusion that this always happens. The fact that the two-dimensional case is essentially more complicated is clear not only from the previous tensor product results but from the following considerations as well.

Let s be a double sequence. For any $t, r \in N$ define the t, r-translate of s to be the double sequence $X^{-t}Y^{-r}(s)$. Observe that s is a solution to (1) and (2) if $\bar{c}(X^{-1})(s) = 0$ and $\bar{d}(Y^{-1})(s) = 0$. Thus if $s \in S(c, d)$, then

$$\bar{c}(X^{-1})(X^{-t}Y^{-r}(s)) = X^{-t}Y^{-r}(\bar{c}(X^{-1})(s)) = 0,$$

with a similar relation resulting when $\bar{d}(Y^{-1})$ replaces $\bar{c}(X^{-1})$. Therefore $S(c, d)$ is closed under the operation of taking t, r-translates. Consequently, that a finite nonempty set A is "an F-module for finite F, closed under t, r-translations and contained in the set of doubly-periodic sequences" is a necessary condition for A to be the set of

solutions to (1) and (2). But, surprisingly, this condition, which is analogous to the one-dimensional case [18, Thm. 3], is not sufficient, as the following counterexample shows.

Consider the binary doubly-periodic sequences generated by the double recurrence relations.

$$s_{ij} + s_{i+1,j} + s_{i+2,j} = 0 \quad \text{for all } i \geqq 0, j = 0, 1, \tag{7}$$

$$s_{ij} + s_{i,j+1} + s_{i,j+2} = 0 \quad \text{for all } j \geqq 2, i \geqq 0. \tag{8}$$

The four double sequences

$$\begin{array}{cccccccccccccccccccc}
 & & & & & & & & & & & & & & & & & & & \\
 & & 0 & 0 & 0 & 0 & & & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & & & & \\
 & & 0 & 0 & 0 & 0 & \cdots & & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & \cdots & & & \\
 & & 0 & 0 & 0 & 0 & & & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & & & & \\
 & & & & \vdots & & & & & & & & \vdots & & & & & & & \\
0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & & & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & \\
1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & \cdots & & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & \cdots \\
1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & & & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & \\
 & & & & \vdots & & & & & & & & & & & \vdots & & & & &
\end{array}$$

satisfy this recurrence, for a Z_2-module, and are closed under t, r-translations as well. Clearly these are not the only sequences satisfying (7) and (8) and, since m and n are >1 here, our Theorem 2 shows that no other double recurrence relation is satisfied by exactly these four sequences. Finding a characterization of such sets A, which, in our $R[x, y]$ module approach, generalizes the one-dimensional result remains an open problem. It should be noted here that Sakata [14] has established such a result assuming that sequences form a module under the action of a certain truncated Laurent series ring.

We also mention here that recurrences similar to (1) and (2), but which instead extend mn given s_{ij}, $0 \leqq i \leqq m-1$, $0 \leqq j \leqq n-1$ first with respect to the first m rows, then with respect to columns would have served equally well as basic definitions, yielding theorems isomorphic or identical to those of this paper.

Finally we remark that there remains a wide class of open problems connected with autocorrelation of a doubly-periodic sequence and with shift registers which generate such sequences.

REFERENCES

[1] G. Birkhoff and T. Bartee, *Modern Applied Algebra*, McGraw-Hill, New York, 1970.

[2] L. Dornhoff and F. Hohn, *Applied Modern Algebra*, MacMillan, New York, 1978.

[3] L. Dickson, *History of the Theory of Numbers*, Vol. I, Chelsea, New York, 1952.

[4] J. Goldman and S. Homer, *Quadratic automata*, J. Comput. System Sci., 24 (1982), pp. 180–196.

[5] B. Grünbaum and G. Shephard, *Satins and twills: An introduction to the geometry of fabrics*, Math. Mag., 53 No. 3 (1980), pp. 139–161.

[6] M. Hall, *An isomorphism between linear recurring sequences and algebraic rings*, Trans. Amer. Math. Soc., 44 (1938), pp. 196–218.

[7] G. Hoffman de Visme, *Binary Sequences*, English Universities Press, London, 1971.

[8] N. Jacobson, *Lectures in Abstract Algebra*, Vol. III, Van Nostrand, Princeton, NJ, 1964.

[9] J. van Lint, *Coding Theory*, Springer-Verlag, Berlin, 1973.

[10] C. L. Liu, *Introduction to Combinatorial Mathematics*, McGraw-Hill, New York, 1968.

[11] F. MacWilliams and N. Sloan, *Pseudo-random sequences and arrays*, Proc. IEEE, 64 (1976), pp. 1715–1729.

[12] A. NERODE, *Linear automaton transformations*, Proc. Amer. Math. Soc., 9 (1958), pp. 541-544.
[13] T. NOMURA, H. MIYAKAWA, H. IMAI, AND A. FUKUDA, *A theory of two-dimensional linear recurring arrays*, IEEE Trans. Inform. Theory, IT-18 (1972), pp. 775-785.
[14] S. SAKATA, *General theory of double periodic arrays over an arbitrary finite field and its applications*, IEEE Trans. Inform. Theory, IT-24 (1978), pp. 719-730.
[15] ———, *On determining the independent point set for doubly periodic arrays and encoding two-dimensional cyclic codes and their duals*, IEEE Trans. Inform. Theory, IT-27 (1981), pp. 556-565.
[16] H. STARK AND R. NAAB, *Application of optimum coding sequences to computerized classical tomography*, Applied Optics, 17 (1978), pp. 3133-3137.
[17] A. WILLSKY, *Digital Signal Processing and Control and Estimation Theory*, MIT Press, Cambridge, MA, 1979.
[18] N. ZIERLER, *Linear recurring sequences*, J. Soc. Indust. Appl. Math., 7 (1959), pp. 31-48.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 3, July 1985

004

AN EXTENSION OF THE MATRIX INVERSION LEMMA*

NARIYASU MINAMIDE†

Abstract. As an extension of the matrix inversion lemma, the representation of the pseudoinverse of the sum of two matrices of the form $(S+\Phi\Phi^*)$ with S hermitian is considered by a geometric approach introducing orthogonal projections associated with the orthogonal decomposition of the related subspaces. As an example, the pseudoinverse of the matrix $(S+\phi\phi^*)$ with ϕ a vector is explicitly calculated.

1. Introduction. A system of equations

$$AXA=A, \quad XAX=X, \quad (XA)^*=XA, \quad (AX)^*=AX$$

has a unique solution for an arbitrary matrix A with complex elements (see Penrose [2]). This is called the (Moore-Penrose) pseudoinverse of A and is written as $X=A^\dagger$.

Recently, Greville [3] developed a representation for the pseudoinverse of an arbitrary matrix A partitioned as $A=[A_0, a]$ where a is a single column vector. From Greville's expression, Cline [4] inferred the structure of the representation for the pseudoinverse of a matrix A partitioned as $A=[U, V]$ in which U and V are submatrices, and extended Greville's representation to any matrix $A=[U, V]$. As a direct application of this extension, Cline [5] then developed a representation for the pseudoinverse of the sum of two nonnegative matrices.

In the present paper, a representation for the pseudoinverse of the sum of matrices of the form $(S+\Phi\Phi^*)$ with S hermitian is developed by a geometric approach based on the orthogonal decomposition of the related subspaces. Since this representation may be regarded as a generalization of the well-known matrix inversion lemma, it is called a matrix pseudoinversion lemma.

Various useful properties of the pseudoinverse are listed below. These are employed in the following discussion without explicit mention. Let $R(A)$ and $N(A)$ denote the range and null spaces of A, respectively.

(p1) $A^{\dagger\dagger}=A$, $A^{\dagger *}=A^{*\dagger}$;

(p2) $R(A^\dagger)=R(A^*)$ and $N(A^\dagger)=N(A^*)$;

(p3) $A^\dagger A$, $AA^\dagger$, $I-A^\dagger A$ and $I-AA^\dagger$ are orthogonal projections onto $R(A^*)$, $R(A)$, $N(A)$ and $N(A^*)$, respectively;

(p4) $(A^*A)^\dagger=A^\dagger A^{*\dagger}$;

(p5) $A^\dagger=(A^*A)^\dagger A^*=A^*(AA^*)^\dagger$;

(p6) If $A=A^*$, $AA^\dagger=A^\dagger A$.

2. Matrix pseudoinversion lemma. Let

$$H=S+\Phi\Phi^* \tag{2.1}$$

and consider the problem of finding a representation for the pseudoinverse of H. Here, S is an $n\times n$ hermitian matrix and Φ is an $n\times m$ matrix with complex elements.

When S is nonsingular, the representation of $H^\dagger=H^{-1}$ is already well known as the matrix inversion lemma. We are interested in the case in which S is singular.

Before proceeding directly to the representation for the pseudoinverse of (2.1), the following two important special cases are first considered.

(C_1) $H=S+\Phi\Phi^*$ with S nonnegative.

(C_2) $H=S-\Phi\Phi^*$ with H nonnegative.

* Received by the editors April 19, 1983, and in revised form April 6, 1984.

† Department of Electrical Engineering, Faculty of Engineering, Nagoya University, Furo-cho, Chikusa-ku, Nagoya, 464, Japan.

2.1. The case (C_1). Let T be an orthogonal projection onto $N(S)$, i.e., $T = I - S^\dagger S = I - SS^\dagger$.

LEMMA 2.1. *Let $H = S + \Phi\Phi^*$ with S nonnegative. Then,*

$$N(H) = N(S) \cap N(\Phi^* T). \tag{2.2}$$

Proof. Since the inclusion $N(H) \supset N(S) \cap N(\Phi^* T)$ is obvious, it suffices to show $N(H) \subset N(S) \cap N(\Phi^* T)$: Let $(S + \Phi\Phi^*)x = 0$. Then, $T(S + \Phi\Phi^*)x = T\Phi\Phi^* x = 0$, and so $\Phi^* x \in N(T\Phi)$. Let B denote the orthogonal projection onto $N(T\Phi)$, i.e., $B = I - (T\Phi)^\dagger (T\Phi)$. Then, $\Phi^* x = B\Phi^* x$. Multiplying both sides of $Sx + \Phi(B\Phi^* x) = 0$ by $S\dagger$ gives

$$x = Tx - S^\dagger \Phi B \Phi^* x. \tag{2.3}$$

Operating on (2.3) with $B\Phi^*$ and using $B\Phi^* Tx = 0$ yields

$$(I + B\Phi^* S^\dagger \Phi B) B\Phi^* x \triangleq DB\Phi^* x = 0, \tag{2.4}$$

where $D = (I + B\Phi^* S^\dagger \Phi B)$. Since, by assumption, S is nonnegative, D is nonsingular. Therefore, $\Phi^* x = B\Phi^* x = 0$ and $Sx = -\Phi\Phi^* x = 0$. Hence $x \in N(S) \cap N(\Phi^* T)$.

COROLLARY 1. *The following identity among the projections holds*:

$$HH^\dagger = SS^\dagger + (T\Phi)(T\Phi)^\dagger. \tag{2.5}$$

Proof. Since $(I - SS^\dagger)$ the orthogonal projection onto $N(S)$ commutes with $\{I - (T\Phi)(T\Phi)^\dagger\}$ an orthogonal projection onto $N(\Phi^* T)$, the product

$$(I - SS^\dagger)\{I - (T\Phi)(T\Phi)^\dagger\} = I - SS^\dagger - (T\Phi)(T\Phi)^\dagger$$

defines an orthogonal projection onto $N(S) \cap N(\Phi^* T)$. Therefore, by Lemma 2.1,

$$I - HH^\dagger = I - SS^\dagger - (T\Phi)(T\Phi)^\dagger$$

which is (2.5).

COROLLARY 2. *$R(H)$ has the following orthogonal decomposition*:

$$R(H) = R(S) \oplus R(T\Phi).$$

We now prove the following result.

THEOREM 2.1. *The pseudoinverse of $H = S + \Phi\Phi^*$ with S nonnegative is given by*

$$\begin{aligned} H^\dagger = {} & \{I - (\Phi^* T)^\dagger \Phi^*\} S^\dagger \{I - \Phi(T\Phi)^\dagger\} + (\Phi^* T)^\dagger (T\Phi)^\dagger \\ & - \{I - (\Phi^* T)^\dagger \Phi^*\} S^\dagger \Phi B D^{-1} B \Phi^* S^\dagger \{I - \Phi(T\Phi)^\dagger\} \end{aligned}$$

where

$$B = I - (T\Phi)^\dagger (T\Phi), \qquad D = I + B\Phi^* S^\dagger \Phi B.$$

Proof. By (2.5),

$$(S + \Phi\Phi^*) H^\dagger = SS^\dagger + (T\Phi)(T\Phi)^\dagger. \tag{2.6}$$

Multiplying (2.6) by T from the left yields

$$T\Phi\{\Phi^* H^\dagger - (T\Phi)^\dagger\} = 0.$$

Therefore,

$$\Phi^* H^\dagger - (T\Phi)^\dagger = B\{\Phi^* H^\dagger - (T\Phi)^\dagger\} \triangleq BY. \tag{2.7}$$

Now, multiplying (2.6) by $S^\dagger$ from the left and using (2.7) yields

$$S^\dagger SH^\dagger = S^\dagger - S^\dagger\Phi(\Phi^* H^\dagger) = S^\dagger - S^\dagger\Phi(T\Phi)^\dagger - S^\dagger\Phi BY. \tag{2.8}$$

Therefore, by (2.6), (2.7) and (2.8),

$$\begin{aligned} H^\dagger &= (HH^\dagger)H^\dagger = SS^\dagger H^\dagger + T\Phi(T\Phi)^\dagger H^\dagger = SS^\dagger H^\dagger + (\Phi^* T)^\dagger\Phi^*(I - SS^\dagger)H^\dagger \\ &= \{I - (\Phi^* T)^\dagger\Phi^*\}SS^\dagger H^\dagger + (\Phi^* T)^\dagger\Phi^* H^\dagger \qquad (2.9) \\ &= CS^\dagger C^* - CS^\dagger\Phi BY + (\Phi^* T)^\dagger(T\Phi)^\dagger \end{aligned}$$

where $C = I - (\Phi^* T)^\dagger\Phi^*$. Multiplying (2.9) by Φ^* from the left and using (2.7) yields

$$(T\Phi)^\dagger + BY = \Phi^* CS^\dagger C^* - \Phi^* CS^\dagger\Phi BY + \Phi^*(\Phi^* T)^\dagger(T\Phi)^\dagger.$$

Therefore, by the identity $\Phi^* C = B\Phi^*$,

$$(I + B\Phi^* S^\dagger\Phi B)BY = B\Phi^* S^\dagger C^*.$$

Hence,

$$BY = D^{-1}B\Phi^* S^\dagger C^*. \tag{2.10}$$

Thus, substituting (2.10) into (2.9) yields the theorem.

2.2. The case (C_2). This case may be regarded as the special case of (2.1) since $(-H)^\dagger = -H^\dagger$ and $H = S - \Phi\Phi^*$ can be rewritten as $H = -(-S + \Phi\Phi^*)$. On this observation and the analogy to the case (C_1), let

$$T = I - SS^\dagger, \qquad B = I - (T\Phi)(T\Phi)^\dagger = I,$$

$$D = I - B\Phi^* S^\dagger\Phi B = I - \Phi^* S^\dagger\Phi, \qquad Q = I - D^\dagger D,$$

where $T\Phi = 0$ since H being nonnegative implies $N(S) \subset N(\Phi^*)$ or equivalently, $R(S) \supset R(\Phi)$.

LEMMA 2.2. *Let $H = S - \Phi\Phi^*$ with H nonnegative. Then $N(H)$ has the following orthogonal decomposition*:

$$N(H) = N(S) \oplus R(S^\dagger\Phi Q).$$

Proof. $N(H) \subset N(S) \oplus R(S^\dagger\Phi Q)$: Let $(S - \Phi\Phi^*)x = 0$. Then $Sx = \Phi\Phi^* x = SS^\dagger\Phi\Phi^* x$. Therefore, $x - S^\dagger\Phi\Phi^* x \in N(S) \subset N(\Phi^*)$. Thus, $\Phi^*(x - S^\dagger\Phi\Phi^* x) = D\Phi^* x = 0$. Hence, $\Phi^* x = Q\Phi^* x \in N(D)$. Then,

$$x = (x - S^\dagger\Phi\Phi^* x) + S^\dagger\Phi\Phi^* x = T(x - S^\dagger\Phi\Phi^* x) + S^\dagger\Phi Q(\Phi^* x) \in N(S) \oplus R(S^\dagger\Phi Q).$$

$N(H) \supset N(S) \oplus R(S^\dagger\Phi Q)$: The inclusion $N(H) \supset N(S)$ being obvious, the inclusion $N(H) \supset R(S^\dagger\Phi Q)$ is shown. This follows from the observation

$$(S - \Phi\Phi^*)(S^\dagger\Phi Q) = \Phi Q - \Phi\Phi^* S^\dagger\Phi Q = \Phi(I - \Phi^* S^\dagger\Phi)Q = 0.$$

COROLLARY 3. *Let $G = S^\dagger\Phi Q$. Then,*

$$H^\dagger H = S^\dagger S - GG^\dagger. \tag{2.11}$$

We now prove the following result.

THEOREM 2.2. *The pseudoinverse of $H = S - \Phi\Phi^*$ with H nonnegative is given by*

$$H^\dagger = (I - GG^\dagger)(S^\dagger + S^\dagger\Phi D^\dagger\Phi^* S^\dagger)(I - GG^\dagger)$$

where

$$D = I - \Phi^* S^\dagger\Phi, \qquad G = S^\dagger\Phi(I - DD^\dagger).$$

Proof. Note first that since $N(H) \supset N(S)$ and $N(H) \supset R(G)$ imply $R(H) \subset R(S)$ and $R(H) \subset N(G^*)$, respectively, the following identities hold;

$$H^\dagger = S^\dagger S H^\dagger, \qquad H^\dagger = (I - GG^\dagger)H^\dagger. \tag{2.12}$$

Now, by (2.11),

$$(S - \Phi\Phi^*)H^\dagger = SS^\dagger - GG^\dagger$$

and so,

$$H^\dagger = S^\dagger S H^\dagger = S^\dagger(SS^\dagger - GG^\dagger + \Phi\Phi^* H^\dagger) = S^\dagger - S^\dagger GG^\dagger + S^\dagger\Phi\Phi^* H^\dagger. \tag{2.13}$$

Multiplying (2.13) by Φ^* from the left yields

$$\Phi^* H^\dagger = \Phi^* S^\dagger - \Phi^* S^\dagger GG^\dagger + (\Phi^* S^\dagger \Phi)\Phi^* H^\dagger.$$

Therefore,

$$(I - \Phi^* S^\dagger \Phi)\Phi^* H^\dagger = \Phi^* S^\dagger - \Phi^* S^\dagger GG^\dagger.$$

Since

$$(I - DD^\dagger)\Phi^* S^\dagger (I - GG^\dagger) = G^*(I - GG^\dagger) = 0$$

i.e., $R(\Phi^* S^\dagger - \Phi^* S^\dagger GG^\dagger) \subset R(D)$, there exists an appropriate matrix Y such that

$$\Phi^* H^\dagger = D^\dagger \Phi^* S^\dagger (I - GG^\dagger) + (I - DD^\dagger)Y. \tag{2.14}$$

Substituting (2.14) into (2.13) yields

$$H^\dagger = S^\dagger - S^\dagger GG^\dagger + S^\dagger \Phi D^\dagger \Phi^* S^\dagger (I - GG^\dagger) + GY.$$

Hence,

$$H^\dagger = (I - GG^\dagger)H^\dagger = (I - GG^\dagger)\{S^\dagger + S^\dagger \Phi D^\dagger \Phi^* S^\dagger\}(I - GG^\dagger).$$

2.3. The general case. We now consider the general case. Let

$$T = I - S^\dagger S, \qquad B = I - (T\Phi)^\dagger (T\Phi),$$
$$D = I + B\Phi^* S^\dagger \Phi B, \qquad Q = I - DD^\dagger,$$
$$G = \{I - (\Phi^* T)^\dagger \Phi^*\} S^\dagger \Phi Q.$$

By definition, it can easily be checked that

(1) $N(D) \subset N(T\Phi)$,
(2) $QB = BQ = Q$,
(3) the subspaces $\{N(S) \cap N(\Phi^* T)\}$ and $R(G)$ are orthogonal, i.e.,

$$\{N(S) \cap N(\Phi^* T)\} \perp R(G).$$

LEMMA 2.3. *Let $H = S + \Phi\Phi^*$. Then, $N(H)$ has the following orthogonal decomposition:*

$$N(H) = \{N(S) \cap N(\Phi^* T)\} \oplus R(G).$$

Proof. $N(H) \subset \{N(S) \cap N(\Phi^* T)\} \oplus R(G)$: Let $(S + \Phi\Phi^*)x = 0$. Then, by (2.3) and (2.4) in the proof of Lemma 2.1, we have $\Phi^* x = B\Phi^* x = Q\Phi^* x \in N(D)$ and

$$x = Tx - S^\dagger \Phi Q \Phi^* x. \tag{2.15}$$

Multiplying (2.15) by $(\Phi^* T)^\dagger \Phi^*$ and noting $(\Phi^* T)^\dagger \Phi^* x = 0$ yields

$$0 = (\Phi^* T)^\dagger \Phi^* T x - (\Phi^* T)^\dagger \Phi^* S^\dagger \Phi Q \Phi^* x. \tag{2.16}$$

Then, by (2.15) and (2.16)

$$\begin{aligned} x &= Tx - S^\dagger \Phi Q \Phi^* x - (T\Phi)(T\Phi)^\dagger x + (\Phi^* T)^\dagger \Phi^* S^\dagger \Phi Q \Phi^* x \\ &= \{I - (T\Phi)(T\Phi)^\dagger\} Tx - \{I - (\Phi^* T)^\dagger \Phi^*\} S^\dagger \Phi Q (\Phi^* x) \\ &\in \{N(S) \cap N(\Phi^* T)\} \oplus R(G). \end{aligned}$$

$N(H) \supset \{N(S) \cap N(\Phi^* T)\} \oplus R(G)$: The inclusion $N(H) \supset \{N(S) \cap N(\Phi^* T)\}$ being obvious, $N(H) \supset R(G)$ is shown. Observe that

$$\begin{aligned} (S + \Phi\Phi^*)\{I - (\Phi^* T)^\dagger \Phi^*\} S^\dagger \Phi Q &= SS^\dagger \Phi Q + \Phi\{I - (\Phi^* T)(\Phi^* T)^\dagger\} \Phi^* S^\dagger \Phi Q \\ &= SS^\dagger \Phi Q + \Phi B \Phi^* S^\dagger \Phi B Q \\ &= SS^\dagger \Phi Q + \Phi(D - I) Q \\ &= (SS^\dagger - I) \Phi Q = -(T\Phi) B Q = 0, \end{aligned}$$

which proves $N(H) \supset R(G)$.

COROLLARY 4. *The following identity among the projections holds*:

$$HH^\dagger = SS^\dagger + (T\Phi)(T\Phi)^\dagger - GG^\dagger. \tag{2.17}$$

We now state the main result.

THEOREM 2.3. *The pseudoinverse of* $H = S + \Phi\Phi^*$ *is given by*

$$H^\dagger = (I - GG^\dagger)\{CS^\dagger C^* + (\Phi^* T)^\dagger (T\Phi)^\dagger - CS^\dagger \Phi B D^\dagger B \Phi^* S^\dagger C^*\}(I - GG^\dagger)$$

where

$$C = I - (\Phi^* T)^\dagger \Phi^*.$$

Proof. By (2.17),

$$(S + \Phi\Phi^*)H = SS^\dagger + (T\Phi)(T\Phi)^\dagger - GG^\dagger. \tag{2.18}$$

Multiplying (2.18) by T from the left yields

$$T\Phi\Phi^* H^\dagger = T\Phi(T\Phi)^\dagger - TGG^\dagger = T\Phi\{(T\Phi)^\dagger + (T\Phi)^\dagger (\Phi^* T)^\dagger \Phi^* S^\dagger \Phi Q G^\dagger\}, \tag{2.19}$$

which shows that $\Phi^* H^\dagger$ has an expression of the form

$$\Phi^* H^\dagger = (T\Phi)^\dagger + {*}G^\dagger + BY \tag{2.20}$$

where Y is an appropriate matrix and $*$ denotes a sequence of matrix products whose precise form is of no interest. On the other hand, multiplying (2.18) by $S^\dagger$ from the left yields

$$S^\dagger S H^\dagger + S^\dagger \Phi\Phi^* H^\dagger = S^\dagger - S^\dagger GG^\dagger. \tag{2.21}$$

Substituting (2.20) into (2.21) then yields

$$S^\dagger S H^\dagger = S^\dagger - S^\dagger \Phi (T\Phi)^\dagger - S^\dagger \Phi B Y + {*}G^\dagger. \tag{2.22}$$

Hence, by (2.17), (2.20), (2.22), $R(H^\dagger) \subset N(G^\dagger)$ and $(\Phi^* T)^\dagger B = 0$,

$$\begin{aligned} H^\dagger &= (HH^\dagger)H^\dagger = SS^\dagger H^\dagger + (T\Phi)(T\Phi)^\dagger H^\dagger - GG^\dagger H^\dagger \\ &= SS^\dagger H^\dagger + (\Phi^* T)^\dagger \Phi^* (I - SS^\dagger) H^\dagger = CSS^\dagger H^\dagger + (\Phi^* T)^\dagger \Phi^* H^\dagger \\ &= CS^\dagger C^* - CS^\dagger \Phi B Y + {*}G^\dagger + (\Phi^* T)^\dagger (T\Phi)^\dagger. \end{aligned} \tag{2.23}$$

Multiplying (2.23) by Φ from the left and using $\Phi^*C = B\Phi^*$ yields

$$\begin{aligned}\Phi^*H^\dagger &= \Phi^*CS^\dagger C^* - \Phi^*CS^\dagger\Phi BY + *G^\dagger + \Phi^*(\Phi^*T)^\dagger(T\Phi)^\dagger \\ &= B\Phi^*S^\dagger C^* - B\Phi^*S^\dagger\Phi BY + *G^\dagger + (T\Phi)^\dagger.\end{aligned} \tag{2.24}$$

Equating the right-hand sides of (2.20) and (2.24) gives

$$(I + B\Phi^*S^\dagger\Phi B)BY = DBY = B\Phi^*S^\dagger C^* + *G^\dagger.$$

Therefore,

$$DBY(I - GG^\dagger) = B\Phi^*S^\dagger C^*(I - GG^\dagger).$$

Since

$$QB\Phi^*S^\dagger C^*(I - GG^\dagger) = Q\Phi^*S^\dagger C^*(I - GG^\dagger) = G^*(I - GG^\dagger) = 0$$

i.e., $R\{B\Phi^*S^\dagger C^*(I - GG^\dagger)\} \subset R(D)$, there exists an appropriate matrix Z such that

$$BY(I - GG^\dagger) = D^\dagger B\Phi^*S^\dagger C^*(I - GG^\dagger) + QZ. \tag{2.25}$$

Consequently, by (2.23) and (2.25),

$$\begin{aligned}H^\dagger &= (I - GG^\dagger)H^\dagger(I - GG^\dagger) \\ &= (I - GG^\dagger)\{CS^\dagger C^* + (\Phi^*T)^\dagger(T\Phi)^\dagger - CS^\dagger\Phi BD^\dagger B\Phi^*S^\dagger C^*\}(I - GG^\dagger) \\ &\quad - (I - GG^\dagger)CS^\dagger\Phi QZ\end{aligned}$$

which, in view of $(I - GG^\dagger)CS^\dagger\Phi QZ = 0$, proves the theorem.

2.4. An example. As an important example, the pseudoinverse of H given by

$$H = S + \phi\phi^*$$

is now calculated, where ϕ is an $n \times 1$ vector.

Case 1. $T\phi \neq 0$. In this case, we have

$$B = I - (T\Phi)^\dagger(T\Phi) = 1 - \frac{\phi^*T\phi}{\phi^*T\phi} = 0,$$

$$D = I + B\Phi^*S^\dagger\Phi B = 1,$$

$$Q = I - DD^\dagger = 0, \qquad C = I - (\Phi^*T)^\dagger\Phi^* = I - \frac{T\phi\phi^*}{\phi^*T\phi},$$

$$G = CS^\dagger\Phi Q = 0.$$

Thus, by Theorem 2.3,

$$(S + \phi\phi^*)^\dagger = \left(I - \frac{T\phi\phi^*}{\phi^*T\phi}\right)S^\dagger\left(I - \frac{\phi\phi^*T}{\phi^*T\phi}\right) + \frac{T\phi\phi^*T}{(\phi^*T\phi)^2}.$$

Case 2. $T\phi = 0$. Similarly, we have

$$B = 1, \qquad D = 1 + \phi^*S^\dagger\phi.$$

To write down Q and G explicitly, we need to consider two cases:

1) $D = 1 + \phi^*S^\dagger\phi \neq 0$,

$$Q = 0, \quad G = 0, \quad C = I.$$

Therefore

$$(S+\phi\phi^*)^\dagger = S^\dagger - S^\dagger\Phi D^\dagger\Phi^* S^\dagger = S^\dagger - \frac{S^\dagger\phi\phi^* S^\dagger}{1+\phi^* S^\dagger\phi}.$$

2) $D = 1+\phi^* S^\dagger\phi = 0,$

$$Q = 1, \quad G = S^\dagger\phi, \quad C = I.$$

Therefore

$$(S+\phi\phi^*)^\dagger = \left(I - \frac{S^\dagger\phi\phi^* S^\dagger}{\phi^* S^\dagger S^\dagger\phi}\right)S^\dagger\left(I - \frac{S^\dagger\phi\phi^* S^\dagger}{\phi^* S^\dagger S^\dagger\phi}\right).$$

REFERENCES

[1] E. H. MOORE, *General Analysis*, Mem. Amer. Philos. Soc., I, 1935.

[2] R. PENROSE, *A generalized inverse for matrices*, Proc. Cambridge Philos. Soc., 51 (1955), pp. 406-413.

[3] T. N. E. GREVILLE, *Some applications of the pseudoinverse of a matrix*, SIAM Rev., 2 (1960), pp. 15-22.

[4] R. E. CLINE, *Representations for the generalized inverse of a partitioned matrix*, SIAM J. Appl. Math., 12 (1964), pp. 588-600.

[5] ———, *Representations for the generalized inverse of sums of matrices*, SIAM J. Numer. Anal., 2 (1965), pp. 99-114.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 3, July 1985

RECURSIVE BEST APPROXIMATE SOLUTION ALGORITHMS*

NARIYASU MINAMIDE†

Abstract. Recursive best approximate solution algorithms for computing a best linear unbiased estimate to the linear multivariable estimation problem are developed by using the matrix pseudoinversion lemma. Depending on how the old data are discounted, exponential and rectangular window algorithms are proposed. The recursive form for computing the current residual error is also pursued.

1. Introduction. A least squares approach has been widely accepted as the most fundamental and useful technique for system identification. For real time computation, sequential-processing algorithms are handy and convenient. Such recursive algorithms can be derived with the help of the matrix inversion lemma. The resulting estimates may coincide with those of batch-processing algorithms provided that correct startup has been taken place.

Recently, as an application of pseudoinverses, recursive best approximate solution algorithms have been proposed by Albert and Sittler [2] for scalar estimation models (see also Albert [1]). Possible advantages of these algorithms are 1) the ability to provide with correct startup, 2) the characterization of the general solution that gives rise to the same residual error. Some generalizations of the algorithms are also considered by Boullion and Odell [3]. Their derivation is based on the representations of pseudoinverses of partitioned matrices developed by Cline [4], [5].

In the present paper, recursive algorithms for computing the best approximate solution to the linear multivariable sequential estimation problem are developed by using the matrix pseudoinversion lemma presented in the companion paper [6] (this issue, pp. 371-377). Two kinds of recursive algorithms, an exponential window algorithm and a rectangular window algorithm, together with the recursive characterization of the current residual error, are considered. The exponential window algorithm is a generalization of the one developed by Boullion and Odell [3]. The rectangular window algorithm may be new, though a similar algorithm for deleting bad observations has also been developed by Boullion and Odell [3], but their algorithm is not of recursive type.

The matrix $X_0 \in R^{n\times r}$ is a best approximate solution of the equation $AX = B$ with $A \in R^{m\times n}$ and $B \in R^{m\times r}$ if for all X, either

(I) $\|AX - B\| > \|AX_0 - B\|$ or
(II) $\|AX - B\| = \|AX_0 - B\|$ and $\|X\| > \|X_0\|$,

where $\|A\|^2 = \operatorname{tr}\{A'A\}$ (trace of $A'A$). The best approximate solution X_0 is given by $X_0 = A^{\dagger}B$ [3].

2. Recursive best approximate solution algorithms.

2.1. Exponential window. Given the measured data $\{Y_k\}$ corrupted by the noise $\{V_k\}$

$$Y_k = \Phi'_{k-1}K_* + V_k \qquad (k = 1, 2, \cdots) \tag{2.1}$$

consider the problem of recursively identifying an unknown matrix K_*, where $Y_k \in R^{m\times r}$, $\Phi_{k-1} \in R^{n\times m}$, $K_* \in R^{n\times r}$ and $V_k \in R^{m\times r}$ and Φ_{k-1} is a known matrix. It is assumed

* Received by the editors December 1, 1983, and in revised form April 6, 1984.

† Department of Electrical Engineering, Faculty of Engineering, Nagoya University, Furo-cho, Chikusa-ku, Nagoya, 464, Japan.

that $\{V_k\}$ is a white Gaussian process having element-wise uncorrelated covariance with zero mean.

Under an appropriate assumption on the magnitude of variance of V_k, a best linear unbiased estimate (BLUE) $\hat{K}_t$ based on the data up to time t is a solution with minimum norm that minimizes the functional $J_t(K)$ defined recursively as

$$\begin{aligned} &J_k(K)=\lambda_1(k)J_{k-1}(K)+\lambda_2(k)\|Y_k-\Phi'_{k-1}K\|^2 \qquad (k=1,2,\cdots,t)\\ &J_0(K)=0, \end{aligned} \tag{2.2}$$

where $0<\lambda_1(k)\leqq 1$ and $0<\lambda_2(k)$ are weighting scalars. In particular, letting $\lambda_1(k)=\lambda$ and $\lambda_2(k)=1$ in (2.2) gives an exponential weighting to the past data values.

THEOREM 2.1. *A* BLUE *solution $\hat{K}_t$ may be computed recursively from the equations*:

$$\hat{K}_t=\hat{K}_{t-1}+(\Phi'_{t-1}T_{t-1})^{\dagger}(Y_t-\Phi'_{t-1}\hat{K}_{t-1})+CP_{t-1}\Phi_{t-1}BD^{-1}B(Y_t-\Phi'_{t-1}\hat{K}_{t-1}) \tag{2.3}$$

where

$$\begin{aligned} C&=I-(\Phi'_{t-1}T_{t-1})^{\dagger}\Phi'_{t-1},\\ B&=I-(T_{t-1}\Phi_{t-1})^{\dagger}(T_{t-1}\Phi_{t-1}),\\ D&=\lambda_1(t)I+\lambda_2(t)B\Phi'_{t-1}P_{t-1}\Phi_{t-1}B, \end{aligned}$$

and P_t and T_t may be computed recursively from

$$\begin{aligned} P_t=&\frac{C}{\lambda_1(t)}[I-\lambda_2(t)P_{t-1}\Phi_{t-1}BD^{-1}B\Phi'_{t-1}]P_{t-1}C'\\ &+\frac{1}{\lambda_2(t)}(\Phi'_{t-1}T_{t-1})^{\dagger}(T_{t-1}\Phi_{t-1})^{\dagger}, \end{aligned} \tag{2.4}$$

$$T_t=T_{t-1}-(T_{t-1}\Phi_{t-1})(T_{t-1}\Phi_{t-1})^{\dagger}, \tag{2.5}$$

with the initial conditions

$$\hat{K}_0=0, \quad P_0=0, \quad T_0=I.$$

Proof. The proof is by induction. Assume that $R(\hat{K}_{t-1})\subset R(P_{t-1})$, P_{t-1} is nonnegative and symmetric and $J_{t-1}(K)$ can be expressed as

$$J_{t-1}(K)=\operatorname{tr}\{(K-\hat{K}_{t-1})'P^{\dagger}_{t-1}(K-\hat{K}_{t-1})\}+E_{t-1},$$

where E_{t-1} is the residual error that is independent of K. Then,

$$\begin{aligned} J_t(K)=&\operatorname{tr}\{\lambda_1(t)(K-\hat{K}_{t-1})'P^{\dagger}_{t-1}(K-\hat{K}_{t-1})\\ &+\lambda_2(t)(Y_t-\Phi'_{t-1}K)'(Y_t-\Phi'_{t-1}K)\}+\lambda_1(t)E_{t-1}\\ =&\operatorname{tr}[K'\{\lambda_1(t)P^{\dagger}_{t-1}+\lambda_2(t)\Phi_{t-1}\Phi'_{t-1}\}K\\ &-2K'\{\lambda_1(t)P_{t-1}\hat{K}_{t-1}+\lambda_2(t)\Phi_{t-1}Y_t\}+\lambda_1(t)\hat{K}'_{t-1}P^{\dagger}_{t-1}\hat{K}_{t-1}\\ &+\lambda_2(t)Y'_tY_t]+\lambda_1(t)E_{t-1}. \end{aligned} \tag{2.6}$$

Let

$$P^{\dagger}_t=\lambda_1(t)P^{\dagger}_{t-1}+\lambda_2(t)\Phi_{t-1}\Phi'_{t-1}. \tag{2.7}$$

Then, $P^{\dagger}_t$ and P_t are both nonnegative and symmetric. Furthermore, note that $R(P^{\dagger}_{t-1})\subset R(P_t)$ and $R(\Phi_{t-1})\subset R(P_t)$. Thus, completing the square of (2.6) gives

$$J_t(K)=\operatorname{tr}\{(K-\hat{K}_t)'P^{\dagger}_t(K-\hat{K}_t)\}+E_t,$$

where

$$\begin{aligned}\hat{K}_t &= P_t\{\lambda_1(t)P_{t-1}^{\dagger}\hat{K}_{t-1}+\lambda_2(t)\Phi_{t-1}Y_t\},\\ E_t &= \operatorname{tr}\{-\hat{K}_t'P_t^{\dagger}\hat{K}_t+\lambda_1(t)\hat{K}_{t-1}'P_{t-1}^{+}\hat{K}_{t-1}+\lambda_2(t)Y_t'Y_t\}+\lambda_1(t)E_{t-1}.\end{aligned} \tag{2.8}$$

It is seen that $K=\hat{K}_t$ is the BLUE solution that satisfies $R(\hat{K}_t)\subset R(P_t)$. Now apply [6, Thm. 2.1] to (2.7) by letting

$$S=\lambda_1(t)P_{t-1}^{\dagger}, \qquad \Phi=\sqrt{\lambda_2(t)}\Phi_{t-1}$$

and substitute the resulting P_t into (2.8). Then there follow (2.4), (2.5) and

$$\begin{aligned}\hat{K}_t &= \{I-(\Phi'T)^{\dagger}\Phi'\}\hat{K}_{t-1}-CS^{\dagger}\Phi BD^{-1}B\Phi'\hat{K}_{t-1}+CS^{\dagger}\Phi BY\\ &\quad +(\Phi'T)^{\dagger}(T\Phi)^{\dagger}\Phi Y-CS^{\dagger}\Phi BD^{-1}B\Phi'S^{\dagger}\Phi BY,\end{aligned} \tag{2.9}$$

where $Y=\sqrt{\lambda_2(t)}\,Y_t$. In deriving (2.9), $R(\hat{K}_{t-1})\subset R(P_{t-1})=R(S)$ and $C'\Phi=\Phi B$ are used. Substituting the equation

$$CS^{\dagger}\Phi BD^{-1}B\Phi'S^{\dagger}\Phi BY=CS^{\dagger}\Phi BY-CS^{\dagger}\Phi BD^{-1}BY$$

into (2.9) thus yields (2.3). Since, for $t=0$, the induction hypothesis is trivially satisfied, the induction is now complete.

COROLLARY 1. *The residual error is subject to the recursion*:

$$E_t=\lambda_1(t)E_{t-1}+\operatorname{tr}\{\lambda_2(t)(Y_t-\Phi_{t-1}'\hat{K}_{t-1})'BD^{-1}B(Y_t-\Phi_{t-1}'\hat{K}_{t-1})\}.$$

COROLLARY 2. *Suppose that the observed data are free from disturbance. Then*

(1) *$\hat{K}_t$ is equal to the orthogonal projection of K_* to the subspace spanned by the column vectors of $\{\Phi_{k-1};\ k=1,2,\cdots,t\}$, i.e., $\hat{K}_t=P_tP_t^{\dagger}K_*$.*

(2) *The general solution $\hat{K}$ satisfying*

$$Y_k-\phi_{k-1}'\hat{K}=0 \quad \text{for } k=1,2,\cdots,t$$

is given by

$$\hat{K}=\hat{K}_t+T_tK$$

where $K\in R^{n\times r}$ is arbitrary.

2.2. Rectangular window. An algorithm of computing estimate based on a finite, fixed number of past data is now derived. This is called a rectangular window algorithm of fixed length. In a recursive form, a new data point is first added and an old data point is then discarded, thus maintaining the active data length N.

Consider the following functional

$$J_{t-N+1}^{t}(K)=\sum_{k=t-N+1}^{t}\|Y_k-\Phi_{k-1}'K\|^2.$$

A solution of minimum norm that minimizes the functional $J_{t-N+1}^{t}(K)$ is denoted by $\hat{K}_{t-N+1}^{t}$.

THEOREM 2.2. *The best approximate solution $\hat{K}_{t-N+1}^{t}$ may be recursively computed by applying the following algorithms*:

(1) *The algorithm for adding the latest data point Y_t is given by Theorem* 2.1 *with $\lambda_1(t)=\lambda_2(t)=1$ and a change of notation such as*

$$(\hat{K}_{t-1},\ P_{t-1},\ T_{t-1})\to(\hat{K}_{t-N}^{t-1},\ P_{t-N}^{t-1},\ T_{t-N}^{t-1}).$$

(2) *The algorithm for discarding the old data point Y_{t-N} is given by*

$$\hat{K}_{t-N+1}^{t}=(I-G_0G_0^{\dagger})\{\hat{K}_{t-N}^{t}-P_{t-N}^{t}\Phi_{t-N-1}D_0^{\dagger}(Y_{t-N}-\Phi_{t-N-1}'\hat{K}_{t-N}^{t})\}, \tag{2.10}$$

where

$$D_0 = I - \Phi'_{t-N-1} P^t_{t-N} \Phi_{t-N-1}, \qquad G_0 = P^t_{t-N} \Phi_{t-N-1}(I - D_0^\dagger D_0)$$

and P^t_{t-N+1} *and* T^t_{t-N+1} *may be computed recursively from*

$$P^t_{t-N+1} = (I - G_0 G_0^\dagger)\{P^t_{t-N} + P^t_{t-N}\Phi_{t-N-1} D_0^\dagger \Phi'_{t-N-1} P^t_{t-N}\}(I - G_0 G_0^\dagger), \tag{2.11}$$

$$T^t_{t-N+1} = T^t_{t-N} + G_0 G_0^\dagger. \tag{2.12}$$

Initially, the adding algorithm (1) *is applied until the data length exceeds N.*

Proof. It suffices to show the discarding algorithm (2). Note first that by arguing as in the proof of Theorem 2.1, $J^t_{t-N+1}(K)$ and $J^t_{t-N}(K)$ can be expressed as

$$J^t_{t-N+1}(K) = \operatorname{tr}\{(K - \hat{K}^t_{t-N+1})'(P^t_{t-N+1})^\dagger (K - \hat{K}^t_{t-N+1})\} + E^t_{t-N+1}, \tag{2.13}$$

$$J^t_{t-N}(K) = \operatorname{tr}\{(K - \hat{K}^t_{t-N})'(P^t_{t-N})^\dagger (K - \hat{K}^t_{t-N})\} + E^t_{t-N}. \tag{2.14}$$

Here, since $\hat{K}^t_{t-N}$ can be generated from $\hat{K}^t_{t-N+1}$ by adding the old data point Y_{t-N}, these are related by

$$\begin{aligned} \hat{K}^t_{t-N} &= P^t_{t-N}\{(P^t_{t-N+1})^\dagger \hat{K}^t_{t-N+1} + \Phi_{t-N-1} Y_{t-N}\}, \\ (P^t_{t-N})^\dagger &= (P^t_{t-N+1})^\dagger + \Phi_{t-N-1}\Phi'_{t-N-1}. \end{aligned} \tag{2.15}$$

On the other hand,

$$\begin{aligned} J^t_{t-N+1}(K) &= J^t_{t-N}(K) - \| Y_{t-N} - \Phi'_{t-N-1} K \|^2 \\ &= \operatorname{tr}[K'\{(P^t_{t-N})^\dagger - \Phi_{t-N-1}\Phi'_{t-N-1}\}K \\ &\quad - 2K'\{(P^t_{t-N})^\dagger \hat{K}^t_{t-N} - \Phi_{t-N-1} Y_{t-N}\} \\ &\quad + (\hat{K}^t_{t-N})'(P^t_{t-N})^\dagger \hat{K}^t_{t-N} - Y'_{t-N} Y_{t-N}] + E^t_{t-N}. \end{aligned} \tag{2.16}$$

Operating on (2.15) with $(P^t_{t-N})^\dagger$ from the left and using $R(P^t_{t-N}) \supset R(P^t_{t-N+1})$ and $R(P^t_{t-N}) \supset R(\Phi_{t-N-1})$ yields

$$(P^t_{t-N})^\dagger \hat{K}^t_{t-N} - \Phi_{t-N-1} Y_{t-N} = (P^t_{t-N+1})^\dagger \hat{K}^t_{t-N+1}.$$

Therefore,

$$\begin{aligned} &P^t_{t-N+1}(P^t_{t-N+1})^\dagger\{(P^t_{t-N})^\dagger \hat{K}^t_{t-N} - \Phi_{t-N-1} Y_{t-N}\} \\ &\quad = (P^t_{t-N})^\dagger \hat{K}^t_{t-N} - \Phi_{t-N-1} Y_{t-N}. \end{aligned} \tag{2.17}$$

Thus, completing the square in (2.16) gives

$$J^t_{t-N+1}(K) = \operatorname{tr}\{(K - \bar{K})'(P^t_{t-N+1})^\dagger (K - \bar{K})\} + \bar{E} \tag{2.18}$$

where

$$\bar{K} = P^t_{t-N+1}\{(P^t_{t-N})^\dagger \hat{K}^t_{t-N} - \Phi_{t-N-1} Y_{t-N}\}, \tag{2.19}$$

$$\bar{E} = E^t_{t-N} - \operatorname{tr}\{\bar{K}'(P^t_{t-N+1})^\dagger \bar{K} - (\hat{K}^t_{t-N})'(P^t_{t-N})^\dagger \hat{K}^t_{t-N} + Y'_{t-N} Y_{t-N}\}. \tag{2.20}$$

It follows from (2.13) and (2.18) that

$$\hat{K}^t_{t-N+1} = \bar{K}, \qquad E^t_{t-N+1} = \bar{E}. \tag{2.21}$$

Now, applying [6, Thm. 2.2] to (2.19) by letting $S = (P^t_{t-N})^\dagger$ and $\Phi = \Phi_{t-N-1}$ and noting (2.17) yields (2.11), (2.12) and

$$\begin{aligned} K^t_{t-N+1} &= (I - G_0 G_0^\dagger)(S^\dagger + S^\dagger \Phi D_0^\dagger \Phi' S^\dagger)(S\hat{K}^t_{t-N} - \Phi Y_{t-N}) \\ &= (I - G_0 G_0^\dagger)\{(I + S^\dagger \Phi D_0^\dagger \Phi')\hat{K}^t_{t-N} - S^\dagger \Phi D_0^\dagger Y_{t-N}\} \end{aligned}$$

thus, establishing (2.10).

COROLLARY. *The residual error* E^t_{t-N+1} *is subject to the recursion*:

(1) *In an adding process,*

$$E^t_{t-N+1} = E^t_{t-N} - \operatorname{tr}\{(Y_t - \Phi'_{t-1}\hat{K}^{t-1}_{t-N})'BD^{-1}B(Y_t - \Phi'_{t-1}\hat{K}^{t-1}_{t-N})\} \tag{2.22}$$

where B and D are defined as in Theorem 2.1.

(2) *In a discarding process,*

$$E^t_{t-N+1} = E^t_{t-N} - \operatorname{tr}\{(Y_{t-N} - \Phi'_{t-N-1}\hat{K}^t_{t-N})'D_0^\dagger(Y_{t-N} - \Phi'_{t-N-1}\hat{K}^t_{t-N})\}. \tag{2.23}$$

Proof. It suffices to show (2.23). Note that, by (2.17), (2.19), (2.10) and $R(P^t_{t-N+1}) \subset N(G'_0)$,

$$\begin{aligned}\bar{K}'(P^t_{t-N+1})^\dagger \bar{K} &= (S\hat{K}^t_{t-N} - \Phi Y_{t-N})'\{\hat{K}^t_{t-N} - S^\dagger \Phi D_0^\dagger(Y_{t-N} - \Phi'\hat{K}^t_{t-N})\} \\ &= (Y_{t-N} - \Phi'\hat{K}^t_{t-N})'D_0^\dagger(Y_{t-N} - \Phi'\hat{K}^t_{t-N}) + (\hat{K}^t_{t-N})'S\hat{K}^t_{t-N} \\ &\quad - Y'_{t-N}D_0D_0^\dagger Y_{t-N} - Y'_{t-N}(I - D_0D_0^\dagger)\Phi'\hat{K}^t_{t-N}.\end{aligned} \tag{2.24}$$

Since, by (2.17),

$$G_0G_0^\dagger(S\hat{K}^t_{t-N} - \Phi Y_{t-N}) = G_0'^\dagger(\Phi'\hat{K}^t_{t-N} - Y_{t-N}) = 0$$

we have

$$\begin{aligned}Y'_{t-N}(I - D_0D_0^\dagger)\Phi'\hat{K}^t_{t-N} &= Y'_{t-N}(I - D_0D_0^\dagger)\Phi'S^\dagger S\hat{K}^t_{t-N} = Y'_{t-N}G'_0G_0'^\dagger G'_0S\hat{K}^t_{t-N} \\ &= Y'_{t-N}G'_0G_0'^\dagger\Phi'\hat{K}^t_{t-N} = Y'_{t-N}G_0^\dagger G_0Y_{t-N}.\end{aligned} \tag{2.25}$$

Thus, substituting (2.24) and (2.25) into (2.20) and using the equation $D_0D_0^\dagger + G_0^\dagger G_0 = I$, which follows from $N(D_0) \cap N(G_0) = \{0\}$, yields (2.23).

2.3. An example. Consider the scalar case ($r = m = 1$) in (2.1), i.e.,

$$y_k = \phi'_{k-1}\theta_* + v_k \qquad (k = 1, 2, \cdots),$$

where θ_* is an unknown parameter vector. A rectangular window algorithm of fixed length N that minimizes the functional

$$J^t_{t-N+1}(\theta) = \sum_{k=t-N+1}^{t} \|y_k - \phi'_{k-1}\theta\|^2$$

is given by the following recursive process.

(1) The algorithm for adding the latest data point y_t.

Case (a). $T^{t-1}_{t-N}\phi_{t-1} \neq 0$. In this case,

$$B = 0, \quad C = I - \frac{T^{t-1}_{t-N}\phi_{t-1}\phi'_{t-1}}{\phi'_{t-1}T^{t-1}_{t-N}\phi_{t-1}}, \quad D = 1$$

and so, by Theorem 2.2 and its corollary,

$$\hat{\theta}^t_{t-N} = \hat{\theta}^{t-1}_{t-N} + \frac{T^{t-1}_{t-N}\phi_{t-1}}{\phi'_{t-1}T^{t-1}_{t-N}\phi_{t-1}}(y_t - \phi'_{t-1}\hat{\theta}^{t-1}_{t-N}),$$

$$P^t_{t-N} = \left(I - \frac{T^{t-1}_{t-N}\phi_{t-1}\phi'_{t-1}}{\phi'_{t-1}T^{t-1}_{t-N}\phi_{t-1}}\right)P^{t-1}_{t-N}\left(I - \frac{\phi_{t-1}\phi'_{t-1}T^{t-1}_{t-N}}{\phi'_{t-1}T^{t-1}_{t-N}\phi_{t-1}}\right) + \frac{T^{t-1}_{t-N}\phi_{t-1}\phi'_{t-1}T^{t-1}_{t-N}}{(\phi'_{t-1}T^{t-1}_{t-N}\phi_{t-1})^2},$$

$$T^t_{t-N} = T^{t-1}_{t-N} - \frac{T^{t-1}_{t-N}\phi_{t-1}\phi'_{t-1}T^{t-1}_{t-N}}{\phi'_{t-1}T^{t-1}_{t-N}\phi_{t-1}},$$

$$E^t_{t-N} = E^{t-1}_{t-N}.$$

Case (b). $T_{t-N}^{t-1}\phi_{t-1}=0$. Similarly,

$$B=I, \quad C=I, \quad D=1+\phi_{t-1}'P_{t-N}^{t-1}\phi_{t-1}.$$

Therefore

$$\hat{\theta}_{t-N}^{t}=\hat{\theta}_{t-N}^{t-1}+\frac{P_{t-N}^{t-1}\phi_{t-1}}{1+\phi_{t-N}'P_{t-N}^{t-1}\phi_{t-1}}(y_t-\phi_{t-1}'\hat{\theta}_{t-N}^{t-1}),$$

$$P_{t-N}^{t}=\left(I-\frac{P_{t-N}^{t-1}\phi_{t-1}\phi_{t-1}'}{1+\phi_{t-1}'P_{t-N}^{t-1}\phi_{t-1}}\right)P_{t-N}^{t-1},$$

$$T_{t-N}^{t}=T_{t-N}^{t-1},$$

$$E_{t-N}^{t}=E_{t-N}^{t-1}+\frac{(y_t-\phi_{t-1}'\hat{\theta}_{t-N}^{t-1})^2}{1+\phi_{t-1}'P_{t-N}^{t-1}\phi_{t-1}}.$$

(2) The algorithm for discarding the old data point y_{t-N}:

Case (a). $D_0=1-\phi_{t-N-1}'P_{t-N}^{t}\phi_{t-N-1}\neq 0$. In this case, $G_0=0$ and so, by Theorem 2.2 and its corollary,

$$\hat{\theta}_{t-N+1}^{t}=\hat{\theta}_{t-N}^{t}-\frac{P_{t-N}^{t}\phi_{t-N-1}}{1-\phi_{t-N-1}'p_{t-N}^{t}\phi_{t-N-1}}(y_{t-N}-\phi_{t-N-1}'\hat{\theta}_{t-N}^{t}),$$

$$P_{t-N+1}^{t}=P_{t-N}^{t}+\frac{P_{t-N}^{t}\phi_{t-N-1}\phi_{t-N-1}'P_{t-N}^{t}}{1-\phi_{t-N-1}'P_{t-N}^{t}\phi_{t-N-1}},$$

$$T_{t-N+1}^{t}=T_{t-N}^{t},$$

$$E_{t-N+1}^{t}=E_{t-N}^{t}-\frac{(y_{t-N}-\phi_{t-N-1}'\hat{\theta}_{t-N}^{t})^2}{1-\phi_{t-N-1}'P_{t-N}^{t}\phi_{t-N-1}}.$$

Case (b). $D_0=0$. Similarly, $G_0=P_{t-N}^{t}\phi_{t-N-1}$ and so,

$$\hat{\theta}_{t-N+1}^{t}=\left(I-\frac{P_{t-N}^{t}\phi_{t-N-1}\phi_{t-N-1}'P_{t-N}^{t}}{\phi_{t-N-1}'(P_{t-N}^{t})^2\phi_{t-N-1}}\right)\hat{\theta}_{t-N}^{t},$$

$$P_{t-N+1}^{t}=\left(I-\frac{P_{t-N}^{t}\phi_{t-N-1}\phi_{t-N-1}'P_{t-N}^{t}}{\phi_{t-N-1}'(P_{t-N}^{t})^2\phi_{t-N-1}}\right)P_{t-N}^{t}\left(I-\frac{P_{t-N}^{t}\phi_{t-N-1}\phi_{t-N-1}'P_{t-N}^{t}}{\phi_{t-N-1}'(P_{t-N}^{t})^2\phi_{t-N-1}}\right),$$

$$T_{t-N+1}^{t}=T_{t-N}^{t}+\frac{P_{t-N}^{t}\phi_{t-N-1}\phi_{t-N-1}'P_{t-N}^{t}}{\phi_{t-N-1}'(P_{t-N}^{t})^2\phi_{t-N-1}},$$

$$E_{t-N+1}^{t}=E_{t-N}^{t}.$$

REFERENCES

[1] A. ALBERT, *Regression and the Moore-Penrose Pseudoinverse*, Academic Press, New York, 1972.

[2] A. ALBERT AND R. SITTLER, *A method for computing least squares estimators that keep up with the data*, SIAM J. Control, 3 (1965), pp. 394-417.

[3] T. L. BOULLION AND P. L. ODELL, *Generalized Inverse Matrices*, John Wiley, New York, 1971.

[4] R. E. CLINE, *Representations for the generalized inverse of a partitioned matrix*, SIAM J. Appl. Math., 12 (1964), pp. 588-600.

[5] ———, *Representations for the generalized inverse of sums of matrices*, SIAM J. Numer. Anal., 2 (1965), pp. 99-114.

[6] N. MINAMIDE, *An extension of the matrix inversion lemma*, this Journal, this issue, pp. 371-377.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 3, July 1985

006

HILL CLIMBING WITH MULTIPLE LOCAL OPTIMA*

CRAIG A. TOVEY†

Abstract. We investigate the behavior of local improvement algorithms applied to combinatorial optimization problems with multiple local optima. In general, these algorithms display two characteristics: speed and inaccuracy. This behavior is correctly predicted by our model: we show that the expected number of iterations is linear for a wide range of randomness assumptions, and that the number of local optima tends to be exponentially large. We also give some results and constructions that suggest that known NP-complete problems cannot be solved even probabilistically by any "reasonable" local improvement method.

Key words. local improvement, average performance of algorithms, clique problem

AMS(MOS) subject classifications. 68C25, 90C10

1. Introduction. In Tovey [1981] it is shown that for problems with one local optimum the average performance of local improvement algorithms is good, while the worst case performance is exponentially bad. Local improvement is also frequently used in combinatorial optimization problems with *multiple* local optima, most notably those that are NP-complete, though of course it is not guaranteed to find a global optimum in such cases. Many algorithms for "hard" combinatorial problems, such as 0-1 integer programming or the travelling salesman problem, make use of local improvement, and are justified because there is no known way to solve them exactly in a reasonable amount of time. Many artificial intelligence applications employ hill climbing, although the problems often turn out to have multiple peaks and ridges (Nilsson [1981], Winston [1977]). The obvious questions to ask are, "What are the chances of a local improvement algorithm working?" and "How long will such a method take?"

We present a structural model of local improvement that confirms its fast and inaccurate performance. That is, local improvement will quickly find a local optimum that is unlikely to be globally optimal. In the next section we extend the model used in Tovey [1981], [1983] to show that the expected number of iterations to find a local optimum is low order polynomial under a broad range of randomness assumptions and is in fact linear for a large class of distributions. In § 3 we investigate the number of local optima and address some complexity issues.

2. How fast? Consider the problem of maximizing a real valued function f defined on the vertices of the n-cube. For ease of presentation, we assume that the values of f are distinct. All that is necessary is a way of breaking ties that prevents cycling, e.g., lexicographic ordering (Dantzig [1963]). Two vertices are *adjacent* or *neighbors* if they differ in exactly one component. A vertex is a local optimum if its function value is better than any of its neighbors. For any f we can construct an *ordering*, a list of the vertices from best to worst function value. In the case where f is *local-global*—a local

* Received by the editors September 29, 1982, and in final form March 23, 1983.

† School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, Georgia 30332. The author was supported by the New Faculty Research Development Program of the Georgia Institute of Technology. This work is based on the author's Ph.D. thesis, performed under George Dantzig at Stanford University 1978-81, at the Systems Optimization Laboratory. At Stanford, his research was supported in part by the U.S. Department of Energy under contract AM03-76SF00326, PA # DE-AT03-76ER72018; the Office of Naval Research under contract N00014-75-C-0267; the National Science Foundation under grants MCS76-81259, MCS-7926009 and ECS-8012974; and the Army Research Office under contract DAS29-79-C-0110.

optimum is a global optimum—only orderings with the property, "Every vertex except the first has at least one neighbor preceding it," are possible. This case is discussed in Tovey [1982]. In this paper multiple local optima are allowed, so the set of possible orderings is not restricted. The first random distribution we consider is that all orderings are equally likely to occur. A local improvement algorithm is defined as a procedure taking the following form:

DEFINITION 2.1. *Local Improvement Algorithm.*

Step 1. Start at some random vertex x.

Step 2. Select a point y adjacent to x such that $f(y) > f(x)$. If no such y exists, stop.

Step 3. Set x equal to y and return to step 2.

Note that the selection rule in step 2 is not precise. If we always choose the neighbor with the best function value, we have the *Optimal Adjacency* (OA) Algorithm; if we choose among all better neighbors with equal probability, we have the Better Adjacency Algorithm.

(Note: in the following, e denotes the logarithmic constant.)

THEOREM 2.1. *Under the assumption that all orderings are equally likely, the expected number of iterations of any local improvement algorithm is less than* $\frac{3}{2}en$.

Proof. Let p_0 denote an arbitrary vertex. A *simple path* of length k starting at p_0 is an ordered set of vertices $\{p_0, p_1, \cdots, p_k\}$ such that the p_i are all distinct and p_i is adjacent to p_{i-1} for $i = 1, 2, \cdots, k$. Since each vertex has n neighbors, the number of simple paths of length k emanating from p_0 is not more than n^k.

We say that a path $P = \{p_0, p_1, \cdots, p_k\}$ is *improving* if $f(p_i) > f(p_{i-1})$ for all $i = 1, \cdots, k$. What is the probability that a particular path P is improving? This is the same as the probability that the sequence $(p_k, p_{k-1}, \cdots, p_0)$ is a subsequence of the ordering. Now, the $(k+1)!$ permutations of P induce a partition of the orderings into equivalence classes, each class being the set of orderings containing a particular permutation as a subsequence. Since all orderings are equally likely to occur and the partitions are of equal cardinality, the probability that P is improving is $1/(k+1)!$. Therefore, the probability that there exists some improving simple path of length k emanating from p_0 is not more than $n^k/(k+1)!$.

Then the probability that the ordering contains some improving simple path of length k is less than or equal to

$$\frac{2^n n^k}{(k+1)!} \approx \frac{2^n (en)^k}{k^k}.$$

Let $k = \frac{3}{2}en$. Then the probability is less than $2^n (\frac{2}{3})^{3en/2} < (\frac{4}{9})^n$. For instances in which no improving path of length k exists, the number of iterations must be less than k. But since the number of iterations is never worse than 2^n, the expected number of iterations must be less than

$$(\tfrac{3}{2}en - 1)(1 - (\tfrac{4}{9})^n) + (2^n)(\tfrac{4}{9})^n < \tfrac{3}{2}en. \qquad \square$$

Note that this proof applies no matter what the rule is for choosing which better adjacent vertex to go to. Even a dumb rule such as picking the worst better neighbor has an expected performance of less than $\frac{3}{2}en$. The key to the proof is the rapid growth of $k!$ compared with n^k. This allows an extension of Theorem 2.1 to a broad range of probability distributions.

THEOREM 2.2. *Suppose the ratio of probabilities of occurrence satisfies*

$$\frac{\text{Prob}[v]}{\text{Prob}[v']} \leqq 2^{\alpha n}$$

for all orderings v, v'. Then the expected number of iterations of any local improvement algorithm is less than $(\alpha+2)en$.

Proof. Let P be a simple path of length k. For each permutation q of the vertices of P construct an equivalence class $\bar{q}$ of all orderings containing q as a subsequence. Since all the equivalence classes have the same cardinality, the maximum ratio between the induced probability measures of the equivalence classes cannot exceed $2^{\alpha n}$, the maximum ratio of the probabilities of the constituent elements of the original space. Hence the probability that P is a subsequence of the random ordering cannot exceed $2^{\alpha n}/k!$. The total number of simple paths of length k is less than $2^n n^k$, so the probability that at least one such path is a subsequence of the ordering is less than

$$\frac{2^{(\alpha+1)n}n^k}{k!} < 2^{-n}, \quad \text{when } k=(\alpha+2)n.$$

The remainder of the proof is the same as for Theorem 2.1. Note that α need not be a constant but can be any polynomial in n. □

The preceding proof does not depend on the exact structure of the adjacency. All it assumes is that each vertex has n neighbors. This leads to a very general result which applies to any reasonable (data independent, polynomially many neighbors) local improvement scheme.

THEOREM 2.3. *Suppose the vertices of the hypercube are assigned neighbors in such a way that every vertex has at most $p(n)$ neighbors, where $p(n) \geqq n$ is a polynomial. Then for any probability distribution satisfying*

$$\frac{\text{Prob}[v]}{\text{Prob}[v']} \leqq 2^{\alpha n} \quad \forall \text{ orderings } v, v',$$

the expected number of iterations of any local improvement algorithm is less than $e(\alpha+2)p(n)$.

Proof. The proof is as for Theorem 2.2, where the total number of simple paths of length k is less than $2^n(p(n))^k$. Again note that α can be any polynomial in n. We remark that the hypothesis of Theorems 2.2 and 2.3 can clearly be weakened to requiring an upper bound on the probability of occurrence of sets the size of the equivalence classes, allowing some orderings to occur with zero probability. □

The assumption that all orderings are equally likely is appealing, partly because it is easily stated. However, it may not be realistic. In particular, it fails to take into account correlations between function values of neighboring points. The family of distributions in Theorem 2.2 retains the low order bound and allows for such correlation by weighting some orderings more (up to exponentially) than others. Now we consider two other classes of distributions which naturally incorporate some positive correlation between neighbors' function values. Some preliminary notions from Tovey [1982], [1981] are necessary first.

Given the function f or the ordering, we can construct an acyclic directed graph as follows:

1) Each point in the graph corresponds to a hypercube vertex.
2) For every point x that is not a local optimum, there is one directed edge (x, y) where x and y are adjacent and $f(x)<f(y)$.

When the ordering is local-global, the graph is a tree. It is called a Better Adjacency Tree, or BAT for short. We adopt the convention that all better neighbors of x are equally likely to be the father of x when BATs are randomly generated. We define the subset called Optimal Adjacency Trees, or OATs, by requiring that y be the optimal

vertex adjacent to x. For orderings in general, the graph is a forest of trees. It is evident that BAFs (Better Adjacency Forests) and OAFs (Optimal Adjacency Forests) depict the actions of the Better Adjacency and the Optimal Adjacency algorithms, respectively.

The *pathlength* of a vertex in a forest is the length of the path to the root of its tree; the mean pathlength is the sum of all pathlengths divided by the number of vertices in the forest [Knuth, 1973]. If the starting point is chosen at random, the question "How fast is the algorithm?" is equivalent to "What is the expected mean pathlength of the forest?" Similarly, Theorems 2.1 and 2.2 are statements about average height.

COROLLARY 2.4. *If all orderings are equally likely, then the expected height of* BAFs *and* OAFs *is less than* $\frac{3}{2}en$.

Let $V = V_1, V_2, \cdots, V_{2^n}$ be random variables taking as their values the vertices in the ordering. With a slight abuse of notation, let $V = v$ denote the condition $V_j = v_j$, $j = 1, \cdots, i$, where $v = v_1, \cdots, v_i$ is a list of i vertices. We define the *boundary* of v, $B(v)$, to be the set of all vertices not in v that are adjacent to one or more members of v. The property that an ordering is local-global is equivalent to the property

$$\{V_{i+1} \in B(v) \mid V = v\} \quad \forall i \forall v.$$

Any distribution on local-global problems can be specified by the conditional probabilities that $V_{i+1} = b$, given $V = v$, for each b in $B(v)$. The *boundary* distribution is defined as giving an equal chance to each boundary member. It automatically includes some positive correlation between function values of neighboring points. We can explicitly increase this correlation by giving to each $b \in B(v)$ a probability proportional to the number of neighbors it has in v. This distribution is called the *coboundary* distribution, because the coboundary of v is the set of all ordered pairs (x, b), where $x \in v$, $b \in B(v)$, and x and b are adjacent.

THEOREM 2.5 (Tovey [1981]). *The expected mean pathlength of an* OAT *or* BAT *with respect to the boundary distribution is less than* en^2.

THEOREM 2.6 (Tovey [1981]). *The expected mean pathlength of a* BAT *under the coboundary distribution is less than* $2\,en \log n$. *For* OATs *the expected mean pathlength is less than* $2\,en^2 \log n$.

We now define two classes of distributions on the non-LG problems which are extensions of the boundary and coboundary distributions. A probability distribution on orderings is said to be *boundary uniform* if all members of the boundary set of the first i vertices in the ordering have an equal probability of being the $i+1$st in the ordering. Similarly, a distribution is said to be *coboundary uniform* if the relative chances of boundary members are weighted according to the number of neighbors they have among the first i points in the ordering. There are no restrictions on vertices not in the boundary: their individual probabilities can differ widely, and the overall probability that a nonboundary member is chosen (and hence that another local optimum is introduced) can vary depending on what the first i vertices are. Since the pathlength of a starting vertex which is a local optimum is one, we can only decrease the average mean pathlength by allowing additional local optima. Therefore, the bounds in Theorems 2.5 and 2.6 extend to the class of boundary uniform and coboundary uniform distributions, respectively. We state this result in the following theorem.

THEOREM 2.7. *The expected number of iterations of the Optimal Adjacency Algorithm or Better Adjacency Algorithm, under any boundary uniform distribution, is less than* en^2. *The expected mean pathlength of a* BAF *from any coboundary uniform distribution is less than* $2en^2 \log n$.

Much like Theorem 2.1, Theorems 2.5 and 2.6 may be broadened to apply to classes of distributions in which the assumption of equal probabilities is replaced by a bound on the maximum probability. For example, the following theorem is proved in Tovey [1982]:

THEOREM 2.8. *Suppose that for some polynomial $p(n)$ the distribution on* LG *orderings satisfies*

$$\text{Prob}[V_{i+1} = x \mid V = v] \leqq \frac{p(n)}{|B(v)|} \quad \forall i, v, \quad \forall x \in B(v).$$

Then the expected mean pathlength of the BAT *is less than* $2en(p) \log n$.

As before, allowing a nonboundary vertex x a chance of being v_{i+1}, the $i+1$st vertex in the ordering, can only decrease the expected mean pathlength and height, giving:

THEOREM 2.9. *Suppose for some random distribution of orderings and some polynomial $p(n)$,*

$$\text{Prob}[V_i = x \mid V = v \text{ and } V_i \in B(v)] \leqq \frac{p(n)}{|B(v)|} \quad \forall i, v, \quad \forall x \in B(v).$$

Then the expected number of iterations of the Better Adjacency Algorithm is less than $2ep(n)n \log n$.

Theorem 2.6 may be extended similarly to coboundary semi-uniform distributions. Also, the method of proof in Theorems 2.2 and 2.3 can easily be applied to other domains. To illustrate, we derive a bound for the space of permutations on n objects, suitable for modeling many sequencing problems. The result is similar to Theorem 2.3.

THEOREM 2.10. *In the space of permutations on n objects, the expected number of iterations of any local improvement algorithm is less than $e(\alpha+2)p(n)$, where $p(n) \geqq n \log n$ is a polynomial upper bound on the number of neighbors, for any probability distribution satisfying* $\text{Prob}[v]/\text{Prob}[v'] \leqq 2^{\alpha n}$ *for all orderings v, v'.*

Proof. The number of simple paths of length k is less than or equal to $(p(n))^k n!$. The probability that there exists an improving simple path of length k is less than

$$\frac{2^{\alpha n}(p(n))^k n!}{k!} < \frac{2^{\alpha n} n!}{(\alpha+2)^k} \approx \frac{2^{\alpha n} n^n}{e^n(\alpha+2)^{e(\alpha+2)p(n)}} < \frac{2^{\alpha n}}{(\alpha+2)^{1.7(\alpha+2)p(n)}} < \frac{1}{(\alpha+2)^{p(n)}} < \frac{1}{n^n},$$

when $k = e(\alpha+2)p(n)$ and $p(n) \geqq n \log n$. Since the number of iterations cannot be more than $n!$, the expected number of iterations is less than $(1-n^{-n})k + n!/n^n \approx e(\alpha+2)p(n)$. This completes the proof. The principal difference between Theorems 2.3 and 2.10 is that the latter requires $p(n) \geqq n \log n$, while the former requires $p(n) \geqq n$. The unifying idea here is that the bound on the expected number of iterations is not less than the logarithm of the number of points in the space.

3. How inaccurate? Just as the question "How many iterations?" is equivalent to "What is the mean pathlength?" the question "How many local optima?" is the same as "How many trees are in the forest?"

PROPOSITION 3.1. *Under the assumption that all orderings are equally likely, the expected number of trees in the* OAF *or* BAF *is equal to* $(2^n)/(n+1)$.

Proof. Let x denote a vertex of the n-cube. For $x = 0$ to $2^n - 1$, let the random variable I_x equal one if x is a local optimum and zero otherwise. Then the expected number of local optima equals

$$E\left(\sum_{x=0}^{2^n-1} I_x\right) = \sum_{x=0}^{2^n-1} E(I_x). \tag{3.1.1}$$

The probability of x being a local optimum is the probability that it is the highest of $n+1$ vertices in the ordering (it and its n neighbors). If all orderings are equally likely, this probability is $1/(n+1)$. Thus

$$E(I_x)=\frac{1}{n+1}, \qquad x=0,\cdots,2^n-1. \tag{3.1.2}$$

Combining equations (3.1.1) with (3.1.2) yields the desired result. □

For problems with all orderings equally likely, then, a local improvement algorithm by itself has little guarantee of attaining a global optimum. This is true even for parallel processing versions that use multiple starting points (unless there are exponentially many).

As before, the results can be generalized to a class of distributions.

THEOREM 3.2. *Suppose that for all orderings w, w' the ratio of probabilities satisfies $P[w]/P[w']\leqq k$. Then the expected number of local optima is at least*

$$\frac{2^n}{kn+1}.$$

Proof. Let x be any vertex, and let C_x denote the set of all orderings in which x precedes all of its neighbors. Under the hypothesis, the smallest possible value of $P[C_x]$ would occur when $P[v]$ is the same for all $v\in C_x$ and is as small as possible, and when $P[v]$ is the same for all $v\notin C_x$ and is as large as possible. A simple calculation shows that this implies

$$P[v]=\frac{1}{(2^n)!}\left\lceil\frac{n+1}{kn+1}\right\rceil, \quad \forall v\in C_x.$$

Multiplication by $(2^n)!/n+1$, the cardinality of C_x, gives the desired result. □

There appears to be a considerable difference between problems that are LG and those that are not. In particular, it seems that the well-known NP-complete problems are not LG. For instance, we have the following proposition:

PROPOSITION 3.3. *In the traveling salesman problem, if two Hamiltonian circuits differ only in the order in which two consecutive cities are visited, they are called adjacent. Then with this notion of adjacency, there exists a class of instances with exponentially many local optima that are not global optima.*

Proof. As the basis for our class of instances, we use a graph with six nodes labeled a, a', b, c, d and e. The nodes a, b, c and d form a rectangle with lengths $ab=cd=24$, $ad=bc=10$ and $ac=db=26$. Node e is located midway between the short sides and a little closer to side cd than to side ab, thus $ed=ec=12.5$, and $ea=eb=14.5$. The node a' is at some very small distance from node a, so its distances from other nodes are the same as for a. We remark that the circuit a, d, c, b, e, a', a is a local optimum but is not globally optimal since its cost is three more than the circuit a, d, e, c, b, a', a. The latter circuit is globally and, of course, locally optimal. Now construct n copies of this graph, setting all distances between nodes in different copies to 100, except that the a and a' nodes are at a distance of 20. Any circuit that starts at some a, goes around that copy with either of the two locally optimal circuits discussed above (leaving out a, a'), proceeds to another copy and goes around it with one of the two locally optimal circuits, etc., will be a local optimum in the n-copy graph. For any order of the copies, only one of these 2^n circuits will be globally optimal (the one that always used the second choice). Moreover there are $(n-1)!$ different ways to arrange the copies. We have constructed a graph with $6n$ nodes which has at least $(n-1)!(2^n-1)$ local optima that are not global optima. □

A similar result is given by Papadimitriou and Steiglitz [1978]. They generate instances with exponentially many local optima with respect to a larger neighborhood. The construction in Proposition 3.3, on the other hand, uses fewer nodes and can easily be modified (by placing the a nodes at the vertices of a regular n-gon) to give planar Euclidean instances with $2^n - 1$ local optima that fail to be global optima.

We can further sharpen the apparent distinction between LG and NP-complete problems by showing that LG problems are essentially in NP$\cap$co(NP) and hence unlikely to be NP-complete. To be precise, we define the set recognition version of the optimization problem,

$$\max_{x \in X} f(x),$$

to be the following question: Given an instance and a number k, does there exist an $x \in X$ such that $f(x)$ is at least k?

THEOREM 3.4. *Suppose that, for some discrete optimization problem,*

$$\max_{x \in X} f(x),$$

there exists a notion of adjacency which assigns neighbors to each point in such a way that: (1) *the assignment of neighbors is independent of the instance of the problem* (*independent of the particular data*); *and* (2) *each vertex has polynomially many neighbors. Then if the problem is* LG, *its set recognition version is in* NP$\cap$co(NP).

Proof. If, given some particular data and a number k, there is no $x \in X$ such that $f(x)$ is at least k, this fact can be proved in nondeterministic polynomial time by "guessing" the true optimum, showing that its value is less than k, and verifying its optimality by comparing its value with the values of its (polynomially many) neighbors. □

A definition of adjacency is obviously not worthwhile if it does not satisfy the two requirements of Theorem 3.4, data independence and polynomially many neighbors. We call an adjacency scheme *reasonable* if it satisfies these requirements. Thus the next theorem is in a sense the broadest possible.

THEOREM 3.5. *The clique problem is not* LG *under any reasonable assignment of adjacency. Also, under the ordinary notion of adjacency* (*two subsets S and T of vertices are adjacent if one is a subset of the other and their cardinality differs by one*), *there exists a class of instances with exponentially many local optima that are not global optima.*

Proof. We play the adversary against an arbitrary fixed adjacency rule. The instance we construct will have n nodes, though we will not specify what size n is until later. Our target clique consists of the first $n/4$ nodes. It will be locally but not globally maximal. We connect all of these $n/4$ nodes with edges so that they form a clique, and we do not make any more edges incident to these nodes. Consider the next $n/2$ nodes: there are $\binom{n/2}{n/4}$ subsets of order $n/4$ and $\binom{n/2}{1+n/4}$ subsets of order $(1+n/4)$. By assumption, there exists a polynomial $p(n)$ which bounds the number of neighbors a subset can have. We choose n to be large enough that $np(n)$ is smaller than $\binom{n/2}{n/4}$. Then there must be a subset of the $n/2$ nodes with the properties that: (i) it is of order $(1+n/4)$; (ii) it is not a neighbor of the target clique; and (iii) it contains no subset of order $n/4$ that is a neighbor of the target clique. We connect the nodes of this subset so as to make it a clique; all pairs of nodes not in the subset and not in the target clique remain unconnected. The subset is therefore the global maximum, but any neighbor of the target clique will not be a clique or will be of order less than $n/4$. Given an arbitrary polynomial adjacency rule, we have constructed an instance that is not LG.

The ordinary notion of adjacency states that two subsets of the nodes of the graph are adjacent if one contains the other and their cardinality differs by one. We start with a complete graph on n nodes, where the nodes are labelled $1, 2, \cdots, n$. Next we delete all of the edges which are of the form $(i, i+1)$. If a subset of the nodes contains i, it cannot contain either $i+1$ or $i-1$ and still be a clique. We can build up cliques $(v_1, \cdots, v_k, \cdots)$ by choosing v_1 equal to either node 1 or 2, and v_{k+1} equal to either v_k+2 or v_k+3. Since the subsets $(k, k+1, k+3)$, $(k, k+1, k+2)$ and $(k, k+2, k+3)$ do not have all of their edges, the cliques we build in this way are all locally optimal. There are more than $2^{n/3}$ of these because we make at least $n/3$ choices when constructing them. Moreover, most of them (all but $2n$, at least) are of order less than $n/2$, the maximal order achieved by the clique $(1, 3, 5, 7, \cdots)$. We therefore have exponentially many local optima that fail to be global optima. □

Theorem 3.5 is proved directly by constructing instances with the desired properties. A different method of proof may be employed to produce a stronger result in some cases.

THEOREM 3.6. *In the clique problem, for any data-independent adjacency in which each element has n or fewer neighbors, there exists a class of instances with exponentially many local optima.*

Proof. We prove the theorem nonconstructively by employing the probabilistic method of Erdös and Spencer [1974]. For n even, construct a graph of order n at random by including each edge with probability p, $0 < p < 1$, p as yet unspecified. Let S be any subset of the vertices with $|S| = n/2$, and let S_k, $k = 1, \cdots, n$ denote the n subsets adjacent to S. (It suffices to prove the theorem when each subset has n neighbors.) For any subset T, let I_T equal 1 if the subgraph of the vertices in T is complete, and 0 otherwise. The probability that S is locally optimal is

$$\begin{aligned} &\text{Prob}[I_S = 1] \cdot \text{Prob}[I_{S_k} = 0 \forall k \text{ s.t. } |S_k| > |S| \,|\, I_S = 1] \\ &= p^{\binom{n/2}{2}} \cdot \text{Prob}[I_{S_k} = 0 \forall k \text{ s.t. } |S_k| > |S| \,|\, I_S = 1] \qquad (3.6.1) \\ &\geqq p^{\binom{n/2}{2}} \prod_{k=1}^{n} (1 - p^{e_k}), \end{aligned}$$

where

$$e_k = \begin{cases} \#\text{ edges in } S_k \text{ not in } S & \text{if } |S_k| > |S|, \\ \infty & \text{if } |S_k| \leqq S. \end{cases}$$

Note. The inequality above derives from the possible positive correlation among the conditional events if S_j and S_k share an edge not in S.

PROPOSITION I. $e_k \geqq n/2$ *for all* k.

PROPOSITION II. $e_k = n/2$ *for at most* $n/2$ *values of* k.

PROPOSITION III. *If* $e_k > n/2$ *then* $e_k \geqq n-1$.

Proof. Proposition II, $S_k = S \cup v$ for some vertex $v \notin S$. In Proposition III, $S_k = \{S \cup v \cup w\} - z$, where $z \in S$; $v, w \notin S$. □

It follows from these propositions that (3.6.1) is greater than or equal to

$$p^{\binom{n/2}{2}}(1-p^{n/2})^{n/2}(1-p^{n-1})^{n/2} \geqq r^{n/4}(1-r)^{n/2}(1-r^2)^{n/2}, \qquad (3.6.2)$$

where $r = p^{n-1/2}$, so $r < p^{n/2-1}$ and $1-r < 1-p^{n/2}$. By elementary calculus, (3.6.2) is maximized when r is close to $\frac{1}{4}$. Therefore, choose p so that $r = \frac{1}{4}$. Then the probability that S is a local optimum is at least

$$(\tfrac{1}{4})^{n/4}(\tfrac{3}{4})^{n/2}(\tfrac{15}{16})^{n/2} = (\tfrac{45}{128})^{n/2},$$

and so the expected number of local optima of size $n/2$ is greater than

$$\binom{n}{n/2}\left(\frac{45}{128}\right)^{n/2} > (1.05)^n \quad \forall n \geqq 12.$$

Note that

$$\left.\binom{n+2}{\frac{n+2}{2}}\right/\binom{n}{n/2} = \frac{1}{4}\left(\frac{n+1}{n+2}\right) \to \frac{1}{4} \quad \text{as } n \to \infty,$$

so the bound on the expectation grows asymptotically as $4^{n/2}(\frac{45}{128})^{n/2} \approx (1.4)^{n/2}$.

Since the expected number is exponentially large, there must exist instances in which the number is large. □

Note that the proof, particularly the propositions and the ensuing inequalities, suggests that the usual notion of adjacency is a rather good one from the standpoint of minimizing the expected number of local optima. Also note that the local improvement algorithm for cliques could be extended to allow movement from S to S', where S and S' are not cliques, and the proportion (or number) of "missing" edges in S' is smaller than in S without altering the result.

Many known NP-complete problems, such as knapsack or three-dimensional matching, originated in the form of optimization problems, so the idea of local optimality applies immediately. Some other problems, such as satisfiability, 3-colorability or 2-partition, are ordinarily set as recognition version (that is, yes/no) problems, so the concept of local optimality may not seem to apply. However, we have found that most such problems can be easily transformed into an optimization version. For example, the Boolean satisfiability problem becomes the problem of assigning Boolean values to the set of variables so as to maximize the total number of clauses that are true. The 3-colorability problem becomes the problem of assigning one of three colors to each node in the graph so as to minimize the number of pairs of nodes that have the same color and are connected by an edge. With 2-partition, we try to minimze the difference between the sums of the two subsets; with subgraph isomorphism (given two graphs G and H, does H contain a subgraph isomorphic to G?) we try to find a mapping from the nodes of G into the nodes of H that minimizes the number of conflicts in the corresponding edge sets.

Using this notion of optimization versions of NP-complete problems, we can now discuss local and global optimality. We believe that all NP-complete problems have the property of exponentially many local optima. However, since this statement implies that P $\neq$ NP, a proof will not be attempted. (If P = NP, then any LG problem in P, such as linear programming, would be NP-complete.) We do remark, however, that many of the polynomial transformations used in NP-completeness results preserve the "topology" of adjacencies in such a way that local optima remain local optima. It is usually easy to show that some particular NP-complete problem is not LG, (at least) with respect to the natural notion of adjacency.

To transform the clique problem on a graph $G = (V, E)$ to a "Boolean maximization" problem, let $X_i = 1$ or 0 as the ith vertex is or is not in the subset, $i = 1, \cdots, |V|$. Now maximize the number of true clauses in

$$\{X_1\} \wedge \{X_2\} \wedge \cdots \wedge \{X_n\} \wedge [n+1 \text{ identical clauses } \{\bar{X}_i \vee \bar{X}_j\} \ \forall i, j \text{ s.t. } (i,j) \notin E].$$

Clearly two properties hold for all subgraphs S, S':

$f(S) > f(S')$ if S is a clique and S' is not.

$f(S) > f(S')$ if $|S| > |S'|$ and both S and S' are cliques.

Hence any local optimum in Theorems 3.4 and 3.5 is a local optimum here (there may indeed be more). The reader may notice that all clauses in the example above have one or two variables. All is in order, however, since although 2-SAT is in P, 2-SAT maximization is NP-hard (Garey et al. [1975]).

We thus have the following corollaries to Theorems 3.5 and 3.6.

COROLLARY 3.7. *The optimization version of the Boolean satisfiability problem is not* LG *under any reasonable adjacency scheme.*

COROLLARY 3.8. *For any data independent adjacency rule which assigns n neighbors to each point, there exist instances of the Boolean satisfiability problem with exponentially many local optima.*

Transformations to integer programming, 3DM (three-dimensional matching) and other problems are not difficult.

4. Conclusions. Local improvement algorithms have been shown to be quite fast on the average. On the other hand, the number of local optima can be quite large, especially when the problem is NP-hard. This indicates that the probability of finding the global optimum may be poor. Note that this probability is not necessarily the reciprocal of the number of local optima: there may be many trees in the forest, but if the tree whose root is the global optimum is large (i.e. contains more than the average share of vertices), the probability of reaching the global optimum is better. However, determining this probability appears to be difficult.

A very effective way to improve the probability of finding the global optimum is to find a better than random starting point or points. Hillier [1969] reports considerable success in solving integer programming problems by using heuristics that identify promising initial solutions, followed by local improvement. Our results show that in general local improvement should be an inexpensive way to improve "good" solutions.

Acknowledgments. This work is based on and extends my thesis on local improvement algorithms. I would like to thank my advisor, George B. Dantzig, for his invaluable guidance and encouragement. The proofs of Theorems 2.1 and 2.2 come from an idea by Jeffrey Ullman. I thank him and the other member of my reading committee, Richard Cottle, for their help and comments.

REFERENCES

GEORGE DANTZIG [1963], *Linear Programming and Extensions*, Princeton Univ. Press, Princeton, NJ.

PAUL ERDÖS AND JOEL SPENCER [1974], *Probabilistic Methods in Combinatorics*, Academic Press, New York.

M. R. GAREY, D. S. JOHNSON AND L. STOCKMEYER [1975], *Some simplified* NP*-complete graph problems*, Theoret. Comput. Sci., 1, pp. 237–267.

F. S. HILLIER [1969], *Efficient heuristic procedures for integer linear programming with an interior*, Oper. Res., 17, pp. 600–637.

DONALD KNUTH [1973], *The Art of Computer Programming, Vol.* I, Addison-Wesley, Reading, MA.

NILS NILSSON [1980], *Principles of Artificial Intelligence*, Tioga Publishing, Palo Alto, CA.

C. H. PAPADIMITRIOU AND K. STEIGLITZ [1978], *Some examples of difficult traveling salesman problems*, Oper. Res., 26, pp. 434–443.

CRAIG TOVEY [1981], *Polynomial local improvement algorithms in combinatorial optimization*, Systems Optimization Laboratory Technical Report SOL 81-21, Dept. Operations Research, Stanford Univ., Stanford, CA.

——— [1982], *Low order bounds on the expected number of iterations of local improvement algorithms*, Math. Programming, to appear, Study on Expected Performance of the Simplex and Related Methods.

——— [1983], *On the number of iterations of local improvement algorithms*, Oper. Res. Letters, 2, No. 5.

PATRICK WINSTON [1977], *Artificial Intelligence*, Addison-Wesley, Reading, MA.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 3, July 1985

007

ASYMPTOTIC NORMALITY IN THE GENERALIZED POLYA–EGGENBERGER URN MODEL, WITH AN APPLICATION TO COMPUTER DATA STRUCTURES*

A. BAGCHI† AND A. K. PAL†

Abstract. In the generalized Polya–Eggenberger urn model, an urn initially contains a given number of white and black balls. A ball is selected at random from the urn, and the number of white and black balls added to (or taken away from) the urn depends on the color of the ball selected. Let w_n be the random variable giving the number of white balls in the urn after n draws. A sufficient condition is derived for the asymptotic normality, as $n \to \infty$, of the standardized random variable corresponding to w_n. This result is then used for estimating the computer memory requirements of the 2-3 tree, a well-known computer data structure for storage organization.

Key words. 2-3 tree, random insertion, method of moments, martingales

AMS (MOS) subject classifications. 60F05, 62E20, 68E99

1. Statement of problem. The generalized Polya–Eggenberger urn scheme has been widely studied and has many applications [11]. In this model, an urn initially contains a total of t_0 balls, of which w_0 are white and $t_0 - w_0$ black. A ball is selected at random from the urn, its color is noted, and it is put back in the urn. If the color of the chosen ball is white, then a white balls and b black balls are added to the urn; if its color is black, then c white balls and d black balls are added to the urn (Table 1). A negative value of a, b, c or d indicates that that many balls are thrown away from the urn instead of being added. The random variable of interest is w_n, the number of white balls in the urn after n draws.

TABLE 1

		Number of balls added to the urn	
		white	black
Color of chosen ball	white	a	b
	black	c	d

The probability distribution of w_n is known for some special cases (see [11, §§ 4.3 and 6.3]). In this paper we derive a sufficient condition for the asymptotic normality (as $n \to \infty$) of the distribution of the standardized random variable corresponding to w_n. We then apply the result to estimate the computer storage requirements of the 2-3 tree, a well-known data structure for organizing information in computers [1]. We make the following assumptions about the parameters a, b, c, d, t_0, and w_0:

i) $a + b = c + d = s \geqq 1$.
ii) $t_0 \geqq 1$, and $0 \leqq w_0 \leqq t_0$.
iii) $a \neq c$.
iv) $b > 0$ and $c > 0$.
v) If $a < 0$ then a divides c and a divides w_0. Similarly, if $d < 0$, then d divides b and d divides $t_0 - w_0$.

* Received by the editors September 29, 1982, and in revised form March 1, 1984.
† Indian Institute of Management Calcutta, P.O. Box 16757, Calcutta - 700 027, India.

Let us say that a generalized Polya-Eggenberger urn model is *tenable* if conditions (i) through (v) hold. Then the basic result of the paper is as follows:

Given a tenable urn model, the distribution of the standardized random variable corresponding to w_n as $n \to \infty$ converges asymptotically to the standard normal distribution whenever $a - c \leqq s/2$.

The proof uses the method of moments. For an example of a tenable urn model, suppose $t_0 = w_0 = 2$, and

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} = \begin{bmatrix} -2 & 3 \\ 4 & -3 \end{bmatrix}.$$

This is the model for random insertions in 2-3 trees, described in § 3.

Assumptions (iii)-(v) are less restrictive than they appear. For instance, (iii) simply rules out a degenerate case. As for (iv), it does not make sense to allow b and c to take on negative values. To see this, suppose $b < 0$, and that the urn contains a white ball. Since $a = s - b > 1$, it is possible that only white balls are drawn subsequently, and after a while there may not be any more black balls left in the urn to throw away, i.e., the model can get "stuck." A parallel situation arises if $c < 0$. When $b = c = 0$, we get the simple Polya-Eggenberger model, for which the distribution of w_n is known [11]. The only case which has not been studied here and which requires further investigation is when

$$\min(b, c) = 0 < \max(b, c).$$

This case presents some curious technical problems and appears to need a separate treatment. Coming to assumption (v), note that if (v) does not hold, it can happen that a white ball is drawn from the urn, but the urn contains fewer than a white balls, so that it is not possible to throw away a white balls as required.

We briefly review some related work. Bernard Friedman [8] looked at the special case that arises when $a = d \geqq 0$ and $b = c \geqq 0$. He derived the differential-difference equation for the characteristic function of w_n, and obtained integral representations for the probabilities of getting a white ball on the nth draw and of having k white balls in the urn after n draws. D. A. Freedman [7] extended Bernard Friedman's analysis and used the method of moments to show, under certain conditions, the asymptotic normality of w_n as $n \to \infty$. More general urn models have been proposed by Athreya and Karlin [2], but they did not allow negative values smaller than -1 of a and d.

2. Main results. Since the method of moments is being used, it is necessary to begin by getting expressions for the expected value and the standard deviation of w_n, and then to compute the higher order central moments. Most of the time only limiting values are of interest, so we use asymptotic methods extensively in the proofs.

Let $q = a - c$. As $b + c > 0$, we have $q < s$. For $n \geqq 0$, let t_n be the total number of balls and w_n the number of white balls in the urn after n draws. Then

$$t_n = t_0 + ns \tag{1}$$

and

$$P[w_{n+1} = w_n + a | w_n] = \frac{w_n}{t_n}, \qquad P[w_{n+1} = w_n + c | w_n] = 1 - \frac{w_n}{t_n}. \tag{2}$$

Thus the sequence $\{w_n\}$ is a Markov process.

LEMMA 1. $E(w_n) \sim ct_n/(b+c)$ *as* $n \to \infty$.

Proof. From (2),

$$E(w_{n+1}) = \left(1+\frac{q}{t_n}\right)E(w_n)+c, \qquad n \geqq 0.$$

Making the substitution

$$y_n = w_n - \frac{c}{b+c}\,t_n, \qquad n \geqq 0 \tag{3}$$

we get

$$E(y_{n+1}) = \left(1+\frac{q}{t_n}\right)E(y_n), \qquad n \geqq 0$$

which has the solution

$$E(y_n) = \left(w_0 - \frac{c}{b+c}\,t_0\right)B(1, n), \qquad n \geqq 0,$$

where

$$B(r, 0) = 1, \qquad B(r, n) = \prod_{j=0}^{n-1}\left(1+\frac{rq}{t_j}\right), \quad n \geqq 1$$

for integer $r \geqq 1$. By Claim 1 of the Appendix $E(y_n) = O(t_n^{q/s})$ as $n \to \infty$ and since $q < s$, we conclude that $E(w_n) \sim ct_n/(b+c)$ as $n \to \infty$. □

LEMMA 2. *When* $n \to \infty$,

i) $$\sigma(w_n) \sim \left(\frac{bc}{s-2q}\right)^{1/2} \cdot \frac{|q|}{b+c} \cdot t_n^{1/2} \quad \text{if } q < \frac{s}{2}, \quad q \neq 0.$$

ii) $$\sigma(w_n) \sim \left(\frac{bc}{s}\right)^{1/2} (t_n \ln t_n)^{1/2} \quad \text{if } q = \frac{s}{2}.$$

Proof. Using the notation of Lemma 1, by (2) and (3), for $n \geqq 0$,

$$E(y_{n+1}^2 | y_n) = \left(y_n + a - \frac{cs}{b+c}\right)^2\left(\frac{c}{b+c}+\frac{y_n}{t_n}\right) + \left(y_n + c - \frac{cs}{b+c}\right)^2\left(\frac{b}{b+c}-\frac{y_n}{t_n}\right).$$

This yields the recurrence

$$E(y_{n+1}^2) = \left(1+\frac{2q}{t_n}\right)E(y_n^2)+\frac{b-c}{b+c}\,\frac{q^2}{t_n}\,E(y_n)+\frac{bcq^2}{(b+c)^2}, \qquad n \geqq 0. \tag{4}$$

i) First suppose $q < s/2$, $q \neq 0$. The homogeneous equation corresponding to (4) is

$$E(y_{n+1}^2) - \left(1+\frac{2q}{t_n}\right)E(y_n^2) = 0,$$

which has a solution

$$Q(2, n) = B(2, n), \qquad n \geqq 0.$$

A particular solution of (4) is

$$R(2, n) = \frac{bc}{s-2q}\left(\frac{q}{b+c}\right)^2 t_n - \left(\frac{b-c}{b+c}\right)qE(y_n), \qquad n \geqq 0,$$

so the complete solution of (4) can be written in the form [12]

$$E(y_n^2) = kQ(2, n) + R(2, n),$$

where k is a constant which can be determined with the help of the initial condition

$$E(y_0^2) = y_0^2 = \left(w_0 - \frac{ct_0}{b+c}\right)^2.$$

The variance of w_n is

$$V(w_n) = E[w_n - E(w_n)]^2 = E[y_n - E(y_n)]^2 = E(y_n^2) - [E(y_n)]^2,$$

so by Lemma 1 and Claim 1 of the Appendix, as $n \to \infty$,

$$V(w_n) = \frac{bc}{s-2q}\left(\frac{q}{b+c}\right)^2 t_n + o(t_n).$$

ii) Now suppose $q = s/2$. Then (4) simplifies to

$$E(y_{n+1}^2) = \frac{t_{n+1}}{t_n} E(y_n^2) + \frac{b^2 - c^2}{t_n} E(y_n) + bc, \qquad n \geqq 0.$$

The homogeneous equation

$$E(y_{n+1}^2) - \frac{t_{n+1}}{t_n} E(y_n^2) = 0$$

has a solution of the form $Q(2, n) = t_n$. To get a particular solution $R(2, n)$ we try the substitution $R(2, n) = t_n g(n) + (c-b)E(y_n)$, $n \geqq 0$ and get the new recurrence

$$t_{n+1} g(n+1) = t_{n+1} g(n) + bc, \qquad n \geqq 0,$$

where we have assumed

$$g(0) = \frac{y_0^2 + (b-c)y_0}{t_0},$$

so that

$$g(n) = g(0) + bc \sum_{j=1}^{n} \frac{1}{t_j}, \qquad n \geqq 1.$$

Thus for $n \geqq 1$,

$$E(y_n^2) = kt_n + (c-b)E(y_n) + t_n g(0) + t_n bc \sum_{j=1}^{n} \frac{1}{t_j},$$

where k is a constant to be determined by the initial conditions. But as $n \to \infty$,

$$\sum_{j=1}^{n} \frac{1}{t_j} \sim \frac{1}{s} \ln t_n.$$

So we can conclude that as $n \to \infty$.

$$E(y_n^2) \sim \frac{bc}{s} t_n \ln t_n,$$

which gives

$$V(w_n) \sim \frac{bc}{s} t_n \ln t_n. \qquad \square$$

COROLLARY 1. *If* $q \leqq s/2$ *then*

i) $\sigma\left(\frac{w_n}{t_n}\right) \to 0$ *as* $n \to \infty$,

ii) $\frac{w_n}{t_n} \xrightarrow{\text{a.s.}} \frac{c}{b+c}$.

Proof. i) By Lemma 2.

ii) By i) and Lemma 1,

$$\frac{w_n}{t_n} \xrightarrow{p} \frac{c}{b+c}.$$

Since the probability space here is countable,

$$\frac{w_n}{t_n} \xrightarrow{\text{a.s.}} \frac{c}{b+c}.$$

(See Chow and Teicher [5, Ex. 2, p. 43].)

Our problem now is to see if the standardized random variable

$$z_n = \frac{w_n - E(w_n)}{\sigma(w_n)} = \frac{y_n - E(y_n)}{\sigma(y_n)}$$

has an asymptotically normal distribution when $q \leqq s/2$. We base our approach on the method of moments. The idea is to determine the higher order raw moments of y_n, with the help of which the asymptotic values of the higher order central moments of w_n, and hence of z_n, can be computed. These can then be shown to converge to the moments of the standard normal variate.

By (2) and (3), for $r \geqq 1$,

$$E(y_{n+1}^r | y_n) = \left(y_n + a - \frac{cs}{b+c}\right)^r \left(\frac{c}{b+c} + \frac{y_n}{t_n}\right) + \left(y_n + c - \frac{cs}{b+c}\right)^r \left(\frac{b}{b+c} - \frac{y_n}{t_n}\right), \qquad n \geqq 0$$

so that for $r \geqq 1$ and $n \geqq 0$

$$E(y_{n+1}^r) - \left(1 + \frac{rq}{t_n}\right) E(y_n^r) = \sum_{j=1}^{r} \left(p_{r,r-j} + \frac{q_{r,r-j}}{t_n}\right) E(y_n^{r-j}), \tag{5}$$

where

$$p_{r,r-j} = \binom{r}{j} \left(\frac{q}{b+c}\right)^j \frac{bc}{b+c} [b^{j-1} + (-1)^j c^{j-1}],$$

$$q_{r,r-j} = \binom{r}{j+1} \left(\frac{q}{b+c}\right)^{j+1} [b^{j+1} + (-1)^j c^{j+1}],$$

$$\binom{j}{j+1} = 0 \quad \text{for } j \geqq 1.$$

It is to be noted that $p_{r,r-1} = 0$, $q_{r,0} = 0$, and $E(y_n^0) = 1$. The initial conditions for the recurrences are

$$E(y_0^r) = y_0^r = \left(w_0 - \frac{ct_0}{b+c}\right)^r, \qquad r \geqq 1.$$

LEMMA 3. *Let* $q < s/2$, $q \neq 0$. *Then as* $n \to \infty$;

i) *For even* $r \geqq 2$,

$$E(y_n^r) = e_r t_n^{r/2} + o(t_n^{r/2})$$

where $e_r = 1 \cdot 3 \cdot 5 \cdot \cdots \cdot (r-1) e_2^{r/2}$ *and* $e_2 = bc(q/(b+c))^2/(s-2q)$.

ii) *For odd* $r \geqq 1$,

$$E(y_n^r) = o(t_n^{r/2}).$$

Proof. For $r \geqq 1$ and $n \geqq 0$, the homogeneous recurrence corresponding to (5) is

$$E(y_{n+1}^r) - \left(1 + \frac{rq}{t_n}\right) E(y_n^r) = 0,$$

which has a solution of the form

$$Q(r, n) = B(r, n), \qquad n \geqq 1.$$

For $q < s/2$, by Claim 1 of the Appendix, as $n \to \infty$

$$B(r, n) = o(t_n^{r/2})$$

so to prove the lemma it is enough to look at a particular solution $R(r, n)$ of (5). By Lemmas 1 and 2, Lemma 3 holds for $r = 1, 2$. So we assume inductively that for $1 \leqq j < r$, as $n \to \infty$,

$$R(j, n) = \begin{cases} e_j t_n^{j/2} + o(t_n^{j/2}) & \text{for } j \text{ even}, \\ o(t_n^{j/2}) & \text{for } j \text{ odd}. \end{cases}$$

Let us suppose r is even, and make the substitution $R(r, n) = f(r, n) + e_r t_n^{r/2}$ in (5). Then

$$e_r(t_n + s)^{r/2} + f(r, n+1) - \left(1 + \frac{rq}{t_n}\right)[e_r t_n^{r/2} + f(r, n)]$$

$$= \frac{q_{r,r-1}}{t_n} \cdot o(t_n^{(r-1)/2}) + \left(p_{r,r-2} + \frac{q_{r,r-2}}{t_n}\right) \cdot [e_{r-2} t_n^{r/2-1} + o(t_n^{r/2-1})] + o(t_n^{(r-3)/2});$$

since $e_r = (r-1) e_2 \cdot e_{r-2}$, we simplify and get

$$f(r, n+1) - \left(1 + \frac{rq}{t_n}\right) f(r, n) = o(t_n^{r/2-1}).$$

Then

$$f(r, n) = B(r, n)\left[f(r, k) + \sum_{j=k}^{n-1} \frac{o(t_j^{r/2-1})}{B(r, j+1)}\right]$$

where $k = 0$ if $rq \neq -t_m$ for all $m \geqq 0$, and $k = m+1$ if $rq = -t_m$ for some $m \geqq 0$. (See, for example, Jordan [12, p. 583].) So by Claim 1, for $q < s/2$,

$$f(r, n) = o(t_n^{r/2}).$$

When r is odd we make $f(r, n) = R(r, n)$ and proceed similarly. □

LEMMA 4. *Let* $q < s/2$, $q \neq 0$. *Then as* $n \to \infty$;

i) *For even* $r \geqq 2$,

$$E[w_n - E(w_n)]^r = e_r t_n^{r/2} + o(t_n^{r/2})$$

where e_r *is defined as in Lemma* 3.

ii) *For odd* $r \geqq 3$,

$$E[w_n - E(w_n)]^r = o(t_n^{r/2}).$$

Proof. Clearly, for $n \geqq 0$ and $r \geqq 2$,

$$E[w_n - E(w_n)]^r = E(y_n - E(y_n))^r$$

$$= E(y_n^r) + \sum_{j=1}^{r} (-1)^j \binom{r}{j} E(y_n^{r-j})[E(y_n)]^j.$$

Now use Lemma 3. □

THEOREM 1. *Let* $q < s/2$, $q \neq 0$. *Then as* $n \to \infty$,

$$P\left\{ z_n = \frac{w_n - E(w_n)}{\sigma(w_n)} < x \right\} \to \Phi(x),$$

where Φ *is the distribution function of the standard normal variate.*

Proof. By Lemmas 2 and 4, as $n \to \infty$,

$$E(z_n^r) \to \begin{cases} 1 \cdot 3 \cdot 5 \cdot \cdots \cdot (r-1), & r \text{ even}, \\ 0, & r \text{ odd}. \end{cases}$$

But this uniquely characterizes the standard normal variate (see Fisz [6]). □

The special case $s = 1$ is of great interest, and we make the following observation:

COROLLARY 2. *Given a tenable urn model with* $s = 1$, *the distribution of* z_n *as* $n \to \infty$ *coincides asymptotically with the standard normal distribution.*

Proof. We observe that for a tenable urn model with $s = 1$, it must be the case that $q < s/2$. □

We now come to the case $q = s/2$.

LEMMA 5. *Let* $q = s/2$. *Then as* $n \to \infty$.

i) *For even* $r \geqq 2$,

$$E(y_n^r) \sim e_r'(t_n \ln t_n)^{r/2}$$

where

$$e_r' = 1 \cdot 3 \cdot 5 \cdot \cdots \cdot (r-1)(e_2')^{r/2} \quad \textit{and} \quad e_2' = bc/s.$$

ii) *For odd* $r \geqq 1$,

$$E(y_n^r) = o(t_n \ln t_n)^{r/2}.$$

Proof. We proceed as in the proof of Lemma 3. This time the solution $Q(r, n)$ to the homogeneous recurrence has the form

$$Q(r, n) = B(r, n) = O(t_n^{r/2})$$

for $r \geqq 1$ as $n \to \infty$. So it is once again enough to look at a particular solution $R(r, n)$. By Lemmas 1 and 2, Lemma 5 holds for $r = 1, 2$. So we assume inductively that the lemma holds for $1 \leqq j < r$, and supposing r even, make the subsitution

$$R(r, n) = f(r, n) + e_r'\left(t_n \ln \frac{t_n}{t_0}\right)^{r/2}.$$

Substituting in (5) and simplifying, we get

$$f(r, n+1) - \left(1 + \frac{rq}{t_n}\right) f(r, n) = o\left(t_n \ln \frac{t_n}{t_0}\right)^{r/2-1}$$

which implies

$$f(r, n) = o\left(t_n \ln \frac{t_n}{t_0}\right)^{r/2}$$

as required. A similar procedure can be followed when r is odd. □

THEOREM 2. *Let $q = s/2$. Then as $n \to \infty$*

$$P\left\{z_n = \frac{w_n - E(w_n)}{\sigma(W_n)} < x\right\} \to \Phi(x),$$

where Φ is the distribution function of the standard normal variate.

Proof. Use Lemmas 2 and 5. □

An open problem is to determine the asymptotic distribution of z_n when $s/2 < q < s$.

We now take a look at some alternative approaches to the problem. Note that for $q < s$, $\{w_n\}$ is essentially a martingale. Let

$$w'_n = \alpha_n w_n + \beta_n, \quad n \geqq 0$$

where α_n, β_n are real numbers. Then

$$E(w'_{n+1} | w'_0, w'_1, \cdots, w'_n) = E(w'_{n+1} | w'_n) = w'_n$$

provided $\alpha_{n+1}/\alpha_n = t_n/(t_n + q)$ and $\beta_{n+1} - \beta_n = -c\alpha_{n+1}$. So we can choose

$$\alpha_n = \frac{\Gamma(t_0/s + n)}{\Gamma((t_0 + q)/s + n)} \sim n^{-q/s}.$$

Then

$$\beta_n = -c \sum_{j=1}^{n} \alpha_j + \beta_0 \sim \frac{-cs}{b + c} n^{1 - q/s}.$$

Moreover,

$$U_n = E[(w'_{n+1} - w'_n)^2 | w'_n] = \frac{w_n}{t_n}\left(1 - \frac{w_n}{t_n}\right)(\alpha_{n+1} q)^2 \leqq \frac{1}{4}(\alpha_{n+1} q)^2.$$

When $q \leqq s/2$, we can now show, by an application of the Hajek-Renyi inequality (see [5, p. 243]), that $\lim_{n \to \infty}(w'_n/\alpha_n t_n) = 0$, which implies

$$\frac{w_n}{t_n} \xrightarrow{\text{a.s.}} \frac{c}{b + c}.$$

On the other hand, when $q > s/2$, $\sum_{n=1}^{\infty} U_n < \infty$. Hence by [9, p. 33], $\lim_{n \to \infty} w'_n$ exists and is finite. Thus in

$$\frac{w_n}{t_n} = \frac{-\beta_n}{\alpha_n t_n} + \frac{w'_n}{\alpha_n t_n}$$

the first term dominates, so we get

$$\frac{w_n}{t_n} \xrightarrow{\text{a.s.}} \frac{c}{b + c}$$

more easily. It is an open question whether the asymptotic normality of w_n can be proved using martingale theory.

Athreya and Karlin [2] (see also Athreya and Ney [3, pp. 219-224]) have suggested a natural embedding of Polya's urn model in a branching process. In their most general

scheme, there are m colors. If color i is drawn (and replaced), then we add R_{ij} balls of color j, $1 \leqq j \leqq m$, where $(R_{i1}, R_{i2}, \cdots, R_{im})$ is a random vector with a preassigned distribution depending on i. While $R_{ii} = -1$ is allowed, other negative values are not allowed. It may be possible to extend their scheme to allow several simultaneous deaths. Multivariate extensions of Theorem 1 of this paper could have interesting applications to computer data structures like B-trees. Recently, Holst [10] has given a unified treatment of various urn problems, but he has not considered the generalized Polya-Eggenberger model.

3. 2-3 trees. A 2-3 *tree* T is a rooted, oriented tree in which each internal node has 2 or 3 sons, and every path from the root to a leaf has the same length. A *key* is associated with each leaf (see [1, pp. 148–152]). The keys, which are positive integers and are all distinct, form an ascending sequence viewed from left to right in T. An internal node in T is *W-type* if it has 2 sons and *B-type* if it has 3 sons. A key is *white* if it is a son of a W-type node; otherwise it is *black*.

Fig. 1(a) shows a 2-3 tree having the keys 10, 30, 50, 60, 70, 80, 90. The internal nodes B_0 and D_0 are W-type, while A_0(root) and C_0 are B-type. The keys 10, 30, 80 and 90 are white, while 50, 60, 70 are black. Suppose we now want to insert a new key with value 20 into the tree. Clearly, 20 must be put in between 10 and 30, and we

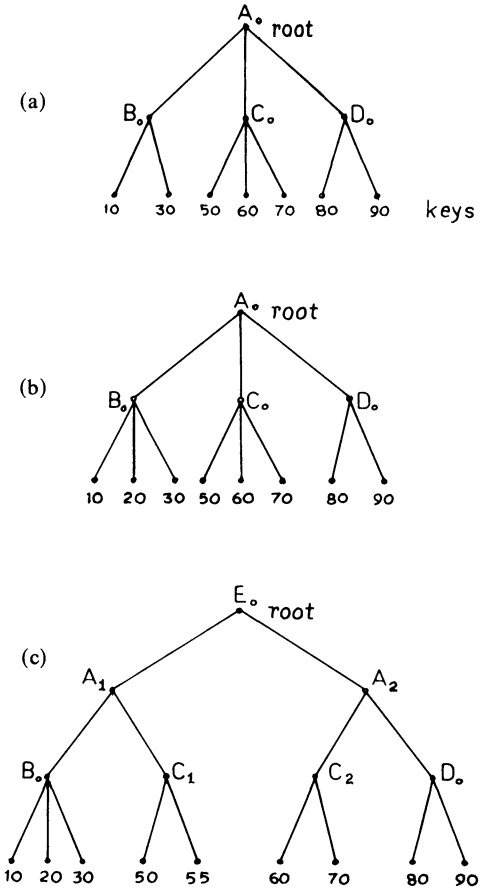


FIG. 1

would get the tree shown in Fig. 1(b). B_0 has now become B-type, and the only white keys are 80 and 90. If we now insert another key with value 55 into the tree, then C_0 would have 4 sons, which is not permissible. So C_0 is split into two W-type nodes C_1 and C_2. As a consequence, A_0 has 4 sons, and gets split into two W-type nodes A_1 and A_2. The tree has a new root node E_0, as shown in Fig. 1(c).

Talking in general terms, let T be a 2-3 tree with r keys for some $r \geqq 2$. Let K_1, $K_2, \cdots, K_r$ be the keys in T in ascending order of magnitude. We make the assumption that $K_r = \infty$; this takes care of an asymmetry inherent in the 2-3 tree insertion algorithm given in [1]. A new key K is to be inserted into T, where K does not equal in value any of the keys already present in T. We find i, $1 \leqq i \leqq r$, such that $K_{i-1} < K < K_i$ (where $K_0 = 0$). We then attach K to the father of K_i, placing K in the tree between K_{i-1} and K_i. If the father of K_i is a W-type node, then the insertion of K makes it a B-type node; if it is a B-type node, then it gets split into two W-type nodes, and this can cause B-type nodes at higher levels of the tree to split also, as Fig. 1(c) shows. The insertion of K is said to be a *random insertion* if K has equal probability of lying in any one of the r intervals (K_{i-1}, K_i), $1 \leqq i \leqq r$.

The 2-3 tree is a widely used data structure for storage organization in computers, so it is important to estimate its memory requirements. One way to do this is as follows. Let T_0 be a 2-3 tree with $t_0 \geqq 2$ keys of which w_0 are white. We make a random insertion of a key into T_0 to get a 2-3 tree T_1, then we make a random insertion of a key into T_1 to get a tree T_2, and so on. Let t_n be total number of keys and w_n the number of white keys in T_n, the 2-3 tree after n random insertions, where $n \geqq 0$. Then w_n is a random variable, and this whole process of random insertions can be modelled using a generalized Polya-Eggenberger urn scheme, where the matrix

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} = \begin{bmatrix} -2 & 3 \\ 4 & -3 \end{bmatrix}.$$

Here $s = 1$, $q = -6$ and $t_n = t_0 + n$. If the distribution of w_n is known, then it is possible to obtain bounds on the random variable N_n, the number of internal nodes in T_n. The value of N_n gives a fair idea about the memory needs of T_n.

In [15], Yao derived an expression for $E(w_n)$, and with its help obtained bounds on $E(N_n)$. However, bounds on N_n are much more meaningful in practice than bounds on $E(N_n)$. Using our results we are able to determine the limiting distribution of w_n, and so are able to get bounds on N_n for large n with high levels of confidence.

LEMMA 6. *As* $n \to \infty$,

i) $$E(w_n) \sim \frac{4}{7}(n + t_0) \sim \frac{4n}{7}.$$

ii) $$\sigma(w_n) \sim \frac{6}{7}\sqrt{\frac{12}{13}}(n + t_0)^{1/2} \sim \frac{12}{7}\sqrt{\frac{3n}{13}}.$$

Proof. Note that $q = -6$ and use Lemmas 1 and 2. □

THEOREM 3. *Let*

$$L_n = \frac{1}{2}(n + t_0) + \frac{w_n}{4} - \frac{1}{2}, \qquad M_n = \frac{2}{3}(n + t_0) + \frac{w_n}{3} - 1.$$

(i) *For* $n \geqq 0$, $L_n \leqq N_n \leqq M_n$.

(ii) $$E(L_n) \sim \frac{9n}{14}, \quad \sigma(L_n) \sim \frac{3}{7}\sqrt{\frac{3n}{13}}, \quad E(M_n) \sim \frac{6n}{7}, \quad \sigma(M_n) \sim \frac{4}{7}\sqrt{\frac{3n}{13}}.$$

(iii) *As $n \to \infty$, the standardized random variables corresponding to L_n and M_n are asymptotically normal.*

Proof. i) Let N' be the number of internal nodes in T_n at the lowest level. Then

$$N' = \frac{w_n}{2} + \frac{t_n - w_n}{3} = \frac{t_n}{3} + \frac{w_n}{6}.$$

Let $N'' = N_n - N'$. So N'' is the number of internal nodes in T_n at higher levels. The minimum value of N'' is $(N'-1)/2$ when all nodes at higher levels are B-type, while the maximum value of N'' is $N'-1$ when all nodes at higher levels are W-type. Hence

$$N' + \frac{N'-1}{2} \leqq N_n \leqq N' + N' - 1 \quad \text{or} \quad L_n \leqq N_n \leqq M_n.$$

(ii) By Lemma 6.

(iii) By Corollary 2. □

The bounds on N_n are not strong. Better bounds on $E(N_n)$ have been derived by Yao [15], but in order adequately to formulate his "second order analysis" in terms of the generalized Polya-Eggenberger model, a multivariate counterpart of Theorem 1 must be proved. It appears that such an extension of Theorem 1 would also make possible an analysis of the memory requirements of B-trees [13], [15]. Brown's work [4] on AVL (height-balanced) trees parallels Yao's work on 2-3 trees, and it should be observed that the results of this section apply with minor changes to AVL trees as well.

Appendix.

CLAIM 1. *Let t_j be as in § 2, i.e. $t_j = t_0 + js$, where $j \geqq 0$. Let x be any real number, and let*

$$f(n) = \prod_{j=0}^{n-1} \left(1 + \frac{x}{t_j}\right).$$

Then as $n \to \infty$, $f(n) = O(t_n^{x/s})$.

Proof. Clearly,

$$f(n) = \frac{\Gamma(t_0/s)}{\Gamma((t_0+x)/s)} \cdot \frac{\Gamma((t_n+x)/s)}{\Gamma(t_n/s)}.$$

But it is known that for real u, when $u \to \infty$

$$\Gamma(u) \sim \sqrt{2\pi}\, u^{u-1/2}\, e^{-u}$$

(see [14, p. 254]). The claim follows. □

REFERENCES

[1] A. V. AHO, J. E. HOPCROFT AND J. D. ULLMAN, *The Design and Analysis of Computer Algorithms*, Addison-Wesley, Reading, MA, 1974.

[2] K. B. ATHREYA AND S. KARLIN, *Embedding of urn schemes into continuous time Markov branching processes and related limit theorems*, Ann. Math. Stat., 39 (1968), pp. 1801-1817.

[3] K. B. ATHREYA AND P. E. NEY, *Branching Processes*, Springer-Verlag, New York, 1972.

[4] M. R. BROWN, *A partial analysis of random height-balanced trees*, SIAM J. Comp., 8 (1979), pp. 33-41.

[5] Y. S. CHOW AND H. TEICHER, *Probability Theory*, Springer-Verlag, New York, 1978.

[6] M. FISZ, *Probability Theory and Mathematical Statistics*, 3rd. ed., John Wiley, New York, 1963.

[7] D. A. FREEDMAN, *Bernard Friedman's urn*, Ann. Math. Stat., 36 (1965), pp. 956-970.

[8] B. FRIEDMAN, *A simple urn model*, Comm. Pure Appl. Math., 2 (1949), pp. 59-70.

[9] P. HALL AND C. C. HEYDE, *Martingale Limit Theory and Its Applications*, Academic Press, New York, 1980.
[10] L. HOLST, *A unified approach to limit theorems for urn models*, J. Appl. Prob., 16 (1979), pp. 154–162.
[11] N. L. JOHNSON AND S. KOTZ, *Urn Models and Their Applications*, John Wiley, New York, 1977.
[12] C. JORDAN, *Calculus of Finite Differences*, 2nd ed., Chelsea, New York, 1960.
[13] D. E. KNUTH, *The Art of Computer Programming, Vol. 3, Sorting and Searching* (2nd pr.), Addison-Wesley, Reading, MA, 1975.
[14] L. M. MILNE-THOMSON, *The Calculus of Finite Differences*, Macmillan, New York, 1960.
[15] A. C-C. YAO, *On random 2-3 trees*, Acta Inform., 9 (1978), pp. 159–170.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 3, July 1985

008

ON NONNEGATIVE SOLUTIONS OF MATRIX EQUATIONS*

H. D. VICTORY, JR.†

Abstract. Let $\mathbf{A}$ be a nonnegative and nontrivial $n \times n$ matrix. In this work, we present necessary and sufficient conditions for the matrix equation, $(\lambda \mathbf{I} - \mathbf{A})x = b$, to possess a nonnegative solution x whenever b is a given nonnegative and nontrivial vector and λ is any given positive parameter. This analysis extends a result of S. Friedland and H. Schneider (SIAM J. Alg. Disc. Meth., 2 (1980), Thm. 7.1, pp. 185-200).

1. Introduction. In some fields of applied mathematics (e.g. radiative transfer, linear transport), numerical approximations of the exact underlying equations often produce conditional equations of the form

$$\lambda x = \mathbf{A}x + b, \tag{1.1}$$

where λ is a positive parameter, b is a nontrivial given vector with nonnegative components, and $\mathbf{A}$ is a nonnegative and nontrivial $n \times n$ matrix. For physical reasons, say, one wishes to conclude the existence of a solution x with nonnegative components. The primary purpose of this note is to utilize the Frobenius structure of the matrix $\mathbf{A}$ to provide equivalent conditions for the "nonnegative" solvability of (1.1) whenever λ is any given, positive number. The results we obtain can be viewed as refinements of the results contained in the work by S. Friedland and H. Schneider [2, Thm. 7.1].

At this point, it may perhaps be useful to provide some definitions and concepts which will play a role in our analysis. The spectral radius of the matrix $\mathbf{A}$ is expressed as $\rho(\mathbf{A})$ and will be assumed positive throughout the discussion. We let $\mathfrak{R}_+^{nn}$ be the set of all $n \times n$ nonnegative matrices, and write $\mathbf{A} \geqq 0$ ($\mathbf{A} \gg 0$) for $\mathbf{A} \in \mathfrak{R}_+^{nn}$ whenever $\mathbf{A}$ has nonnegative (positive) entries, and $\mathbf{A} > 0$ if $\mathbf{A} \geqq 0$ and $\mathbf{A} \neq 0$. Similar definitions will apply to vectors. We shall also assume that $\mathbf{A}$ has been expressed in *Frobenius normal form*,

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} & \cdots & \mathbf{A}_{1\nu} \\ & \mathbf{A}_{22} & & \\ & & \ddots & \vdots \\ 0 & & & \mathbf{A}_{\nu\nu} \end{bmatrix} \tag{1.2}$$

where the diagonal blocks $\mathbf{A}_{\alpha\alpha}$, $\alpha = 1, \cdots, \nu$, are either irreducible or 1×1 null matrices. All subdiagonal blocks are zero. If $h \in \mathfrak{R}^n$, we write $|h|$ to mean the nonnegative vector whose components are the absolute values of the corresponding components of h. $\mathfrak{R}_+^n$ will denote the cone of nonnegative vectors in $\mathfrak{R}^n$.

We recall that the (reduced) graph $G(\mathbf{A})$ is a subset of $\langle \nu \rangle \times \langle \nu \rangle$, where the vertex set $\langle \nu \rangle$ is given by $\langle \nu \rangle = \{1, 2, \cdots, \nu\}$ and $G(\mathbf{A}) = \{(\alpha, \beta) \in \langle \nu \rangle \times \langle \nu \rangle : \mathbf{A}_{\alpha\beta} \neq 0\}$. Each (α, β) is called an arc of $G(\mathbf{A})$ and we see from (1.2) that $\alpha \leqq \beta$. Also, $(\alpha, \alpha) \in G(\mathbf{A})$, $1 \leqq \alpha \leqq \nu$, unless $\mathbf{A}_{\alpha\alpha}$ is the 1×1 null matrix 0. A *simple path* or *chain from α to β in $G(A)$* is a sequence $\pi = (\alpha_0, \alpha_1, \cdots, \alpha_s)$ where either $s \geqq 1$, $1 \leqq \alpha = \alpha_0 < \cdots < \alpha_s = \beta \leqq \nu$, with $(\alpha_{i-1}, \alpha_i) \in G(\mathbf{A})$, $i = 1, \cdots, s$, or $s = 0$ and $\alpha = \alpha_0 = \beta$, with $(\alpha, \alpha) \in G(\mathbf{A})$. The

* Received by the editors April 7, 1983, and in revised form January 29, 1984.

† Department of Mathematics, Texas Tech University, Lubbock, Texas 79409. This research was completed while the author was an Alexander von Humboldt Research Fellow in the Mathematics Departments of the Universities of Frankfurt and Kaiserslautern, Federal Republic of Germany. Generous support was also provided by the Faculty Developmental Leave Program of Texas Tech University for the academic year, 1982-1983.

support of π is the set $\operatorname{supp}\pi=\{\alpha_0, a_1, \cdots, \alpha_s\}\subseteq\langle\nu\rangle$, where α_i, $i=0,\cdots,s$, are assumed listed in strictly ascending order. For $1\leqq\alpha,\ \beta\leqq\nu$, we shall say that β has *access to* (*from*) α *in* $G(\mathbf{A})$ if there is a simple path from (to) β to (from) α in $G(\mathbf{A})$.

For any matrix $\mathbf{B}$ and eigenvalue λ, we define $\mathcal{N}_\lambda^n(\mathbf{B})$ to be the space of right or column vectors annihilated by $[\lambda\mathbf{I}-\mathbf{B}]^n$. The *index* of the eigenvalue λ is defined to be the smallest nonnegative integer n_0 such that $\mathcal{N}_\lambda^{n_0}(\mathbf{B})=\mathcal{N}_\lambda^{n_0+1}(\mathbf{B})$. This subspace is the *algebraic eigenspace* of $\mathbf{B}$ belonging to λ, and its elements are called *generalized eigenvectors.*

In our analysis, we shall regard b as partitioned conformably with $\mathbf{A}$ in (1.2) as $b^t=(b_{(1)}^t, b_{(2)}^t, \cdots, b_{(\nu)}^t)$ where the superscript t denotes transpose. The support of $b>0$ is the set $\{\alpha=1,\cdots,\nu\colon b_{(\alpha)}\neq 0\}$ and is denoted supp b. Before stating our main result, we call a nonnegative eigenvalue of $\mathbf{A}\in\mathfrak{N}_+^{nn}$ *distinguished* if there is an associated, nonnegative right (i.e. column) eigenvector. We list the distinguished eigenvalues as $\rho(\mathbf{A})=\lambda_1>\lambda_2>\cdots>\lambda_\eta$, where the integer $\eta\geqq 1$. The result we prove is:

THEOREM. *Let* $\mathbf{A}\in\mathfrak{N}_+^{nn}$ *with* $\rho(\mathbf{A})>0$, *and suppose the right* (*column*) *vector* $b>0$. *Moreover, assume that* $\mathbf{A}$ *is in Frobenius normal form* (1.2) *and that the vector* b *is partitioned conformably with* $\mathbf{A}$. *Then the following are equivalent*:

(i) *There is an* $x\in\mathfrak{N}_+^n$, $x>0$, *such that* $(\lambda\mathbf{I}-\mathbf{A})x=b$.

(ii) *No* α, *for which* $\rho(\mathbf{A}_{\alpha\alpha})\geqq\lambda$, *has access in* $G(\mathbf{A})$ *to any vertex in* supp b.

(iii) $|h|^t b=0$ *for every* h *in the algebraic subspace of* $\mathbf{A}^t$ *associated with only the distinguished eigenvalues* $\lambda_i\geqq\lambda$.

(iv) $\lim_{m\to\infty}\sum_{j=0}^m(\mathbf{A}/\lambda)^j b$ *exists.*

(v) $\lim_{m\to\infty}(\mathbf{A}/\lambda)^m b=0$.

Further, if (iv) *holds and* $x=\lim_{m\to\infty}\sum_{j=0}^m(\mathbf{A}/\lambda)^j b$, *then*

(1.3a) $\qquad x_{(\beta)}=0$ *if* β *does not have access to* $\alpha\in$ supp b;

(1.3b) $\qquad x_{(\beta)}\gg 0$ *if* β *has access to at least one* $\alpha\in$ supp b.

As pointed out by Friedland and Schneider in their concluding remarks to [2, Thm. 7.1], the equivalence of conditions (i) and (ii) in this theorem was shown by D. H. Carlson [1, Thm. 1]; thus the equivalence of conditions (i) and (ii) of our theorem is a direct generalization of Carlson's results.

It is interesting to see that the graph theoretic condition expressed by (ii) can be reformulated in terms of a purely algebraic condition (iii) on the distinguished eigenvalues of $\mathbf{A}$. The latter condition is fundamentally a condition about indices not included in supp b; and states that nonnegative solvability of $(\lambda\mathbf{I}-\mathbf{A})x=b$ is equivalent to the nonintersection of the support of the vector b with the union of supports of the generalized eigenvectors belonging to the distinguished eigenvalues $\lambda_i\geqq\lambda$. In § 2, we shall prove our main result and give some illustrative examples in § 3.

2. Proof of the main result. The bulk of this section will be devoted to proving the equivalence of the "alternative condition" (iii) to the other conditions in the statement of the theorem. Toward this end, we first prove a simple result about the distinguished eigenvalues of the matrix $\mathbf{A}$ which is very close to a result stated and proved in [3] by Frobenius. Then we give some insight into the structure of the *adjoint* algebraic eigenspaces associated with the distinguished eigenvalues. This latter result can be proven by an analysis similar to that by U. Rothblum in [5] and seems to be interesting in its own right.

PROPOSITION 1. *Let* $\lambda_0>0$ *and let* $(\mathbf{A}_{\alpha_i\alpha_i}\colon i=1,2,\cdots,N(\lambda_0))$ *be the collection of diagonal blocks in* (1.2) *with spectral radius* λ_0. *Then* λ_0 *itself is a distinguished eigenvalue of* $\mathbf{A}$, *if, and only if, there is at least one* i_0, $1\leqq i_0\leqq N(\lambda_0)$, *such that the vertex* α_{i_0} *has*

access (a) *either from only itself in* $G(A)$ *or* (b) *from only those vertices* β *in* $G(\mathbf{A})$ *for which* $\rho(\mathbf{A}_{\beta\beta}) < \lambda_0$.

Proof. Suppose either (a) or (b) is true. Then, let $\Pi_{\alpha_{i_0}}$ consist of precisely those vertices having access to α_{i_0} in $G(A)$. The matrix $\mathbf{A}$ is invariant on the subspace of (right or column) vectors in $\mathfrak{R}^n$ which have support at most $\Pi_{\alpha_{i_0}}$. We can construct a nonnegative (right) eigenvector to λ_0 by first defining it zero on those vertices not in $\Pi_{\alpha_{i_0}}$, and then using the constructive procedure in Gantmacher [4, Thm, 6 (Part 2), pp. 77-78] to generate the portion which is positive on those vertices comprising $\Pi_{\alpha_{i_0}}$.

Conversely, let λ_0 be a distinguished eigenvalue and x_0 an associated nonnegative right eigenvector. Let Π_0 be a collection of those vertices comprising the support of x_0. Since $\mathbf{A}$ leaves invariant the subspace S_0 of (column or right) vectors with support at most Π_0, we can employ a suitable permutation to express $\mathbf{A}$ as

$$\mathbf{A} = \begin{pmatrix} \mathbf{A}_0 & \mathbf{B} \\ \mathbf{0} & \mathbf{C} \end{pmatrix}, \tag{2.1}$$

where $\mathbf{A}_0 = \mathbf{A}|S_0$ and $\rho(\mathbf{A}_0) = \lambda_0$. Without loss of generality, we can assume that $\mathbf{A}_0$ is expressed in Frobenius normal form (1.2). Again, the analysis in Gantmacher [4, Thm. 6 (Part 2), pp. 77-78) enables us to deduce the existence of a vertex α_{i_0} having the properties asserted in the statement of the proposition. Just as importantly, we see by Gantmacher's arguments that any other vertex $\beta \in \Pi_0$, for which $\rho(\mathbf{A}_{\beta\beta}) = \lambda_0$, has access neither to nor from α_{i_0} in $G(\mathbf{A})$. This completes the proof of Proposition 1.

Remark. A result similar to Proposition 1 is valid for the distinguished eigenvalues to the adjoint matrix $\mathbf{A}^t$ if we replace "access from" by "access to" in the statement of the proposition.

We now proceed to give some insight into the analytic properties of the algebraic eigenspace of $\mathbf{A}^t$ associated with a distinguished eigenvalue λ_0 of $\mathbf{A}$. In order to correctly formulate our result, we shall require some additional terminology which can be seen to be similar to that of [2, § 4] for describing the *singular vertices* of $G(\mathbf{A})$ (i.e. those vertices α for which $\rho(\mathbf{A}_{\alpha\alpha}) = \rho(\mathbf{A})$). Let π be a simple path from α to β in $G(\mathbf{A})$ and let $k_{\lambda_0}(\pi)+1$ be the number of vertices in the support of π whose associated diagonal blocks in (1.2) possess eigenvalue λ_0. We label such vertices, "λ_0-vertices". We set

$$k_{\lambda_0}(\alpha, \beta) = \max\{k_{\lambda_0}(\pi): \pi \text{ is a path from } \alpha \text{ to } \beta \text{ in } G(\mathbf{A})\} \tag{2.2}$$

(and set $k_{\lambda_0}(\alpha, \beta) = -\infty$ if there is no path from α to β and equal to -1 if there is no λ_0-vertex in any π). The integer $k_{\lambda_0}(\alpha, \beta)$ is denoted as the λ_0-*distance* from α to β. A simple path π from α to β is labeled a *maximal* λ_0-*path* if the number of λ_0-vertices in supp π is $k_{\lambda_0}(\alpha, \beta)+1$. We say that vertex α has λ_0-access to β in exactly n steps if $k_{\lambda_0}(\alpha, \beta)+1 = n$.

Let Γ_{λ_0} be the collection of M vertices $\alpha_j \in \langle \nu \rangle$, $j = 1, \cdots, M$, for which $\mathbf{A}_{\alpha_j\alpha_j}$ has λ_0 as an eigenvalue, and select a particular $\alpha_J \in \Gamma_{\lambda_0}$. Define $\Pi_{\lambda_0}(J)$ to be the collection of all vertices $\alpha_i \in \langle \nu \rangle$ having access from α_J in $G(\mathbf{A})$ and $k_{\lambda_0}(J)+1$ to be the largest number of λ_0-vertices in any simple path with initial vertex α_J. We then partition $\Pi_{\lambda_0}(J)$ in the following manner:

(2.3)

$$\mathfrak{P}_1^{\lambda_0}(J) = \{\alpha_l \in \Pi_{\lambda_0}(J): \alpha_l \text{ has } \lambda_0\text{-access from } \alpha_J \text{ in exactly one step}\},$$

$$\mathfrak{P}_2^{\lambda_0}(J) = \{\alpha_l \in \Pi_{\lambda_0}(J): \alpha_l \text{ has } \lambda_0\text{-access from } \alpha_J \text{ in exactly two steps}\},$$

$$\vdots$$

$$\mathfrak{P}_{k_{\lambda_0}(J)+1}^{\lambda_0}(J) = \{\alpha_l \in \Pi_{\lambda_0}(J): \alpha_l \text{ has } \lambda_0\text{-access from } \alpha_J \text{ in exactly } (k_{\lambda_0}(J)+1) \text{ steps}\}.$$

(We note that such a partitioning provides a recipe for constructing a simple path between any two vertices in $\Pi_{\lambda_0}(J)$ with a maximum number of λ_0-vertices.)

The next proposition describes the support structure of the algebraic eigenspace of $\mathbf{A}^t$ belonging to λ_0:

PROPOSITION 2. (1) *A basis for the algebraic eigenspace of* $\mathbf{A}^t$ *belonging to* λ_0 *can be chosen to consist of generalized (right) eigenvectors* $x_l(J, \lambda_0)$, $J \in \Gamma_{\lambda_0}$, $l = 1, \cdots, N(J)$, *where each* $x_l(J, \lambda_0)$ *has support at most on* $\Pi_{\lambda_0}(J)$. *Also* $x_l(J, \lambda_0)_{(\alpha_J)}$ *is any one of the* $N(J)$ *linearly independent (column) vectors annihilated by* $(\lambda_0 I_{\alpha_J\alpha_J} - \mathbf{A}_{\alpha_J\alpha_J})^{m_l(J)}$, *where* $m_l(J)$ *is the index of* λ_0 *as an eigenvalue of* $\mathbf{A}_{\alpha_J\alpha_J}$ *(here,* $\mathbf{I}_{\alpha_J\alpha_J}$ *is the submatrix of the identity matrix associated with vertex* $\alpha_J \in \langle \nu \rangle$*)*;

(2)

$$\text{Index } \lambda_0 \leqq \max_{J \in \Gamma_{\lambda 0}} (m_1(J) + m_2(J) + \cdots + m_{k_{\lambda_0}(J)+1}(J)), \tag{2.4}$$

where each $m_i(J)$ *is the maximum of the indices of* λ_0 *as an eigenvalue of those* $\mathbf{A}_{\beta\beta}$, *with* β *any* λ_0*-vertex in* $\mathfrak{B}_i^{\lambda_0}(J)$, $i = 1, \cdots, k_{\lambda_0}(J) + 1$.

The proof of Proposition 2 proceeds in the same manner as the proof in [5, Thm. 3.1 (Part 1), pp. 284–288]. We now turn to proving the theorem stated in § 1. Before showing the equivalence of (ii) and (iii), we first show (i)⇒(ii)⇒(iv)⇒(i) and then (iv)⇒(v)⇒(ii) as done in [2, Thm. 7.1].

(i)⇒(ii). Suppose that $(\lambda \mathbf{I} - \mathbf{A})x = b$, $b > 0$, has a nonnegative solution x for $\lambda \leqq \rho(\mathbf{A})$. Then the system

$$(\rho(\mathbf{A})I - \mathbf{A})x = b', \qquad b' = b + (\rho(\mathbf{A}) - \lambda)x \tag{2.5}$$

also has a nonnegative solution, and from [2, Thm. 7.1] we can conclude that no singular vertex β has access to any vertex α in supp b' and thereby to any vertex α in supp b.

To deduce statement (ii), we repeat this analysis. We write (2.5) in block form

$$(\lambda \mathbf{I} - \mathbf{A}_{\alpha\alpha})x_{(\alpha)} - \sum_{\beta > \alpha} \mathbf{A}_{\alpha\beta} x_{(\beta)} = b_{(\alpha)}, \qquad \alpha = 1, \cdots, \nu, \tag{2.6}$$

and observe that the truncated system

$$(\lambda \mathbf{I} - \mathbf{A}_{\alpha\alpha})x_{(\alpha)} - \sum_{\kappa \geqq \beta > \alpha} \mathbf{A}_{\alpha\beta} x_{(\beta)} = \tilde{b}_{(\alpha)}, \qquad \alpha = \delta, \delta + 1, \cdots, \kappa, \quad \kappa \in \langle \nu \rangle, \tag{2.7a}$$

$$\tilde{b}_{(\alpha)} = \sum_{\beta > \kappa} \mathbf{A}_{\alpha\beta} x_{(\beta)} + b_{(\alpha)} \tag{2.7b}$$

also has a nonnegative solution. Therefore, any vertex β satisfying $\rho(\mathbf{A}_{\beta\beta}) \geqq \lambda$ will have no access to any vertex α in supp $\{\tilde{b}_{(\alpha)};\ \alpha = \delta, \cdots, \kappa\}$ and thereby to any vertex α in supp b. Condition (ii) follows.

From (ii), we see that Γ_b, the set of all vertices in $\langle \nu \rangle$ which have access to any vertex in supp b, includes no vertex α for which $\rho(\mathbf{A}_{\alpha\alpha}) \geqq \lambda$. The subspace of (column or right) vectors, S_b, having support at most equal to the vertices in $\Gamma_b \cup$ supp b, reduces $\mathbf{A}$; and hence a suitable permutation transformation enables us to express $\mathbf{A}$ in the form (2.1), where $\mathbf{A}_0$ here is $\mathbf{A}|S_b$ and $\rho(\mathbf{A}_0) < \lambda$. Thus, by using the arguments in Friedland and Schneider [2, Thm. 7.1], we can deduce (iv), and then (i) (via, say, a Neumann series argument).

Showing (iv)⇒(v) is trivial, and we now turn to proving (v)⇒(ii) by a contradiction argument. Suppose there is a vertex α in $\Gamma_b \cup$ supp b for which $\rho(\mathbf{A}_{\alpha\alpha}) \geqq \lambda$. Then, in (2.1), $\rho(\mathbf{A}_0) \geqq \lambda$, and there is an integer N such that the support of $(\mathbf{A}/\lambda)^N b$ will

include at least some of the components comprising the vertex α. Letting b_0 (say) be the block of b conformable with $\mathbf{A}_0$, we have for any $n > N$ that

$$\left[\left(\frac{\mathbf{A}}{\lambda}\right)^n b\right]_{(\alpha)} = \left[\left(\frac{\mathbf{A}_0}{\lambda}\right)^n b_0\right]_{(\alpha)} > \left(\frac{\rho(\mathbf{A}_{\alpha\alpha})}{\lambda}\right)^{n-N}\left(y_\alpha^t\left[\frac{\mathbf{A}_0^N}{\lambda^N} b_0\right]_{(\alpha)}\right)x_\alpha$$

where y_α^t and x_α are the left (row) and right (column) eigenvectors, respectively, to $\mathbf{A}_{\alpha\alpha}$ belonging to $\rho(\mathbf{A}_{\alpha\alpha})$. Statement (v) is thereby violated.

We now show (iii)$\Rightarrow$(ii). Let us first list the distinguished eigenvalues greater than or equal to λ as $\lambda_1(=\rho(\mathbf{A})) > \lambda_2 > \cdots > \lambda_M \geqq \lambda$. For λ_1, we collect all the singular vertices $\alpha_1, \cdots, \alpha_N$ in a set Γ_{λ_1} and we let Π_{λ_1} be Γ_{λ_1} plus other vertices in $G(\mathbf{A})$ which have access from at least one singular vertex in Γ_{λ_1}. The results of U. Rothblum [5, Thm. 3.1] can now be exploited to produce a basis for the algebraic eigenspace of $\mathbf{A}^t$ corresponding to λ_1 consisting solely of nonnegative vectors. The union of the supports of these basis vectors is precisely Π_{λ_1}. The requirement that $|h^t|b = 0$ for any h in the algebraic eigenspace of $\mathbf{A}^t$ belonging to λ_1 then means that b must vanish at all the vertices in Π_{λ_1}.

We note that Π_{λ_1} includes all vertices α for which $\rho(\mathbf{A}_{\alpha\alpha}) = \lambda > \lambda_2$ and some (but not all!) for which $\rho(\mathbf{A}_{\alpha\alpha}) = \lambda_2$. That all vertices α for which $\rho(\mathbf{A}_{\alpha\alpha}) > \lambda_2$ are in Π_{λ_1} is easy to see: if all are not in Π_{λ_1}, we list the excluded ones as $\alpha_1, \alpha_2, \cdots, \alpha_r$ and pick the α_{r_0} with the largest spectral radius. Then it (or possibly some other vertex whose diagonal block in (1.2) has the same spectral radius) has access only from itself or from those vertices β for which $\rho(\mathbf{A}_{\beta\beta}) < \rho(\mathbf{A}_{\alpha_{r_0}\alpha_{r_0}})$, and therefore $\rho(\mathbf{A}_{\alpha_{r_0}\alpha_{r_0}})$ would then be a distinguished eigenvalue, by Proposition 1. A similar analysis applies to the other α_j, $j = 1, \cdots, r$. Because λ_2 is a distinguished eigenvalue, we also know from Proposition 1 that Π_{λ_1} cannot contain all of those vertices α for which $\rho(\mathbf{A}_{\alpha\alpha}) = \lambda_2$. Let us enumerate the λ_2-vertices excluded from Π_{λ_1} as $\alpha_1, \alpha_2, \cdots, \alpha_{N(\lambda_2)}$, and define Π_{λ_2} to be the set of all vertices in $\langle\nu\rangle$ not in Π_{λ_1} which have access from at least one of the α_i, $i = 1, \cdots, N(\lambda_2)$.

Now, a close perusal of the proof of Proposition 2 outlined in the beginning of this section—or, equivalently, a perusal of part (i) of [5, Thm. 3.1, pp. 285–288]—will convince us that each α_i, $i = 1, 2, \cdots, N(\lambda_2)$, will be associated with a generalized eigenvector of $\mathbf{A}^t$ associated with λ_2 whenever the set of accessible vertices from each α_i is partitioned as in (2.3). The important thing to note here is that each eigenvector associated with each α_i, $i = 1, 2, \cdots, N(\lambda_2)$, can be chosen to be positive on the appropriately accessible vertices of Π_{λ_2}. Therefore, (iii) forces b to vanish on Π_{λ_2}.

The proof of (ii) is a repetition of this analysis. We finally note that $\Pi_{\lambda_1} \cup \Pi_{\lambda_2} \cup \cdots \cup \Pi_{\lambda_M}$ includes all vertices α for which $\rho(\mathbf{A}_{\alpha\alpha}) \geqq \lambda$ (if $\lambda \neq \lambda_M$). If not, there would exist one excluded vertex which would generate a distinguished eigenvalue not equal to any λ_i, $i = 1, \cdots, M$. This contradicts the fact that all λ_i, $i = 1, \cdots, M$, were the only distinguished eigenvalues greater than or equal to λ.

Condition (iii) trivially follows from (ii), since a basis for the algebraic eigenspace of $\mathbf{A}^t$ belonging to a distinguished eigenvalue $\lambda_0 \geqq \lambda$ can be constructed via Proposition 2 to have support only on the subset of $\langle\nu\rangle$ described in (ii). Any other set of basis elements can, of course, be represented as linear combinations of elements constructed via Proposition 2.

Items (1.3a,b) are easily shown. This completes the proof of our main result.

Remark. The proof that (ii) is a consequence of (iii) shows that we can *equivalently* replace $|h^t|$ by h^t in the statement of (iii).

3. Some examples. We now give some examples to illustrate our theory.

Example 1. Let **A** be given by

$$\mathbf{A} = \begin{bmatrix} 1 & \frac{1}{4} & \frac{1}{2} \\ 0 & \frac{3}{4} & 0 \\ 0 & 0 & \frac{1}{2} \end{bmatrix}. \tag{3.1}$$

We easily see that $\rho(\mathbf{A}) = 1$, and 1 is the only distinguished eigenvalue. We also see that vertices $\{1, 2, 3\}$ have access from the singular vertex 1, and thus there is *no* purely nonnegative solution of $(\lambda \mathbf{I} - \mathbf{A})x = b$ for $b > 0$ for any $\lambda \in (0, 1]$. We also note that $(1, 1, 1)^t$ is a basis for the algebraic eigenspace of $\mathbf{A}^t$, and, of course, there is no $b > 0$ annihilated by this vector. In other words, nonnegative solvability is possible for $b^t = (0, 0, 0)$, when $\lambda \in (0, 1]$, with the result that $x^t = (0, 0, 0)$, $\lambda < 1$ and can be a nonnegative multiple of $(1, 0, 0)$ when $\lambda = 1$.

Example 2. Let **A** be given by

$$\mathbf{A} = \begin{bmatrix} \frac{1}{2} & 0 & \frac{1}{2} \\ 0 & \frac{3}{4} & \frac{1}{4} \\ 0 & 0 & 1 \end{bmatrix}. \tag{3.2}$$

We see that $\rho(\mathbf{A}) = \lambda_1 = 1$ with $\lambda_2 = \frac{3}{4}$ and $\lambda_3 = \frac{1}{2}$ also distinguished eigenvalues. We note that $v(1) = (1, 1, 1)^t$ and $h(1) = (0, 0, 1)^t$ are eigenvectors of **A** and $\mathbf{A}^t$ respectively corresponding to $\lambda = 1$. Corresponding to $\lambda_2 = \frac{3}{4}$ are eigenvectors $v(\frac{3}{4}) = (0, 1, 0)^t$ and $h(\frac{3}{4}) = (0, 1, -1)^t$ of **A** and $\mathbf{A}^t$ respectively. Similarly, corresponding to $\lambda_3 = \frac{1}{2}$ are eigenvectors $v(\frac{1}{2}) = (1, 0, 0)^t$ and $h(\frac{1}{2}) = (1, 0, -1)^t$.

For values of $\lambda \in (\frac{3}{4}, 1]$, we see that $b > 0$ must have support at most on vertices 1 and 2 in order to guarantee a solution $x > 0$. For $\lambda \in (\frac{1}{2}, \frac{3}{4}]$, b must have support on vertex 1 only, and there is no solution $x > 0$ of $(\lambda \mathbf{I} - \mathbf{A})x = b$ for any $b > 0$ when $\lambda \in (0, \frac{1}{2}]$.

Example 3. The following is the Frobenius normal form of an example given by Rothblum [5, p. 284]:

$$\mathbf{A} = \begin{bmatrix} 3 & 7 & 0 & 6 & 0 & 0 & 0 & 0 & 0 \\ & 3 & 0 & 0 & 3 & 0 & 4 & 0 & 0 \\ & & 1 & 8 & 0 & 8 & 0 & 0 & 0 \\ & & & 2 & 1 & 0 & 0 & 0 & 0 \\ & & & & 3 & 5 & 0 & 0 & 0 \\ & & 0 & & & 3 & 0 & 0 & 0 \\ & & & & & & 0 & 2 & 0 \\ & & & & & & & 3 & 3 \\ & & & & & & & & 1 \end{bmatrix}. \tag{3.3}$$

From Proposition 1, we see that $\rho(\mathbf{A}) = \lambda_1 = 3$ and $\lambda_2 = 1$ are the only distinguished eigenvalues. So, for the matrix equation $(\lambda \mathbf{I} - \mathbf{A})x = b$ with $1 < \lambda \leqq 3$, we must require that $b > 0$ have support on vertex 3 in order to guarantee a solution $x > 0$. For $0 < \lambda \leqq 1$, there is no $b > 0$ for which a nonnegative, nontrivial solution of $(\lambda \mathbf{I} - \mathbf{A})x = b$ will exist.

Remark. In all three examples, we have not treated the case $\lambda > \rho(\mathbf{A})$, as nonnegative solvability of $(\lambda \mathbf{I} - \mathbf{A})x = b$ is well known for any $b > 0$.

Acknowledgments. The author wishes to express his appreciation to Professor Drs. F. Stummel and H. Neunzert of the Mathematics Departments of the Universities of Frankfurt and Kaiserslautern, respectively, for hosting him as an Alexander von Humboldt Research Fellow during the two years, 1982-1984. Moreover, he would like

to acknowledge the kind hospitality shown him by these departments during the period of his stay in the Federal Republic of Germany.

He also wishes to acknowledge the many valuable suggestions of the referees for considerably improving the original statement and presentation of the main result.

REFERENCES

[1] D. H. CARLSON, *A note on M-matrix equations*, J. Soc. Indust. Appl. Math., 11 (1963), pp. 1027–1033.

[2] S. FRIEDLAND AND H. SCHNEIDER, *The growth of powers of a nonnegative matrix*, this Journal, 1 (1980), pp. 185–200.

[3] G. F. FROBENIUS, *Über Matrizen aus nichtnegativen Elementen*, S. B. Kön. Preuss. Akad. Wiss. Berlin (1912), pp. 456–477.

[4] F. R. GANTMACHER, *The Theory of Matrices* II, K. A. Hirsh, transl., Chelsea, New York, 1959.

[5] U. G. ROTHBLUM, *Algebraic eigenspaces of nonnegative matrices*, Linear Algebra and Appl., 12 (1975), pp. 281–292.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 3, July 1985

009

DECOMPOSITION OF A COMPLETE MULTI-PARTITE GRAPH INTO ISOMORPHIC CLAWS*

SHINSEI TAZAWA†

Abstract. A graph is called a complete m-partite graph, denoted by $K_m(n_1, n_2, \cdots, n_m)$, if its point set is partitioned into m subsets of $n_1, n_2, \cdots, n_m$ points each such that every pair of points in the same subset is not adjacent and if each point in one subset is adjacent to all points in the other subsets. A complete bipartite graph $K_2(1, c)$ with $c+1$ points and c lines is called a claw of degree c. $K_m(n_1, n_2, \cdots, n_m)$ is said to have a claw-decomposition of degree c if it is a union of line-disjoint subgraphs each isomorphic to a claw of degree c. In this paper, a necessary and sufficient condition for $K_m(n_1, n_2, \cdots, n_m)$ to have a claw-decomposition of degree c is given.

1. Introduction. The reader is referred to [1] for any term not defined below. Consider a graph without loops or multiple lines. Let $m(\geqq 2)$ be an integer and let $K_m(n_1, n_2, \cdots, n_m)$ denote the complete m-partite graph whose point set may be partitioned into $V_1, V_2, \cdots, V_m$, where the cardinality $|V_i| = n_i$ for $i = 1, 2, \cdots, m$ and points u, v are adjacent if and only if they belong to distinct sets V_i, V_j in the partition. Each set V_i is called an independent set. In the particular case when $n_1 = n_2 = \cdots = n_m = 1$, the graph is a complete graph. A complete bipartite graph $K_2(1, c)$ with $c+1$ points and c lines is called a claw or star of degree $c(\geqq 2)$, and it is denoted by S_c. Since a claw is a tree, we call the point of degree c the root. Given a claw S_c, a complete m-partite graph $K_m(n_1, n_2, \cdots, n_m)$ is said to have a claw-decomposition of degree c if it is a union of line-disjoint subgraphs each isomorphic to S_c.

The problems on claw-decomposition of graphs arise in filing theory, since a claw-decomposition yields a file organization scheme. Yamamoto et al. [4] completely solved the problem of claw-decomposability of a complete graph. The decomposition yields an optimal binary-valued balanced file organization scheme of order two under a measure of redundancy [5]. As for a multiple-list structure well known in filing theory, Yamamoto et al. [6] showed that a claw-decomposition of $K_m(n_1, n_2, \cdots, n_m)$ gives an efficient structure. Ushio et al. [3], in the case $n_1 = n_2 = \cdots = n_m$, gave a necessary and sufficient condition for $K_m(n_1, n_2, \cdots, n_m)$ to have a claw-decomposition of degree c. In this paper, we shall establish a necessary and sufficient condition for a complete m-partite graph to have a claw-decomposition of degree c. The result covers both the claw-decomposition theorem in [4] and the one in [3].

2. Main theorem. Let $N = \sum_{i=1}^{m} n_i$ and $\nu = \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} n_i n_j$. The notation N, ν will often be used throughout this paper. In order for $K_m(n_1, n_2, \cdots, n_m)$ to have a claw-decomposition of degree c, the condition that the number of all lines

(i) ν is an integral multiple of c

is obviously necessary. The following theorem states that this is not sufficient. In the theorem we can assume $n_1 \leqq n_2 \leqq \cdots \leqq n_m$ without loss of generality.

THEOREM 2.1. *Let $n_1 \leqq n_2 \leqq \cdots \leqq n_m$ be $m(\geqq 2)$ positive integers satisfying condition* (i). *Then a complete m-partite graph $K_m(n_1, n_2, \cdots, n_m)$ has a claw-decomposition of degree c if and only if the following conditions hold*:

(ii) *ν/c is an integral multiple of $N - n_m$ if $N - n_m < c$, and*

(iii) *$\nu/c \geqq N - n_m$ if $N - n_m \geqq c$.*

The proof of this theorem will be given in § 4.

* Received by the editors December 9, 1980, and in final revised form March 1, 1984.
† Department of Mathematics, Kinki University, Osaka 577, Japan.

3. Some lemmas. The lemmas given in this section will be used to prove the sufficiency of conditions (ii) and (iii).

LEMMA 3.1. *Let a_{ip} $(p=1,2,\cdots,n_i;\ i=1,2,\cdots,m)$ be N nonnegative integers satisfying $\sum_{i=1}^{m}\sum_{p=1}^{n_i} a_{ip}c=\nu$ and $a_{i1}\leqq a_{i2}\leqq\cdots\leqq a_{in_i}$ for all i. Then $K_m(n_1,n_2,\cdots,n_m)$ has a claw-decomposition of degree c such that the point v_{ip}, $1\leqq p\leqq n_i$, $1\leqq i\leqq m$, is the root of a_{ip} copies of S_c if and only if*

$$c\left(\sum_{i=1}^{m}\sum_{p=1}^{k_i} a_{ip}\right)\geqq\sum_{i=1}^{m-1}\sum_{j=i+1}^{m} k_ik_j \tag{3.1}$$

holds for every set of m integers k_i where $0\leqq k_i\leqq n_i$.

The necessity of the above condition is obvious. The sufficiency follows from Moon's result in [2].

LEMMA 3.2. *Let $c=\nu/N$. Suppose k_i satisfies $0\leqq k_i\leqq n_i$ for $1\leqq i\leqq m$. Then*

$$c\sum_{i=1}^{m} k_i\geqq\frac{1}{2}\sum_{i\neq j} k_ik_j. \tag{3.2}$$

Proof. Let $K=\sum_{i=1}^{m} k_i$. By straightforward calculation we have

$$K\left(N^2-\sum_{i=1}^{m} n_i^2\right)-N\left(K^2-\sum_{i=1}^{m} k_i^2\right)\geqq\sum_{i=1}^{m}(n_i-k_i)\{(N-n_i)K-(K-k_i)k_i\}.$$

Thus we have (3.2) by $c=\nu/N$.

LEMMA 3.3. *Let $n_1\leqq n_2\leqq\cdots\leqq n_m$ be positive integers satisfying conditions* (i) *and* (iii). *We write ν/c in the form*

$$\frac{\nu}{c}=Na+n_1+n_2+\cdots+n_b+l\qquad(0\leqq b<m,\ 0\leqq l<n_{b+1}). \tag{3.3}$$

Then if $N-n_{b+1}\geqq(a+1)c$ is satisfied, $K_m(n_1,n_2,\cdots,n_m)$ has a claw-decomposition of degree c.

Proof. Let $I_1=\{1,2,\cdots,b\}$ and $I_2=\{b+2,b+3,\cdots,m\}$, and define

$$a_{ip}=\begin{cases}a+1 & \text{for } i\in I_1,\\ a+1 & \text{for } i=b+1,\ n_{b+1}-l<p\leqq n_{b+1},\\ a & \text{otherwise.}\end{cases} \tag{3.4}$$

Then clearly $c(\sum_{i=1}^{m}\sum_{p=1}^{n_i} a_{ip})=\nu$. Let k_i, $1\leqq i\leqq m$, be integers satisfying $0\leqq k_i\leqq n_i$ and let $K=\sum_{i=1}^{m} k_i$, $N_\beta=\sum_{i\in I_\beta} n_i$ and $K_\beta=\sum_{i\in I_\beta} k_i$ for $\beta=1,2$. Put

$$S=c(aK+K_1+k')-\frac{1}{2}\left(K^2-\sum_{i=1}^{m} k_i^2\right), \tag{3.5}$$

where $k'=\max(0,k_{b+1}-n_{b+1}+l)$. Then by Lemma 3.1, it suffices to establish (3.1) which is equivalent to $S\geqq 0$. Define c_1, c_2, c_1' and c_2' as follows:

$$c_\beta=\frac{1}{2N_\beta}\left(N_\beta^2-\sum_{i\in I_\beta} n_i^2\right), \tag{3.6}$$

$$c_\beta'=\frac{1}{2N_\beta(N_1+N_2)}\left\{N_1N_2(N_1+N_2)-\varepsilon_\beta\left(N_1\sum_{i\in I_2} n_i^2-N_2\sum_{i\in I_1} n_i^2-2N_1N_2c\right)\right\} \tag{3.7}$$

for $\beta=1,2$, where $\varepsilon_1=+1$, $\varepsilon_2=-1$. Then S in (3.5) can be expressed as the sum of four parts:

$$S=S_1+S_2+S_3+S_4, \tag{3.8}$$

where $S_\beta = c_\beta K_\beta - (K_\beta^2 - \sum_{i\in I_\beta} k_i^2)/2$ for $\beta = 1, 2$, $S_3 = c_1' K_1 + c_2' K_2 - K_1 K_2$ and $S_4 = \{(a+1)c - c_1 - c_1'\}K_1 + (ac - c_2 - c_2')K_2 + ack_{b+1} + ck' - (K_1 + K_2)k_{b+1}$. We shall show $S_\beta \geqq 0$ for each $\beta = 1, 2, 3, 4$, so that $S \geqq 0$. An application of Lemma 3.2 gives $S_\beta \geqq 0$ for $\beta = 1, 2$.

Next we consider S_3. If $a = 0$, then it is seen from condition (iii) that $b = m - 1$ in (3.3). Consequently S_3 vanishes. Suppose $a \geqq 1$. Then we have $S_3 = \{K_1 N_1 (N_2 - K_2)c_1' + (N_1 - K_1)K_2 N_2 c_2'\}/(N_1 N_2)$. Obviously $c_1' \geqq 0$. Since $n_1 \leqq n_2 \leqq \cdots \leqq n_m$ and $N - n_{b+1} \geqq (a+1)c \geqq 2c$, we get $c_2' \geqq 0$. Hence we have $S_3 \geqq 0$.

We shall finally show $S_4 \geqq 0$. After a few calculations we have $(a+1)c - c_1 - c_1' = ac - c_2 - c_2' = n_{b+1} - (acn_{b+1} + lc)/(N_1 + N_2)$. Since $K_1 + K_2 \leqq N_1 + N_2$ it follows that

$$S_4 \geqq \frac{K_1 + K_2}{N_1 + N_2}\{(n_{b+1} - k_{b+1})(N - n_{b+1} - (a+1)c) + c(n_{b+1} - k_{b+1} - l + k')\}. \tag{3.9}$$

Applying the inequality $N - n_{b+1} \geqq (a+1)c$ which is an assumption, we get $S_4 \geqq 0$. This completes the proof.

4. Proof of Theorem 2.1. In this section we denote m independent sets of $K_m(n_1, n_2, \cdots, n_m)$ by $V_1, V_2, \cdots, V_m$ where $|V_i| = n_i$, $1 \leqq i \leqq m$, and the pth point of V_i by v_{ip}, $1 \leqq p \leqq n_i$.

4.1. Necessity. Let y_{ip}, $1 \leqq p \leqq n_i$, $1 \leqq i \leqq m$, be the number of claws whose roots are a point v_{ip}. Suppose first that $N - n_m < c$. Then we can see easily that $y_{mp} = 0$ for every $p = 1, 2, \cdots, n_m$. We also have $n_m \leqq y_{ip}c \leqq N - n_i$ for all i except m, i.e.,

$$\left\lceil \frac{n_m}{c} \right\rceil \leqq y_{ip} \leqq \left\lfloor \frac{N - n_i}{c} \right\rfloor \quad \text{for } p = 1, 2, \cdots, n_i,\ i = 1, 2, \cdots, m-1, \tag{4.1}$$

where $\lfloor x \rfloor$ is the greatest integer not exceeding x and $\lceil x \rceil$ is the smallest integer not less than x. Since $N - n_m < c$, it can easily be verified from (4.1) that $y_{ip} = \lceil n_m/c \rceil$ $(1 \leqq p \leqq n_i;\ 1 \leqq i \leqq m-1)$, so that we have $\nu/c = (N - n_m)\lceil n_m/c \rceil$, summing y_{ip} over all p and i. Hence we obtain condition (ii). Put $y_i = \sum_{p=1}^{n_i} y_{ip}$, $1 \leqq i \leqq m$. Then if $N - n_m \geqq c$, since $K_m(n_1, n_2, \cdots, n_m)$ has a claw-decomposition of degree c, the inequality $y_i \geqq n_i$ holds for every i except at most one, say j_0. Therefore, we have

$$\frac{\nu}{c} = \sum_{i=1}^{m} y_i \geqq \sum_{\substack{i=1 \\ i \neq j_0}}^{m} n_i = \sum_{i=1}^{m} n_i - n_{j_0}. \tag{4.2}$$

Hence we obtain condition (iii), since $n_{j_0} \leqq n_m$.

4.2. Sufficiency. There are two cases to prove the sufficiency of conditions (ii) and (iii).

Case 1. $N - n_m < c$. Let $a = \nu/(c(N - n_m))$, and put $a_{ip} = a$ for $i = 1, 2, \cdots, m-1$ and $a_{ip} = 0$ for $i = m$. Clearly $c(\sum_{i=1}^{m}\sum_{p=1}^{n_i} a_{ip}) = \nu$. We shall show that (3.1) holds. Consider c_1 such that $(N_1^2 - \sum_{i=1}^{m-1} n_i^2)/2 = N_1 c_1$, where $N_1 = N - n_m$. Since the left side of (3.1) becomes acK_1, where $K_1 = K - k_m$, then we have

$$c\left(\sum_{i=1}^{m}\sum_{p=1}^{k_i} a_{ip}\right) - \sum_{i=1}^{m-1}\sum_{j=i+1}^{m} k_i k_j = \left\{c_1 K_1 - \frac{1}{2}\left(K_1^2 - \sum_{i=1}^{m-1} k_i^2\right)\right\} + (ac - c_1 - k_m)K_1. \tag{4.3}$$

From the relation $ac - c_1 = n_m$ and Lemma 3.2, it follows that (3.1) holds. Hence condition (ii) is sufficient in the case $N - n_m < c$.

Case 2. $N - n_m \geqq c$. We here introduce a new set of m positive integers $n_1', n_2', \cdots, n_i', \cdots, n_m'$ with $n_i' \leqq n_i$ for each i such that they satisfy conditions (i) and

(iii). These integers will be given explicitly later. We partition V_i into two parts for each i: $V_i = V_i^{(1)} \cup V_i^{(2)} (i = 1, 2, \cdots, m)$, where $V_i^{(1)} = \{v_{ip} \mid p = 1, 2, \cdots, n_i'\}$ and $V_i^{(2)} = \{v_{ip} \mid p = n_i' + 1, n_i' + 2, \cdots, n_i\}$. Next we consider a partition of the set E of all lines of $K_m(n_1, n_2, \cdots, n_m)$. We denote by an unordered pair (u, v) the line joining two distinct points u and v. Partition E as follows:

$$E = \left(\bigcup_{h=1}^{2} E^{(h)} \right) \cup \left(\bigcup_{i=1}^{m} E_i \right), \tag{4.4}$$

where

$$E^{(h)} = \{(v_{ip}, v_{jq}) \mid v_{ip} \in V_i^{(h)}, v_{jq} \in V_j^{(h)}, i \neq j\}, \qquad h = 1, 2,$$

$$E_i = \{(v_{ip}, v_{jp}) \mid v_{ip} \in V_i^{(1)}, v_{jq} \in V_j^{(2)}, i \neq j\}, \qquad i = 1, 2, \cdots, m.$$

Then it is reasonable that we consider the above subsets as follows, where $n_i'' = n_i - n_i' (i = 1, 2, \cdots, m)$:

Let $E^{(1)}$ correspond to the set of all lines of $K_m(n_1', n_2', \cdots, n_m')$; $E^{(2)}$ to that of $K_m(n_1'', n_2'', \cdots, n_m'')$; and E_i to that of $K_2(n_i', \sum' n_j'')$ for $i = 1, 2, \cdots, m$, where $\sum'$ indicates that the sum is taken over all j except i.

This consideration states that it is enough to prove that these complete m-partite graphs all have claw-decompositions of degree c for appropriate values of n_i'. We shall now proceed to the definition of n_i'. We write the original parameter n_i in the form $n_i = cx_i + y_i (0 \leqq y_i < c)$ for each $i = 1, 2, \cdots, m$. Then $x_1 \leqq x_2 \leqq \cdots \leqq x_m$, since $n_1 \leqq n_2 \leqq \cdots \leqq n_m$. Thus three cases are considered.

Case 1. $x_m = 0$. Put $n_i' = y_i (i = 1, 2, \cdots, m)$.

Case 2. $x_m \geqq 1$ and $x_{m-1} = 0$. Put $n_i' = y_i (i = 1, 2, \cdots, m-1)$ and $n_m' = c + y_m$.

Case 3. $x_{m-1} \geqq 1$. Put $z_1 = y_1$, $z_2 = y_2, \cdots, z_{m-2} = y_{m-2}$, $z_{m-1} = c + y_{m-1}$ and $z_m = c + y_m$. Let n_1' be the smallest of these z_i's, n_2' the next z_i in order of magnitude, $\cdots$, and n_m' the largest z_i.

As seen in the above cases, it follows that these integers n_i' just defined have a monotone increasing sequence $n_1' \leqq n_2' \leqq \cdots \leqq n_m'$. Let $\nu' = \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} n_i' n_j'$. Then it is seen that ν' is an integral multiple of c, that is, condition (i) is satisfied. Moreover, it can be checked easily that the inequalities

(iii)* $\nu'/c \geqq N' - n_m' \geqq c$

hold, where $N' = \sum_{i=1}^{m} n_i'$, that is, condition (iii) holds. This fact shows that the properties which the parameters n_i have are preserved.

Since $n_i'' (= n_i - n_i')$ is an integral multiple of c for every $i = 1, 2, \cdots, m$, we can observe easily that $K_2(n_i', \sum' n_j'')$, $1 \leqq i \leqq m$, have a claw-decomposition of degree c and that $K_m(n_1'', n_2'', \cdots, n_m'')$ has a claw-decomposition of degree c. Thus it remains only to be proved that $K_m(n_1', n_2', \cdots, n_m')$ has a claw-decomposition of degree c. We first prove the next lemma.

LEMMA 4.1. *With respect to n_i' just defined above, if we write ν'/c in the form*

$$\frac{\nu'}{c} = N'a + n_1' + n_2' + \cdots + n_b' + l \qquad (0 \leqq b < m, 0 \leqq l < n_{b+1}'), \tag{4.5}$$

then the inequality $N' - n_{b+1}' \geqq (a+1)c$ holds.

Proof. Note first that $n_m' < 2c$, which is obvious from the way of constructing n_i'. We may write $N' - n_i' = 2ca_i + \beta_i (0 \leqq \beta_i < 2c)$ for $i = 1, 2, \cdots, m$. Using both $n_m' - n_1' = 2c(a_1 - a_m) + \beta_1 - \beta_m$ and $n_1' \leqq n_2' \leqq \cdots \leqq n_m' < 2c$, we can see that $a_m + 1 \geqq a_1 \geqq a_2 \geqq \cdots \geqq a_m$, that is by putting $\alpha = a_m$ the integers a_i can be presented as

$$a_1 = \alpha + 1, \quad a_2 = \alpha + 1, \cdots, \quad a_s = \alpha + 1, \quad a_{s+1} = \alpha, \quad a_{s+2} = \alpha, \cdots, \quad a_m = \alpha, \tag{4.6}$$

where $0 \leqq s < m$. As the consequence the left side of (4.5) becomes

$$\frac{\nu'}{c} = N'\alpha + r, \tag{4.7}$$

where $r = \sum_{i=1}^{s} n_i' + \sum_{i=1}^{m} n_i'\beta_i/(2c)$. From (4.5) and (4.7), we obtain $a = \lfloor \nu'/(N'c) \rfloor = \alpha + \lfloor r/N' \rfloor$. Since $0 \leqq r < 2N'$, we have two cases to consider. Case (I): $r < N'$. We have $a = \alpha$ and $b \geqq s$ in this case. Case (II): $N' \leqq r < 2N'$. In this case $n_1' + \cdots + n_b' \leqq (\nu'/c) - N'a = r - N' < n_1' + \cdots + n_{b+1}'$ and $\sum_{i=1}^{m} n_i'\beta_i/(2c) < N'$ give us $a = \alpha + 1$ and $b < s$. Thus we have $a_{b+1} = \alpha = a$ in the former case and $a_{b+1} = \alpha + 1 = a$ in the latter case. Therefore, $(N' - n_{b+1}') - (a+1)c = c(a-1) + \beta_{b+1}$. Thus $N' - n_{b+1}' \geqq (a+1)c$ for $a \geqq 1$. In the case $a = 0$, the first inequality of (iii)* gives $b = m - 1$. Thus we can conclude after all that $N' - n_{b+1}' \geqq (a+1)c$ holds for all $a \geqq 0$. This completes the proof of Lemma 4.1.

By using the Lemma 4.1 and Lemma 3.3, it follows that the remaining complete m-partite graph $K_m(n_1', n_2', \cdots, n_m')$ has a claw-decomposition of degree c. This completes the proof of Theorem 2.1.

Acknowledgments. The author wishes to express his thanks to the referees for their valuable criticism and helpful comments.

REFERENCES

[1] F. HARARY, *Graph Theory*, Addison-Wesley, Reading, MA, 1969.

[2] J. W. MOON, *On the score sequence of an n-partite tournaments*, Canad. Math. Bull., 5 (1962), pp. 51-58.

[3] K. USHIO, S. TAZAWA AND S. YAMAMOTO, *On claw-decomposition of a complete multi-partite graph*, Hiroshima Math. J., 8 (1978), pp. 207-210.

[4] S. YAMAMOTO, H. IKEDA, S. SHIGE-EDA, K. USHIO AND N. HAMADA, *On claw-decomposition of complete graphs and complete bigraphs*, Hiroshima Math. J., 5 (1975), pp. 33-42.

[5] ———, *Design of a new balanced file organization scheme with the least redundancy*, Inform. Control, 28 (1975), pp. 156-175.

[6] S. YAMAMOTO, S. TAZAWA, K. USHIO AND H. IKEDA, *Design of a generalized balanced multiple-valued file organization scheme of order two*, Proc. 8th ACM-SIGMOD International Conference on Management of Data, 1978.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 3, July 1985

010

TOPOLOGICAL BANDWIDTH*

F. S. MAKEDON†, C. H. PAPADIMITRIOU‡ AND I. H. SUDBOROUGH§

Abstract. An assignment of unique integers to the vertices of a graph is called a linear layout. The bandwidth of a linear layout is the maximum difference between integers assigned to adjacent vertices. The bandwidth of a graph is the minimum bandwidth of any layout of the graph. The topological bandwidth of a graph is the minimum bandwidth of all graphs that can be obtained from this graph by subdividing its edges with some number of degree two vertices.

Topological bandwidth is compared to other, seemingly unrelated, parameters of a graph: its cutwidth [5], [8], its modified cutwidth [12], its search number [16], and its node search number [20]. It is shown that the topological bandwidth of a graph is never greater than its modified cutwidth plus one and never smaller than its node search number. Furthermore, for any degree 3 graph G, the topological bandwidth of G is identical to the modified cutwidth of G plus one and is also identical to the node search number of G. It is also shown that the topological bandwidth of any graph is never greater than its cutwidth and never less than its search number minus one.

The topological bandwidth of a binary tree is also considered. A forbidden subtree characterization of topological bandwith k, for each $k \geq 1$, in binary trees is given. It is also noted that there is a $O(n \log n)$ algorithm to compute the topological bandwidth of an arbitrary binary tree and that the topological bandwidth of a complete binary tree of height h is $\lceil h/2 \rceil$. Furthermore, a lower bound on the size of any binary tree with topological bandwidth k is given.

It is shown that the problem of determining, given a graph G and an integer k, whether the topological bandwidth of G is at most k is NP-complete. In fact, the problem is shown to be NP-complete even when restricted to graphs with degree 3. It is also shown that the Min Cut Linear Arrangement problem, the Search Number problem, the Modified Cutwidth problem, and the Node Search Number problem are NP-complete even when restricted to graphs with maximum vertex degree three.

Finally, graphs with topological bandwidth two are characterized. This suggests a linear time algorithm for recognizing graphs with topological bandwidth two. It is also noted that the problem of deciding, given a graph G, whether the topological bandwidth of G is at most k can be solved in $O(|G|^k)$ steps, for all $k \geq 1$.

1. Introduction.

Let G be a finite undirected graph. A (*one-dimensional*) *layout* or linear layout of G is a function assigning to each node of G a unique integer. The *bandwidth of G with respect to a layout L*, denoted by $b(G, L)$, is $\max \{|L(x) - L(y)| \{x, y\}$ is an edge in $G\}$. The *bandwidth of G*, denoted by $b(G)$, is $\min \{b(G, L) | L$ is a layout of $G\}$.

The problem of determining the bandwidth of a graph arises in the manipulation of sparse matrices. Let $A = (a_{ij})$ be a square matrix. One can define the graph $G(A)$, which has one vertex for each row (column) of A and an edge connecting vertex i to vertex j exactly when either the entry a_{ij} or the entry a_{ji} is nonzero. The graph $G(A)$ has bandwidth k if and only if there is a simultaneous row-column permutation of A such that all nonzero entries appear within k of the main diagonal. That is, the matrix A has bandwidth k if and only if there is a permutation matrix P such that $P \cdot A \cdot P^T$

* Received by the editors April 11, 1983, and in revised form January 23, 1984. An extended abstract of this work appears in the Proceedings of the 8th Colloquium on Trees in Algebra and Programming, held in L'Aquila, Italy, March 9-11, 1983.

† Department of Computer Science, Illinois Institute of Technology, Chicago, Illinois 60616. The work of this author was supported in part by the National Science Foundation under grant MCS 81-09280.

‡ Department of Computer Science, National Technical University of Athens, Athens, Greece. A portion of this work was completed while this author was at the National Technical University of Athens in Athens, Greece.

§ Electrical Engineering and Computer Science Departments, Northwestern University, Evanston, Illinois 60201. The work of this author was supported in part by a research grant awarded by the Fulbright Exchange program, and by the National Science Foundation under grant MCS 81-09280. A portion of this work was completed while this author was at the National Technical University of Athens in Athens, Greece.

has this $2k+1$ diagonal form. Being able to rewrite a sparse matrix so that all of its nonzero entries are close to the main diagonal is clearly a desirable feature for efficient storage and processing. See [1], [2], [3], [13] for a discussion of bandwidth and its applications. More recently, it has been shown that bandwidth provides some very useful insights into the computational complexity of graph problèms and other combinatorial problems [17], [18], [26].

It is known that finding the bandwidth of a matrix is an NP-hard problem [21]. In the case of graphs it is known that the problem remains NP-hard even when restricted to trees with maximum vertex degree three [6].

In this paper we study a natural generalization of bandwidth. We consider the *topological bandwidth* of a graph. A graph G' is said to be a homeomorphic image of a graph G if G' can be obtained from G by subdividing edges in G with an arbitrary number of degree two vertices. The topological bandwidth of G, denoted by $tb(G)$, is $\min\{b(G')|G' \text{ is a homeomorphic image of } G\}$. See Fig. 1.1 below for an example.

Topological bandwidth has an interesting sparse matrix interpretation. Let A be a matrix arising from a linear system $Ax = b$. It is, of course, quite possible that there exists no permutation matrix P such that $P \cdot A \cdot P^T$ has all of its nonzero entries close to the main diagonal. In such a case, we may try the following approach. We may replace a term $a_{ij}x_j$ of the system by a new variable y and add a new equation of the form $a_{ij}x_j = y$. This has the effect of adding a degree 2 vertex into the edge $\{i, j\}$ of $G(A)$. So, the topological bandwidth of $G(A)$ is the smallest bandwidth of any system equivalent to $Ax = b$ that can be obtained by a sequence of such substitutions. If the number of degree 2 vertices added is not too large, the resulting small bandwidth system may be more economical to work with.

As another application, one may think of a graph G as representing a data structure that is to be embedded into a linear list, e.g. sequential storage in a computer. The topological bandwidth of G can then be interpreted as a bound on the maximum distance spanned by any pointer. That is, items that are adjacent in the graph G are connected now by a sequence of pointers. Also, in a VLSI application, the graph G can be interpreted as a circuit that for reasons of automation is to have all of its gates laid out along a linear line [19], [28], [29], [30]. The degree 2 vertices that are inserted into the edges of G can be interpreted as "drivers" or "repeaters," which are used to propagate the signal along a long interconnection. The goal in this application is to minimize the length of the longest interconnection.

We prove various graph theoretic and algorithmic properties about topological bandwidth. In § 2 we establish close connections between the topological bandwidth of a graph and other, seemingly unrelated, parameters of a graph: its cutwidth [4], [5], [8], [11], [14], its search number [10], [15], [16], its node search number [20], and its modified cutwidth [12], [22]. We show that, for any graph G, the topological bandwidth

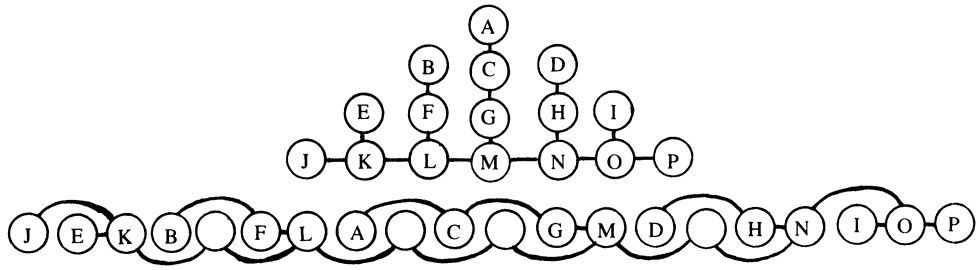


FIG. 1.1. (a) *A tree with bandwidth* 3 *and topological bandwidth* 2. (b) *A bandwidth* 2 *layout of a homeomorphic image of the tree in* (a).

of G is at most its modified cutwidth plus one. This improves the result that, for all graphs G, tb (G) is at most the cutwidth of G, which has been independently observed by F. S. Makedon [14] and F. R. K. Chung [4]. We show also that, for all graphs G with maximum vertex degree three, tb (G) is identical with the modified cutwidth of G plus one and identical with the node search number of G. In addition, for any graph G, tb (G) is at least as large as the node search number of G and, consequently, at least as large as the search number of G minus one. The topological bandwidth of a graph can get arbitrarily far from these other graph parameters when there is no degree restriction placed on the graphs. However, F. R. K. Chung has shown [4] that, for all trees T, $\text{cw}(T) \leqq \text{tb}(T) + \log_2 \text{tb}(T) + 2$, where cw (T) denotes the cutwidth of T, and also that there are arbitrarily large trees T such that $\text{cw}(T) \geqq \text{tb}(T) + \log_2 \text{tb}(T) - 1$.

In § 3 we focus on the topological bandwidth of trees. A forbidden subtree characterization of binary trees with topological bandwidth k is given for all $k \geqq 1$. It is also shown that the topological bandwidth of a complete binary tree of height h is $\lceil h/2 \rceil$. An $O(n \log n)$ algorithm for computing the modified cutwidth of a binary tree has been described recently by Sudborough and Turner [27]. Since modified cutwidth plus one and topological bandwidth are identical for all degree 3 graphs, it follows immediately that this algorithm can be used to compute the topological bandwidth of any binary tree. This appears to be in sharp contrast with the bandwidth minimization problem, which is NP-hard even for binary trees [6].

In § 4 we give a proof that the problem of deciding, given a graph G and an integer k, whether tb $(G) \leqq k$ or not, is NP-complete. In fact, we show that the problem is NP-complete even when restricted to graphs with maximum vertex degree three. Thus, it follows that the problems of determining the modified cutwidth and the node search number of a graph with maximum vertex degree three is NP-hard. This improves the earlier NP-completeness results [12], [20], which produced graphs with arbitrarily large vertex degree. Furthermore, the same technique is used to show that the problem, given a graph G and an integer k, of determining if the cutwidth of G is at most k, called the *Min Cut Linear Arrangement* problem, is NP-complete even when restricted to graphs with maximum vertex degree three. It is known that, for graphs with maximum degree three, search number and cutwidth are the same [15]; consequently, the problem of determining the search number of a graph is NP-hard even when restricted to graphs with maximum vertex degree three. This improves the NP-completeness results [8], [16] written earlier, which produced graphs with arbitrarily large vertex degree.

Finally, in § 5 we characterize graphs with topological bandwidth two. The characterization given suggests a linear time algorithm to determine if a graph has topological bandwidth two. In [6] a linear time algorithm for the problem of deciding if a graph has bandwidth 2 was given; however, no characterization of graphs with bandwidth two is known. It is also noted that the dynamic programming techniques described in [9], [24] can be modified to yield an algorithm that runs in time $O(n^k)$ and decides whether a graph G with n vertices has topological bandwidth k or not, for each $k \geqq 3$.

We shall assume throughout that the graphs considered are connected. That is, since the topological bandwidth of a graph is identical to the largest topological bandwidth of any of its connected components, this assumption can be made without any loss of generality.

2. Relating topological bandwidth to search number and width. Let G be a finite undirected graph and let L be a linear layout of G. The *width of G at the ith gap (in L)*, denoted by wg (i), is the number of edges in the set $\{\{x, y\} | L(x) \leqq i \text{ and } L(y) > i\}$. The *width of G at the ith vertex (of L)*, denoted by $w(i)$, is the number of edges in the

set $\{\{x, y\} | L(x) < i$ and $L(y) > i\}$. The *cutwidth of G under L*, denoted by cw (G, L), is max $\{\text{wg}(i) | 1 \leqq i < |G|\}$. The *modified cutwidth of G under L*, denoted by mcw (G, L), is max $\{w(i) | 1 \leqq i \leqq |G|\}$. The *cutwidth of G*, denoted by cw (G), is min $\{\text{cw}(G, L) | L$ is a layout of $G\}$. The *modified cutwidth of G*, denoted by mcw (G), is min $\{\text{mcw}(G, L) | L$ is a layout of $G\}$. The problem of determining the cutwidth of a graph has application in VLSI [15], [19], [28], [29], [30]. It is NP-complete in the general case and can be done in polynomial time for any fixed degree tree [5], [8], [25]. The problem of determining the modified cutwidth of a graph has been studied in the field of register allocation [22] and pebble games [12].

The search number of a graph has also recently been investigated [10], [15], [16]. The search number of an undirected graph is the minimum number of searchers needed to guarantee finding a fugitive who is lurking about on the vertices and edges of the graph, where the fugitive is assumed to possess unlimited speed and complete knowledge about the movements of his pursuers. The reader should consult [14], [16] for additional details. Terminology from the literature on search number will be used in this paper. For example, an edge is said to be *clear* when the searchers have moved in such a way that the fugitive cannot possibly be on that edge; otherwise, the edge is *contaminated.* An edge is *re-contaminated* if it is contaminated and at some earlier step has been clear. It is known that allowing *re-contamination* does not reduce the number of searchers [10]. Thus, we may assume, without loss of generality, that once an edge becomes clear it remains clear during the remainder of the searching sequence.

In this section we show that the topological bandwidth of a graph is not greater than its modified cutwidth plus one. That is, for any graph G, $\text{tb}(G) \leqq \text{mcw}(G) + 1$. Since for any graph G, $\text{mcw}(G) + 1 \leqq \text{cw}(G)$, this shows that $\text{tb}(G) \leqq \text{cw}(G)$. We also show that, for any degree 3 graph G, $\text{mcw}(G) + 1 = \text{tb}(G)$. Furthermore, for any graph G, the search number of G is not greater than the topological bandwidth of G plus one.

Let $G = (V, E)$ be a finite undirected graph. A *partial layout* of G is a function L that maps a subset V' of the set of vertices V into the set of natural numbers $\{1, 2, \cdots, |V'|\}$. An edge $e = \{x, y\}$ is *dangling from the partial layout L*, or simply dangling, when the partial layout is understood, if x is in the domain of L and y is not. A vertex x is *active* (in a partial layout L) if it is incident to a dangling edge. An edge in $w(i)$ is said to *pass over* the ith vertex.

THEOREM 2.1. *For any graph G,* $\text{tb}(G) \leqq \text{mcw}(G) + 1$.

Proof. Let $G = (V, E)$ be a graph and let L be a linear layout of G such that $\text{mcw}(G, L) = k$. We construct a homeomorphic image G' of G and simultaneously a linear layout L' of G' such that $b(G', L') \leqq k + 1$. This is done by the following procedure:

1. Let the first vertex of G' be the first vertex of G under the layout L and set an auxiliary variable i to 2. (All edges of G' are initially unmarked.)
2. If there are no more dangling edges in the partial layout of G' constructed so far, then stop; otherwise, let x be the lowest numbered active vertex of this partial layout. If all dangling edges incident to x are marked, then go to (4); otherwise, let e be an arbitrary unmarked edge incident to x and go to (3).
3. If e is incident to the ith vertex of G, then go to (4); otherwise, go to (5).
4. Make the next vertex of G' the ith vertex of G, add one to i, unmark all edges of G', and go to (2).
5. Make the next vertex of G' a new degree 2 vertex incident to e and a new edge e'. The edge e' is also made incident to the as yet unnumbered vertex that e used to be incident to. (That is, the new degree 2 vertex is inserted into the old edge e.) Mark the new edge e' and go to (2).

It is straightforward to verify that the above procedure produces a homeomorphic image G' of G and a layout L' of G' such that: (a) for all vertices x and y in G, if $L(x) < L(y)$, then $L'(x) < L'(y)$, and (b) for all vertices x in G, the edges passing over x in the layout L are the same as the edges passing over x in the layout L', with degree 2 vertices possibly having been added to these edges in G'. We need to show that the algorithm terminates with a layout L' such that $b(G', L') \leqq k+1$. For this purpose, consider any partial layout I obtained by restricting the layout L' to the first p vertices of G', for some $p \geqq 1$. Let $D(I) = \{e_1, e_2, \cdots, e_m\}$ be the set of dangling edges in this partial layout. By showing that every one of the edges in $D(I)$ is made incident to one of the next $k+1$ vertices, we show that G' has bandwidth at most $k+1$ under L'.

Let $y_1, y_2, \cdots, y_m$ be the vertices of G' incident to the edges in $D(I)$. If none of these are vertices in the original graph G, then there can be at most k dangling edges, because all of these vertices would be added degree 2 vertices having a new dangling edge not incident to the next vertex in the original graph G. So there would be more than k edges passing over this vertex. It should be noted that, because of the marking of edges and the fact that the procedure does not assign new degree 2 vertices to marked dangling edges, the algorithm eventually assigns the next vertex of G an integer. So there cannot be more than k dangling edges not incident to the next vertex of G. Again after degree 2 vertices are assigned to each dangling edge the next vertex of G would be assigned an integer and there would be more than k edges passing over it.

Assume now that y_i is a vertex in G and that $y_1, \cdots, y_{i-1}$ are added degree two vertices. Consider the partial layout I' obtained by extending the partial layout I to the vertices $y_1, y_2, \cdots, y_i$. There must be a dangling edge incident to each of the vertices $y_1, \cdots, y_{i-1}$, since these are added degree two vertices. Furthermore, each of these dangling edges passes over the vertex y_i. Thus, there can be at most $k-i+1$ additional edges passing over y_i, since at most k edges pass over every vertex in the original graph G in the layout L'. This means that at most $k-i+1$ edges in the set $D(I)$ remain dangling in the partial layout I'. Since the procedure assigns vertices for dangling edges incident upon the lowest numbered vertices of G' first, all of the remaining dangling edges in $D(I)$ are made incident to one of the next $k-i+1$ vertices. Thus, we see that all edges in $D(I)$ are made incident to one of the next $k+1$ vertices in G' under L'. Since this is true for every partial layout I of L', it follows immediately that $b(G', L') \leqq k+1$. Therefore, since G has a homeomorphic image G' with bandwidth $k+1$, the topological bandwidth of G is at most $k+1$. □

COROLLARY 2.1. *For any graph* G, $\mathrm{tb}(G) \leqq \mathrm{cw}(G)$.

The corollary follows from the straightforward observation that, for any graph G, $\mathrm{mcw}(G) \leqq \mathrm{cw}(G) - 1$. Theorem 2.1 is an improvement of the result indicated in Corollary 2.1, which has been independently described by F. R. K. Chung [4] and the first author [14].

We are interested now in the converse of Theorem 2.1. That is, for which graphs G is it true that $\mathrm{mcw}(G) + 1 = \mathrm{tb}(G)$. We show that this is true for all degree 3 graphs. It is not true for all degree 4 graphs. For example, the complete graph of five vertices, C_5, has topological bandwidth 4 and modified cutwidth 4.

As a tool for showing that, for all degree 3 graphs G, $\mathrm{mcw}(G) + 1 \leqq \mathrm{tb}(G)$, we consider the following modified searching problem on an undirected graph, called *node searching*. The rules for node searching a graph G are:

1. No vertex of G has a searcher placed on it more than once.
2. A searcher can be added to an unvisited vertex at any time.
3. A searcher can be deleted from a vertex x provided that all neighbors of x have been visited by searchers.

4. A searcher can be deleted from a vertex x and placed on a neighboring unvisited vertex y provided that all other neighbors of x have been visited by searchers.

In [23] this game was called "breadth first pebbling" and it was shown that the length of time searchers need to stay on vertices of the graph is identical to the bandwidth of the graph. Node searching has also recently been investigated in [20], where the number of searchers, or node search number, was considered. The goal of node searching is the same as the usual game of searching a graph (which can be referred to as *edge searching*): to guarantee finding a fugitive who is hiding somewhere on the edges of the graph. In node searching the fugitive is caught if he is on an edge with searchers on both ends, while in edge searching a searcher must actually be moved through the edge. We use the same terminology as in the edge searching game. An edge $\{x, y\}$ is *clear* if either two searchers have been placed on opposite ends of the edge or a searcher has been placed on one end and shifted, as in rule (4) above, to the other end. An uncleared edge is *contaminated.* It is straightforward to observe that after a play of the node searching game in which every vertex is visited and all searchers are removed that all edges are clear. This follows simply from the fact that a searcher cannot be removed until all edges incident to the vertex where it sits are clear. The *node search number of* G, denoted by $\mathrm{ns}(G)$, is the minimum number of searchers needed in the node searching game to clear all of the edges of G, i.e. to guarantee capturing the fugitive. It is straightforward to verify that, for every graph G, $s(G) \leqq \mathrm{ns}(G)+1$, since if an edge is cleared in the node searching game by placing two searchers at either end, then it can be cleared in the edge searching game with the same two searchers on the endpoints and an extra searcher that moves through the edge. Furthermore, any edge searching sequence without re-contamination is also a node searching sequence, so $\mathrm{ns}(G) \leqq s(G) \leqq \mathrm{ns}(G)+1$, for any graph G. We note that, for any graph G, any homeomorphic image of G has node search number at least as large as the node search number of G.

LEMMA 2.1. *For any graphs* G, $\mathrm{ns}(G) \leqq \mathrm{tb}(G)$.

Proof. Let G be a graph. Let G' be a homeomorphic image of G and let L be a layout of G' such that $b(G', L) = k$. We show that $\mathrm{ns}(G') \leqq k$ and, since $\mathrm{ns}(G) \leqq \mathrm{ns}(G')$, this means that $\mathrm{ns}(G) \leqq k$. The node searching sequence is the following:

(a) Place a searcher on the first k vertices of G'.

(b) While there is still a vertex to be visited do the following:
if there is an edge connecting the lowest numbered vertex containing a searcher to an unvisited vertex, then shift the searcher from this vertex to its unvisited neighbor; otherwise, remove the searcher from this vertex and install it on the lowest numbered unvisited vertex of G'.

It is straightforward to verify that the sequence of steps described above is a valid node searching sequence and clears all the edges of G'. Furthermore, the number of searchers is never greater than k. When the process above is finished, all of the searchers are removed. Thus, we have shown G' can be searched in the node search game with k searchers. □

COROLLARY 2.2. *For any graph* G, $s(G) \leqq \mathrm{tb}(G)+1$.

The corollary follows immediately from the earlier observed fact that $s(G) \leqq \mathrm{ns}(G)+1$.

LEMMA 2.2. *For any graph* G *with maximum vertex degree* 3, $\mathrm{mcw}(G) \leqq \mathrm{ns}(G)-1$.

Proof. Let G be a graph such that every vertex has degree at most 3. Let S be a node searching sequence that clears all of the edges of G and uses k searchers. We describe a layout L_S such that $\mathrm{mcw}(G, L_S) \leqq k-1$.

First, define a function f_S, which maps vertices of G to natural numbers, by:

$f_S(x) = i$ if and only if i is the smallest integer such that after step i of S the vertex x has been visited by a searcher and at most one edge incident to it is contaminated.

Observe that more than one vertex can reach the condition stated above at the same time. For example, consider the graph shown in Fig. 2.1. Suppose searchers have been placed on vertices A, B, E, F, G, and H. So edges $\{A, B\}$, $\{E, F\}$, and $\{G, H\}$ are clear. By placing a searcher on vertex D one clears edges $\{B, D\}$, $\{D, E\}$, and $\{H, D\}$. Consequently, all four vertices B, D, E, and H satisfy the indicated condition simultaneously.

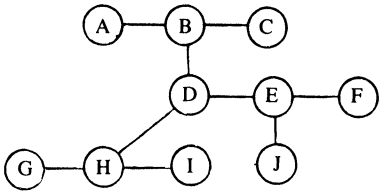


FIG. 2.1. *Adding a searcher to vertex D causes vertices B, D, E, and H to have a majority of their incident edges cleared, when vertices A, B, E, F, G, and H have searchers.*

So the function f_S is not, in general, a layout. It can map more than one vertex to the same integer. Note that in any single step all edges cleared are incident to the same vertex. Let v be a vertex of G such that $f_S(v) = i$. If w_1, w_2, and w_3 are neighbors of v and $f_S(w_1) = f_S(w_2) = f_S(w_3) = i$, then create a layout L_S by assigning v, w_1, w_2, and w_3 to consecutive integers. In particular, let one neighbor of v be assigned an integer larger than that assigned to v and one neighbor of v be assigned an integer smaller than that assigned to v. Thus, in L_S the vertices are laid out in the manner shown in Fig. 2.2. If there is only one neighbor of v that satisfies the indicated property simultaneously with v, then in the layout L_S the relative position of v and its neighbor are chosen arbitrarily.

In this way we arrive at a layout L_S of G such that: (1) if $f_S(x) < f_S(y)$, then $L_S(x) < L_S(y)$ and (2) if $f_S(x) = f_S(y)$, then the relative order of x and y is chosen in a manner consistent with that indicated above. We show that mcw $(G, L_S) \leqq k - 1$. That is, we show that, for all i $(1 \leqq i \leqq |G|)$, the set of edges $E(i) = \{\{x, y\} | L_S(x) < i$ and $L_S(y) > i\}$ has at most $k - 1$ elements. This is accomplished by demonstrating a unique searcher, say $s(e)$, for each edge e in $E(i)$ that is positioned at time t_i on a vertex incident to e, where t_i is the step in the sequence S when the ith vertex of G under the layout L_S first satisfies the indicated property.

Let $e = \{x, y\}$ be an edge in $E(i)$ that is contaminated after step t_i. Since e is in $E(i)$, $L_S(x) < i$. Since $L_S(x) < i$, x has been visited by a searcher and at most one edge incident to x is contaminated after step t_i. Consequently, there is a searcher on vertex x after step t_i. This searcher is needed to separate cleared edges from contaminated ones or, if x has degree one, the searcher can only be removed when x's contaminated

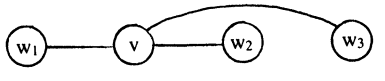


FIG. 2.2. *An arrangement provided by the layout L_S for a vertex v and its neighbors that are mapped to the same integer by f_S.*

edge is cleared. Let the searcher on x be $s(e)$. Vertex x cannot be incident to any other edge that is contaminated at time t_i, since x satisfies the indicated condition by step t_i. So, the searcher $s(e)$ is not located at the end of any other contaminated edge; it is unique for edge e.

Now, let $e=\{x, y\}$ be an edge in $E(i)$ that is clear after step t_i. Since e is in $E(i)$, $L_S(y) > i$. Since $L_S(y) > i$, it follows that at least two edges incident to y were contaminated before step t_i. There are two cases: (1) at least two edges incident to y are still contaminated after step t_i or (2) one edge incident to y is contaminated after step t_i.

Case 1. In this case vertex y has two contaminated edges after step t_i. Consequently, there must be a searcher on vertex y after step t_i in order to separate the contaminated edges from the cleared edge e. Let this searcher be $s(e)$. Vertex y cannot be incident to any edge other than e that is clear after step t_i, since y has degree at most three. Furthermore, $s(e)$ is located after step t_i on a vertex to the right of vertex i, so it cannot be the same as any searcher assigned to contaminated edges in $E(i)$ which are located to the left of vertex i at this time. So $s(e)$ is unique for edge e.

Case 2. In this case one edge incident to y is contaminated after step t_i. So the indicated property must be true for vertex y after step t_i. However, since $L_S(y) > i$, it follows that more than one vertex satisfies the property simultaneously. If e is the only edge in $E(i)$ incident to y that is clear after step t_i, then we can assign the searcher on vertex y uniquely to edge e. However, it is possible that there is another edge, say e', that is incident to y and is clear after step t_i. One of these edges, say e', must connect y to the vertex, say z, that has a searcher placed on it during step t_i and which satisfies the indicated property for the first time. It is only possible for e' to be in $E(i)$ if i is the position of a vertex, say w, positioned between y and z such that w also satisfies the indicated property for the first time at step t_i. For example, let z, w, and y be the last three vertices pictured in Fig. 2.2. Because of the requirements of our layout L_S there must be another neighbor of vertex z assigned a position to its left which also satisfies the indicated property for the first time at step t_i. It follows that all three edges incident to z are cleared during step t_i. Let the searcher on z be designated as $s(e')$ and the searcher on y be designated as $s(e)$. We then have unique searchers for both edges e and e'. Furthermore, even though $L_S(z) < i$ there are no contaminated edges incident to z after step t_i and, consequently, the searcher $s(e')$ on z is not one of those assigned earlier for contaminated edges.

We have described, for all i, a unique searcher $s(e)$ for each edge e in $E(i)$. Each searcher $s(e)$ sits at time t_i on a vertex incident to edge e. No edge in $E(i)$, by definition, is incident to the ith vertex of G under L_S. As the ith vertex satisfies the indicated property for the first time at step t_i it must contain a searcher after step t_i. Note that if a searcher is moved from this vertex through an edge at step t_i there must be another searcher that remains. Because searchers can only be shifted through an edge when all other incident edges are clear, this vertex must have two contaminated edges before step t_i. So, one of the k searchers used in the search sequence S must be on the ith vertex after step t_i. Consequently, there are at most $k-1$ searchers to be associated with edges in $E(i)$, for all i, and, as each edge gets a unique searcher, there are at most $k-1$ edges in $E(i)$. □

Lemma 2.2 can be extended to show that, for any graph G, $\text{mcw}(G) \leqq \lfloor \deg(G)/2 \rfloor \cdot \text{ns}(G) - 1$. We will not give the details here. A similar statement is known for the relationship between the (edge) search number of a graph and its cutwidth [15].

THEOREM 2.2. *For any degree* 3 *graph* G, $\text{mcw}(G)+1 = \text{tb}(G)$.

The theorem follows immediately from Lemmas 2.1-2.2 and Theorem 2.1. The next theorem follows in a similar way from the same lemmas and theorem.

THEOREM 2.3. *For any degree* 3 *graph* G, $\mathrm{tb}(G) = \mathrm{ns}(G)$.

We note that the topological bandwidth of a graph G and its node search number and its modified cutwidth can get arbitrarily far apart without any degree restriction. For example, the topological bandwidth of the complete graph with n vertices is $n-1$, while its modified cutwidth is $\lfloor (n-1)/2 \rfloor \cdot \lceil (n-1)/2 \rceil$. The topological bandwidth of the "k-star," the tree with $k+1$ vertices and k leaves, is $\lceil k/2 \rceil$, while its node search number is 2. It should also be noted that the trees constructed by F. R. K. Chung [4] to exhibit, for each $n \geqq 1$, trees with topological bandwidth n and cutwidth at least $n + \log_2 n - 1$ are sufficient to exhibit that topological bandwidth and modified cutwidth can get arbitrarily far apart even in trees.

By Corollary 2.1 we have that, for any graph G, $\mathrm{tb}(G) \leqq \mathrm{cw}(G)$. By Corollary 2.2 we have that, for all graphs G, $s(G) - 1 \leqq \mathrm{tb}(G)$. Consequently, the topological bandwidth of a graph is always between these two quantities: search number minus one and cutwidth. In [15] it was show that search number and cutwidth are identical for degree 3 graphs. Thus, for any degree 3 graph G, $\mathrm{cw}(G) - 1 \leqq \mathrm{tb}(G) \leqq \mathrm{cw}(G)$.

3. The topological bandwidth of a tree. In this section we describe a forbidden subtree characterization for degree three trees with topological bandwidth k, for each $k \geqq 1$. Although we do not explicitly give it here, there is an $O(n \log n)$ algorithm to compute the topological bandwidth of an arbitrary binary tree. This follows from the description of such an algorithm for the modified cutwidth of trees [27], since we have shown that, for all degree 3 graphs G, $\mathrm{mcw}(G) + 1 = \mathrm{tb}(G)$. This stands in sharp contrast with the bandwidth minimization problem which is NP-hard even for binary trees [6].

Let T be a tree and let x and $y_1, y_2, \cdots, y_m$ be arbitrary vertices in T. The tree $T[x, y_1, y_2, \cdots, y_m]$ is the largest subtree of T containing the vertex x but not containing any of the vertices $y_1, y_2, \cdots, y_m$.

THEOREM 3.1. *Let T be a tree with maximum vertex degree three. Then* $\mathrm{tb}(T) \leqq k$ *if and only if for every vertex x of degree three in T one can choose two neighbors y_1 and y_2, leaving the neighboring vertex y_3, such that*:

(a) $\mathrm{tb}(T[x, y_2, y_3]) \leqq k$,

(b) $\mathrm{tb}(T[x, y_1, y_3]) \leqq k$, *and*

(c) $\mathrm{tb}(T[y_3, x]) \leqq k - 1$.

Proof. Let T be a degree three tree such that $\mathrm{tb}(T) \leqq k$. By definition there is a tree T' which is a homeomorphic image of T such that $b(T') \leqq k$. Let x be a degree 3 vertex in T'. Since x has degree 3 it must be a vertex also in the original tree T. Let A and B be the vertices in T' that are assigned to the first and last integers under a layout L such that $b(T', L) \leqq k$. Let z_1, z_2, and z_3 be the neighbors of x. The vertex A must be in at least one of the subtrees $T'[x, z_1, z_2]$, $T'[x, z_1, z_3]$, and $T'[x, z_2, z_3]$. Also, the vertex B must be in at least one of these three subtrees. It follows that in deleting two of these subtrees and the vertex x one deletes a path that connects the leftmost vertex A with the rightmost vertex B. Since one deletes all the nodes of a path connecting the leftmost vertex A with the rightmost vertex B, the remaining subtree of T' must have bandwidth at most $k-1$. That is, in general, if one deletes at least one vertex from every block of k consecutive vertices in a layout with bandwidth k, then one obtains a graph with bandwidth at most $k-1$.

The vertices z_1, z_2, and z_3, which are adjacent to x in T', need not be vertices in the original tree T. Let y_1, y_2, and y_3 be vertices in the original tree T such that z_1, z_2, and z_3, respectively, are added to the edges $\{x, y_1\}$, $\{x, y_2\}$, and $\{x, y_3\}$. Let y_1 and y_2 be the two of these three vertices contained in the two subtrees deleted in deleting a

path from A to B. It follows that $T[y_3, x]$ has topological bandwidth at most $k-1$ since the remaining subtree $T'[z_3, x]$ of T', obtained by deleting the subtrees $T'[x, z_1, z_3]$ and $T'[x, z_2, z_3]$ and the vertex x, which we have seen has bandwidth at most $k-1$, is a homeomorphic image of $T[y_3, x]$. Furthermore, the subtrees $T[x, y_1, y_3]$ and $T[x, y_2, y_3]$ have topological bandwidth at most k, since they are subtrees of T. So, we have shown that the stated properties are true for vertex x. Since the vertex x was chosen arbitrarily, the required properties are, in fact, true for every degree 3 vertex in T.

Conversely, let T be a tree with maximum vertex degree 3 such that, for every vertex x having degree 3, one can choose two neighbors, y_1 and y_2, leaving a remaining neighbor y_3, such that: (a) $\text{tb}(T[x, y_2, y_3]) \leqq k$, (b) $\text{tb}(T[x, y_1, y_3]) \leqq k$, and (c) $\text{tb}(T[y_3, x]) \leqq k-1$. It will be shown that T has topological bandwidth at most k. The basic idea in the proof is to show that there exists a chain C of vertices $x_1, x_2, \cdots, x_k$ (for some $k \geqq 1$) such that, for all i $(1 \leqq i \leqq k)$, if y_i is a neighbor of x_i and $y_i \notin C$, then $\text{tb}(T[y_1, x_i]) \leqq k-1$.

We show now that such a chain C exists. Color a vertex x in T red if, for every neighbor y of x, $\text{tb}(T[x, y]) \geqq k$. We observe first that no vertex of T can have three red neighbors. That is, the hypothesis, condition (c), asserts that, for every vertex x of T, there exists a neighbor y, such that $\text{tb}(T[y, x]) \leqq k-1$. So, the vertex y cannot be colored red and, therefore, every vertex in T has at least one neighbor that is not colored red. We observe next that, if x has two neighbors, y_1 and y_2, colored red, then x must itself be colored red. That is, if x were not colored red, then there would exist a neighbor, say y, of x such that $\text{tb}(T[x, y]) \leqq k-1$. However, either $T[y_1, x]$ or $T[y_2, x]$ is a subtree of $T[x, y]$ and, consequently, at least one of these trees would then have topological bandwidth at most $k-1$. This contradicts the fact that both y_1 and y_2 are colored red.

It follows that the red vertices of T form a chain. This chain satisfies the required condition. That is, let y be a nonred neighbor of a red vertex x. Since y is not colored red, there exists a neighbor z of y such that $\text{tb}(T[y, z]) \leqq k-1$. In fact, the vertex z must be x, since otherwise the tree $T[x, y]$ would be a subtree of the tree $T[y, z]$, implying that $T[x, y]$ has topological bandwidth at most $k-1$. This contradicts the fact that x is colored red. Thus, if y is a neighbor of a red vertex and y is not red, then $\text{tb}(T[y, x]) \leqq k-1$.

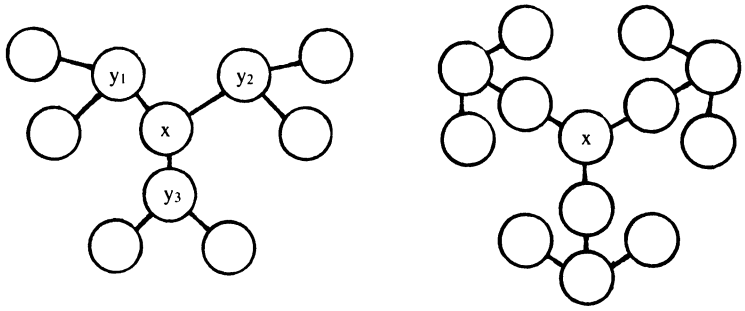
Let C be the chain of red vertices. Extend the chain C so that the beginning and end of the new chain are leaves of T. In general there is more than one way to extend the chain C; however, one can choose the extension in an arbitrary manner. Let the new chain be $C' = x_1, x_2, \cdots, m$ (for some $m \geqq 1$). Each vertex x_i $(1 \leqq i \leqq m)$ has at most one neighbor y_i. Furthermore, as we have shown, if x_i has a neighbor y_i, then $\text{tb}(T[y_i, x_i]) \leqq k-1$. For all i $(1 \leqq i \leqq m)$ such that x_i has a neighbor y_i, let L_i be a linear layout of a homeomorphic image T_i of $T[y_i, x_i]$ such $b(T_i, L_i) \leqq k-1$. Consider a layout L of the tree T such that:

(1) for all i, j $(1 \leqq i < j \leqq m)$, if x is in T_i and y is in T_j, then $L(x) < L(y)$,
(2) for all i $(1 \leqq i \leqq m)$, if $L_i(x) < L_i(y)$, then $L(x) < L(y)$,
(3) for all i $(1 \leqq i \leqq m)$, if x_i has a neighbor y_i, then $|L(x_i) - L(y_i)| \leqq 1$, and,
(4) for all i $(1 \leqq i < m)$, $L(x_i) < L(x_{i+1})$.

The only edges in T which can have length greater than $k-1$ under the layout L are the edges $\{x_i, x_{i+1}\}$, which connect successive vertices of the chain. However, by adding degree 2 vertices to these edges and assigning these degree 2 vertices to positions that are k apart in an expanded layout, one arrives at a layout of a homeomorphic image of T which has bandwidth k. Therefore, the tree T has topological bandwith at most k. □

Example 3.1. The tree shown in Fig. 3.1(a) has topological bandwidth 2. Clearly it cannot have topological bandwidth one, since it contains a degree 3 vertex. It has topological bandwidth at most 2, by Theorem 3.1, since every degree three vertex x in this tree has neighbors y_1, y_2, y_3 such that: (a) $\text{tb}(T[x, y_1, y_3]) \leqq 2$, (b) $\text{tb}(T[x, y_2, y_3]) \leqq 2$, and (c) $\text{tb}(T[y_3, x]) \leqq 1$. For example, consider the vertex x shown in Fig. 3.1(a). Choose y_1, y_2, and y_3 to be the neighbors of x in the manner shown. Then, $\text{tb}(T[x, y_1, y_3]) = 2$, $\text{tb}(T[x, y_2, y_3]) = 2$, and $\text{tb}(T[y_3, x]) = 1$.

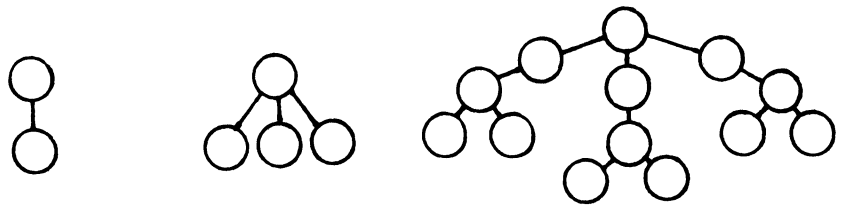
The tree shown in Fig. 3.1(b) has topological bandwidth 3. That is, it cannot have topological bandwidth 2, by Theorem 3.1, since there exists a vertex x, shown in the figure, such that for any neighbor y of x, $\text{tb}(T[y, x]) = 2$.



(a) (b)

FIG. 3.1. (a) *A tree with topological bandwidth* 2. (b) *A tree with topological bandwidth* 3.

Let $T_3(k)$ denote the set of smallest trees having maximum vertex degree 3 and topological bandwidth k. The sets $T_3(1)$, $T_3(2)$, and $T_3(3)$ have only one element. These sets are described in Fig. 3.2(a), (b), and (c), respectively. In general, the set $T_3(k+1)$ is formed by taking three (not necessarily distinct) trees from $T_3(k)$, creating a new vertex, say x, and joining x by an edge to a degree one or a degree two vertex in each of the three trees. To illustrate this process and to show that, in general, the sets $T_3(i)$ contain more than one tree, we describe two of the trees in $T_3(4)$ in Fig. 3.3.



(a) (b) (c)

FIG. 3.2. *The smallest degree* 3 *trees with*: (a) *topological bandwidth* 1, (b) *topological bandwidth* 2, *and* (c) *topological bandwidth* 3.

We need to show the correctness of our construction of the sets $T_3(k)$, for $k \geqq 2$. This can be done by induction on k. For the basis step we observe that the tree shown in Fig. 3.2(b) is the unique tree having topological bandwidth 2 and having the smallest number of vertices. Assume that the set $T_3(k)$ constructed is correct. That is, the set $T_3(k)$ contains exactly those trees which have the minimum number of vertices, have maximum vertex degree three, and have topological bandwidth k. Let T be a tree that is placed in $T_3(k+1)$. That is, T is a tree that is formed by taking a new vertex x and three trees (not necessarily distinct) from $T_3(k)$ and joining x by an edge to a degree

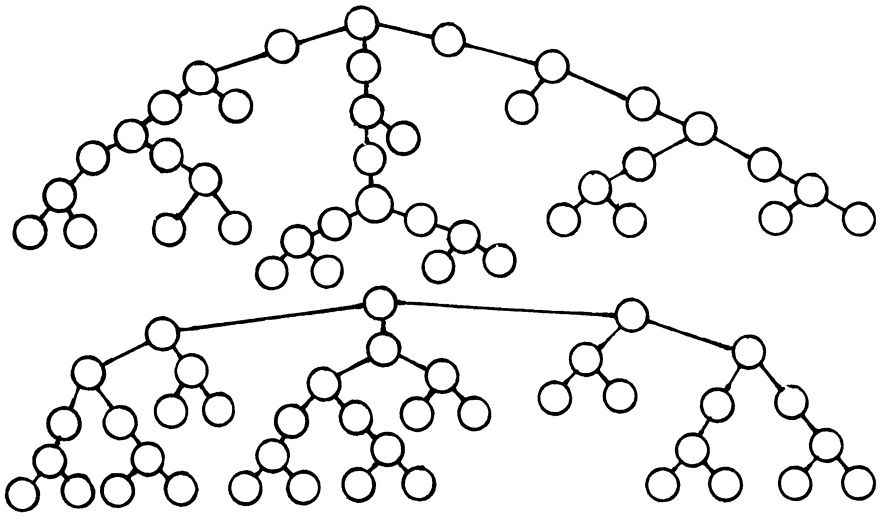


FIG. 3.3. *Two trees in the set* $T_3(4)$.

one or a degree two vertex in each of the three trees. It follows from Theorem 3.1 that the constructed tree T has topological bandwidth at least $k+1$, since for every one of the neighbors y of the new vertex x, the tree $T[y, x]$ is in $T_3(k)$ and, therefore, has topological bandwidth k. In fact, T has topological bandwidth exactly $k+1$. That T has topological bandwidth no greater than $k+1$ can be shown as follows. Each of the trees in $T_3(k)$ has topological bandwidth k, so there is a homeomorphic image of each of these trees which has bandwidth k. One can lay out these homeomorphic images so that their vertices are completely separated. That is, all vertices of one tree are placed either before or after all the vertices of any other tree. Make the layout so the bandwidth is still k. Now one adds the vertex x to a position adjacent to the vertex it is joined to in the middle tree of this layout. The vertex x is then joined to the appropriate vertices in the other two trees by adding into these edges vertices that are placed into positions $k+1$ apart in the global layout. It follows that the result is a homeomorphic image of the tree T with bandwidth $k+1$. So, T has topological bandwidth at most $k+1$. Furthermore, any tree T' with topological bandwidth greater than k, by Theorem 3.1, must have at least one vertex x with three neighbors y_1, y_2, and y_3 such that, for all i $(1 \leqq i \leqq 3)$, tb $(T'[y_i, x]) \leqq k$. Since each of the three trees $T'[y_1, x]$, $T'[y_2, x]$, and $T'[y_3, x]$ have topological bandwidth at least k, they must have at least as many vertices as the trees in $T_3(k)$. Therefore, the tree T' has at least as many vertices as the tree T we constructed.

On the other hand, let T be a tree which has the minimal number of vertices of any tree having topological bandwidth $k+1$ and maximum vertex degree 3. By Theorem 3.1, since T does not have topological bandwidth k, there is a vertex x with three neighbors y_1, y_2, and y_3 such that tb $(T[y_i, x]) \geqq k$, for all i $(1 \leqq i \leqq 3)$. Since T has the smallest number of vertices, it follows that each of the trees $T[y_1, x]$, $T[y_2, x]$, and $T[y_3, x]$ are in $T_3(k)$. For otherwise, we could form a smaller tree that still had topological bandwidth $k+1$ by replacing whichever tree is not in $T_3(k)$ by one that is in $T_3(k)$. So, it follows that the tree T is one of the trees we construct in the process described. Thus, $T_3(k)$, for all k, is the correct set.

Let $n_3(k)$ denote the number of vertices in the smallest degree 3 tree with topological bandwidth k. It follows from the construction of the sets $T_3(k)$, for $k \geqq 1$,

that $n_3(1)=2$, $n_3(2)=4$, and, for $k\geq 2$, $n_3(k+1)=3\cdot n_3(k)+1$. Solving this recurrence we see that, for $k\geq 2$, $n_3(k)=\frac{1}{2}(3^k-1)$. Therefore, a tree with $n\geq 4$ vertices can have topological bandwidth at most $\lfloor \log_3(2n+1)\rfloor$.

Let $\mathcal{F}$ be a set of trees, each tree in the set having maximum vertex degree 3. The set $\mathcal{H}(\mathcal{F})$ denotes the set consisting of all the trees in $\mathcal{F}$ together with all trees that can be obtained by inserting a single degree 2 vertex into an edge connecting two degree 3 vertices in a tree from $\mathcal{F}$. Define the set of trees $\mathcal{F}_i$, for $i\geq 1$, by: $\mathcal{F}_1=T_3(1)$, $\mathcal{F}_2=T_3(2)$, and, for all $k\geq 2$, $\mathcal{F}_{k+1}$ is the set of all trees that can be formed by taking three trees from $\mathcal{H}(\mathcal{F}_k)$, creating a new vertex x, and joining x by an edge to a degree one or a degree 2 vertex in each of the three trees. It should be noted that $\mathcal{F}_3=T_3(3)$, $\mathcal{F}_4=T_3(4)$, but $\mathcal{F}_5\neq T_3(5)$. That is, there are trees in $\mathcal{F}_5$ that are not in $T_3(5)$. This can be seen by observing that there are edges in one of the trees shown in Fig. 3.3 that connect degree 3 vertices. Hence, there are trees in $\mathcal{H}(T_3(4))$ that are not in $T_3(4)$ and, consequently, trees in $\mathcal{F}_5$ that are not in $T_3(5)$.

THEOREM 3.2. *Let T be a tree with maximum vertex degree* 3. $\text{tb}(T)\geq k$ *if and only if T contains a subtree that is a homeomorphic image of a tree in $\mathcal{F}_k$.*

Proof. It is straightforward to show that all trees in $\mathcal{F}_k$ have topological bandwidth at least k. That is, it follows from the construction of these sets and an argument similar to that used in proving the correctness of the construction of the sets $T_3(k)$, for all $k\geq 1$. Therefore, if T contains a subtree that is a homeomorphic image of a tree in $\mathcal{F}_k$, then T must have topological bandwidth at least k.

We show now by induction on k that, if $\text{tb}(T)\geq k$, then T contains a subtree that is a homeomorphic image of a tree in $\mathcal{F}_k$. Clearly, the statement is true for the basis step. That is, any tree with bandwidth one must contain two vertices connected by an edge and any tree with bandwidth two must contain a vertex of degree three. Assume now for the induction hypothesis that, if $\text{tb}(T)\geq k$, then T contains a subtree that is a homeomorphic image of a tree in $\mathcal{F}_k$. We show that the statement is true for $k+1$. Let T be a degree three tree such that $\text{tb}(T)\geq k+1$. By Theorem 3.1 it follows that there exists a vertex x with three neighbors y_1, y_2, and y_3 such that, for all i $(1\leq i\leq 3)$, $\text{tb}(T[y_i, x])\geq k$. By the inductive hypothesis, each of the three trees $T[y_1, x]$, $T[y_2, x]$, and $T[y_3, x]$ contains a subtree that is a homeomorphic image of a tree in $\mathcal{F}_k$. There must be a path from x to a degree one or a degree two vertex in each of the three homeomorphic images of these trees. Let $T[y_1, x]$, $T[y_2, x]$, and $T[y_3, x]$ contain homeomorphic images, say T_1^h, T_2^h, and T_3^h, of the trees T_1, T_2, and T_3 in $\mathcal{F}_k$, respectively. Let z_1, z_2, z_3 be the first vertices in T_1^h, T_2^h, and T_3^h, respectively, that lie in the path from the vertex x. We construct a tree U in $\mathcal{F}_{k+1}$ and demonstrate that a subtree of T is a homeomorphic image of U. For each i $(1\leq i\leq 3)$, if z_i is a degree two vertex that has been inserted into an edge e connecting two degree 3 vertices of T_i, then let the tree T_i' be that tree in $\mathcal{H}(\mathcal{F}_k)$ that is obtained by adding a degree two vertex, say z_i' to the edge e; otherwise, if z_i is a vertex in the original tree T_i, then let $T_i'=T_i$ and $z_i'=z_i$ and, if z_i is a degree two vertex that has been inserted into an edge incident to a degree two vertex, say v_i, then let $T_i'=T_i$ and $z_i'=v_i$. We note that, for all i $(1\leq i\leq 3)$, T_i' is a tree in $\mathcal{H}(\mathcal{F}_k)$ and z_i' has degree at most two. Construct the tree U by creating a new vertex w and connecting w by an edge to each of the three vertices z_1', z_2', and z_3'. Clearly, U is in $\mathcal{F}_{k+1}$ by construction. We show that T contains a subtree that is a homeomorphic image of U. Consider the subtree obtained by deleting all of the nodes from T except those in T_1^h, T_2^h, and T_3^h and those on the paths connecting x with z_1, z_2, and z_3 (including the vertex x). This subtree is a homeomorphic image of U. Note that T_1^h, T_2^h, and T_3^h are homeomorphic images of T_1', T_2', and T_3', respectively, and all of the vertices in the paths connecting x with z_1, z_2, and z_3 are degree two

vertices that can be added into the edges $\{x, z_1'\}$, $\{x, z_2'\}$, $\{x, z_3'\}$ *of* U. Therefore, T has a subtree that is a homeomorphic image of a tree in $\mathcal{F}_{k+1}$. □

COROLLARY 3.1. *Let T be a tree with maximum vertex degree* 3. tb $(T) = k$ *if and only if T contains a subtree that is a homeomorphic image of a tree in $\mathcal{F}_k$ and does not contain a subtree that is a homeomorphic image of a tree in $\mathcal{F}_{k+1}$.*

The corollary follows imediately from Theorem 3.2 and the definition of topological bandwidth. It should perhaps also be explicitly remarked that Corollary 3.1 is still valid when "mcw $(T)+1$" replaces "tb (T)," since we have shown in Theorem 2.2 that these two quantities are the same for all degree 3 graphs. Furthermore, in [27] an $O(n \log n)$ algorithm has been given for determining the modified cutwidth of a degree 3 tree and, consequently, it follows that the topological bandwidth of a degree 3 tree can be found in $O(n \log n)$ steps.

Determining the topological bandwidth of trees with degree greater than three would seem to be a more difficult problem. We note that the recursive statement made in Theorem 3.1 is not valid for trees with degree greater than 3. Basically the problem in making a similar statement for trees with degree greater than 3 is that, if x is a vertex with neighbors $y_1, y_2, \cdots, y_d$ $(d \geqq 4)$, then when x is deleted one is left with a forest and when two of the trees in this forest are thrown away (as we can consider happening in the statement of Theorem 3.1 for the degree 3 case) one still has a forest of at least two trees. The resulting disconnection of the tree is the basic problem in trying to formulate a similar statement for general trees.

On the other hand, one can make a somewhat similar statement for general trees. Let G be a graph and L a linear layout of G. Suppose there is a function f that assigns each vertex of G to an integer. We denote the graph G together with this function f by $G(f)$. Define the bandwidth of $G(f)$ under the layout L, denoted by $b(G(f), L)$, by:

$$b(G(f), L) = \max\left\{1 + \sum_{i=j+1}^{k-1} f(i) \,\middle|\, \{L^{-1}(j), L^{-1}(k)\} \text{ is an edge in } G\right\}.$$

It is straightforward to observe that the bandwidth of a graph G, as we have defined earlier, is the bandwidth of $G(f)$, where f is the function that maps each vertex to one. A statement similar to that made in Theorem 3.1 which is valid for all trees is the following. A tree T has topological bandwidth k if and only if for every vertex x in T with degree $d \geqq 3$ one can choose two neighbors y_1 and y_2, leaving the neighbors $y_3, \cdots, y_d$, such that: (a) tb $(T[x, y_2, y_3, \cdots, y_d]) \leqq k$, (b) tb $(T[x, y_1, y_3, \cdots, y_d]) \leqq k$, and (c) tb $(T(f)[x, y_1, y_2]) \leqq k-1$, where f is the function that assigns each vertex the integer one except for x, which is assigned to zero. We shall not give the proof of this statement, but simply observe that the proof follows basically the same pattern as shown in the proof of Theorem 3.1. The problem with this statement, at least from an algorithmic point of view, is that it defines the topological bandwidth of a tree in terms of the topological bandwidth of a vertex labeled tree and we do not have an algorithm to determine the topological bandwidth of labeled trees. However, we can use this statement to obtain the smallest trees with topological bandwidth 3 and 4 when the maximum degree of a vertex is restricted to 4 and 5. These are shown in Fig. 3.4.

In [11] Lengauer showed that the cutwidth of complete m-ary trees with height h is $\lceil h \cdot (m-1)/2 \rceil + 1$, for all $h \geqq 2$. We observe here that the topological bandwidth of a complete binary tree of height h is $\lceil h/2 \rceil$. This follows from the characterization that we have given, since it can be shown by induction on h that a copy of a tree in $\mathcal{F}_k$, where $k = \lceil h/2 \rceil$, is in the complete binary tree of height h and the complete binary tree of height h is the smallest complete binary tree that contains a subtree that is a homeomorphic image of a tree in $\mathcal{F}_k$, where $k = \lceil h/2 \rceil$. That is, assume that a homeo-

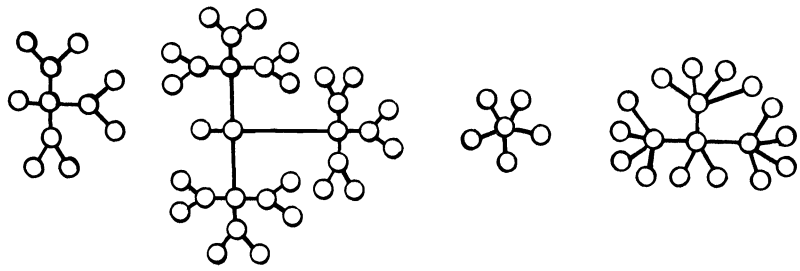


FIG 3.4. *A smallest degree* 4 *tree with topological bandwidth* (a) 3 and (b) 4, *and a smallest degree* 5 *tree with topological bandwidth* (c) 3 *and* (d) 4.

morphic image of a tree in $\mathcal{F}_k$, where $k = \lceil h/2 \rceil$, is in the complete binary tree of height h. Consider the complete binary tree of height $h+1$. If $\lceil (h+1)/2 \rceil = \lceil h/2 \rceil$, then clearly the binary tree of height $h+1$ has a subtree that is a homeomorphic image of a tree in $\mathcal{F}_k$, since $k = \lceil (h+1)/2 \rceil = \lceil h/2 \rceil$ and, consequently, such a tree is known to be in the smaller tree of height h. On the other hand, if $k = \lceil (h+1)/2 \rceil = \lceil h/2 \rceil + 1$, then there is a subtree of the complete binary tree of height $h-1$ which is a homeomorphic image of a tree in $\mathcal{F}_{k-1}$, since $\lceil h/2 \rceil = \lceil (h-1)/2 \rceil$. Consequently, there are three copies of such a tree in the complete binary tree of height $h+1$ and, thus, by the construction of the trees in $\mathcal{F}_k$, a subtree that is a homeomorphic image of a tree in $\mathcal{F}_k$.

4. NP-completeness. In this section we establish the NP-completeness of the problem of determining for an arbitrary graph G and integer k whether G has topological bandwidth at most k. Moreover, we show that the problem of determining, for an arbitrary degree 3 graph G and integer k, whether G has modified cutwidth at most k is NP-complete. The same technique is used to show that the Min Cut Linear Arrangement problem [5], [8], [11] and the Search Number problem [10], [15], [16] remain NP-complete even for graphs with maximum vertex degree three.

Consider the so-called rectangle graph with m "rows" and n "columns," denoted by $R(m, n)$, defined by: (1) the vertices of $R(m, n)$ are all pairs (i, j) such that $1 \leqq i \leqq m$ and $1 \leqq j \leqq n$, and (2) the edges of $R(m, n)$ are the following:

(a) $\{(i, j), (i, j+1)\}$, for all i, j $(1 \leqq i \leqq m$ and $1 \leqq j < n)$,
(b) $\{(i, 1), (i+1, 1)\}$, for all i $(1 \leqq i < m)$,
(c) $\{(i, n), (i+1, n)\}$, for all i $(1 \leqq i < m)$,
(d) $\{(2i-1, 2j), (2i, 2j)\}$, for all i, j $(1 \leqq i \leqq \lfloor m/2 \rfloor, 1 \leqq j < \lfloor n/2 \rfloor)$, and
(e) $\{(2i, 2j+1), (2i+1, 2j+1)\}$, for all i, j $(1 \leqq i < \lfloor m/2 \rfloor, 1 \leqq j < \lfloor n/2 \rfloor)$.

The rectangle $R(4, 8)$ with four rows and 8 columns is shown below in Fig. 4.1.

We will show that, for all $m \geqq 3$ and all $n \geqq 4m+8$, $R(m, n)$ has cutwidth $m+1$ and modified cutwidth $m-1$. This is equivalent to showing that $R(m, n)$ has search

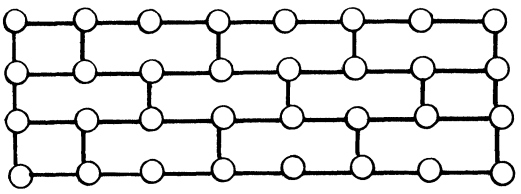


FIG. 4.1. *The rectangle* $R(4, 8)$.

number $m+1$ and topological bandwidth m, since $R(m, n)$ is a graph with maximum vertex degree three.

First, observe that the following layout L gives $R(m, n)$ cutwidth $m+1$ and modified cutwidth $m-1$. L is the layout that maps the vertex (i, j) to the integer $i \cdot (m-1)+j$, for all i, j ($1 \leqq i \leqq m$ and $1 \leqq j \leqq n$). For example, the layout L for the rectangle $R(4, 8)$ is shown in Fig. 4.2. It is straightforward to show, in general, that $R(m, n)$ has cutwidth $m+1$ and modified cutwidth $m-1$ under the layout L.

Next, observe that if we partition the vertices of $R(m, n)$, where $n \geqq 4m+8$, into two disjoint subsets V_1 and V_2 such that both V_1 and V_2 either contain one vertex from each of the m rows of $R(m, n)$ or one vertex from $2m$ distinct columns of $R(m, n)$, then there are vertex disjoint paths connecting m vertices in V_1 to m vertices in V_2. This can be established by induction on m and n. For example, if V_1 and V_2 both have one vertex from each of the m rows of $R(m, n)$, then m vertex disjoint paths can be formed along each of the rows to connect the m pairs of vertices.

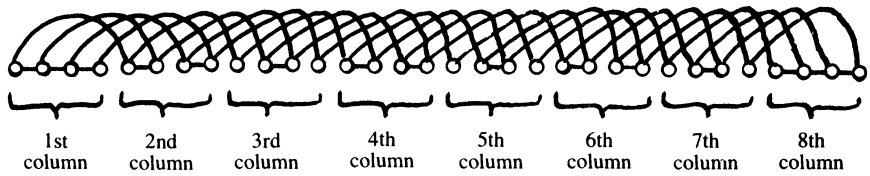


FIG. 4.2. *A layout of* $R(4, 8)$ *with cutwidth* 5 *and modified cutwidth* 3.

Note that any partition of the vertices of $R(m, n)$ into two disjoint sets V_1 and V_2 such that both V_1 and V_2 have at least $2m^2$ vertices must be such that both V_1 and V_2 have either one vertex from each of the m rows of $R(m, n)$ or one vertex from $2m$ distinct columns of $R(m, n)$. It follows that $R(m, n)$ has modified cutwidth at least $m-1$ under any linear layout, when $n \geqq 4m^2+8$, since any linear layout partitions the set of vertices in $R(m, n)$ into the set V_1, consisting of the first $2m^2$ vertices in the layout, and V_2, consisting of all the remaining vertices. That is, since there are m vertex disjoint paths connecting V_1 and V_2, there must be at least $m-1$ edges passing over the $(2m^2+1)$st vertex under this layout. Furthermore, we observe by the same argument that there are at least $m-1$ edges passing over every vertex in any linear layout of $R(m, n)$ which is assigned to a position between the $(2m^2+1)$st and the position $2m^2+1$ from the right end of the layout. Thus, if n is large, the number of edges passing over the vertices in the middle of $R(m, n)$ under any linear layout is at least $m-1$.

Observe that next to any degree 3 vertex in $R(m, n)$ which has at least $m-1$ edges passing over it there must be a cut with at least $m+1$ edges. That is, if x is a degree 3 vertex, is the ith vertex in the linear layout, and has at least $m-1$ edges passing over it, then either passing between the ith vertex and the $(i-1)$st vertex or between the ith vertex and the $(i+1)$st vertex there must be at least $m+1$ edges. This follows simply from the fact that at least two of the edges incident to the vertex x must contribute to one of these two cuts. We note that $R(m, n)$ has at most n degree 2 vertices, so if n is large there will be at least one vertex in the middle of any linear layout which has degree 3. $R(m, n)$ has $m \cdot n$ vertices. At most $4m^2$ vertices of $R(m, n)$ need to be used at the beginning and end of any linear layout to guarantee having at least one vertex from each row or at least one vertex from each of $2m$ distinct columns in the set of starting vertices and the set of terminating vertices. Also, we have seen that at most n vertices in $R(m, m)$ have degree 2. Therefore, there must be at least

$m \cdot n - 4m^2 - n$ vertices with degree 3 in $R(m, n)$ that are in the "middle" of any linear layout and, consequently, have at least $m-1$ edges passing over them. Since $n \geqq 4m+8$, $m \cdot n - 4m^2 - n = (m-1) \cdot n - 4m^2 \geqq (m-1) \cdot (4m+8) - 4m^2 = 4m - 8$ and, since $m \geqq 3$, this is a positive integer. Therefore, there is at least one degree 3 vertex with $m-1$ edges passing over it in any linear layout of $R(m, n)$, when $m \geqq 3$ and $n \geqq 4m+8$. This implies that $R(m, n)$ has cutwidth at least $m+1$. Since we have already seen a layout that gave $R(m, n)$ cutwidth $m+1$, it follows that $R(m, n)$ has cutwidth exactly $m+1$. We note also that in any linear layout of $R(m, n)$ every cut from the $(2m+1)$st to the $(n \cdot m - 2m + 1)$st has at least m edges.

The problem Modified Cutwidth is the following: Given a graph G and an integer k, one asks if there is a linear layout L such that $\text{mcw}(G, L) \leqq k$. It is known that Modified Cutwidth is NP-complete [12]; however, the graph constructed in this reduction has very large degree. In the following we show that Modified Cutwidth remains NP-complete when restricted to graphs with maximum vertex degree three. It follows from Theorem 2.2 that the Topological Bandwidth problem is also NP-complete for degree three graphs.

THEOREM 4.1. *Modified Cutwidth is* NP*-complete for graphs with maximum vertex degree three.*

Proof. We reduce the NP-complete problem Min Cut into Equal Size Subsets [7] to the Modified Cutwidth problem. Min Cut into Equal Size Subsets denotes the problem in which, given a graph G and an integer k, one asks whether there is a partition of the vertices of G into two equal size subsets, say V_1 and V_2, such that the number of edges with one endpoint in V_1 and one endpoint in V_2 is not greater than k.

Let $G = (V, E)$ be a graph with N vertices and let k be an arbitrary positive integer. We construct a graph G' with maximum vertex degree three and an integer k' such that (G, k) is a positive instance of Min Cut into Equal Size Subsets if and only if (G', k') is a positive instance of the Modified Cutwidth problem.

G' consists of several rectangle components. In fact, G' has one copy of the rectangle graph $R(m, n)$, where $m = 2 \cdot N^4$ and $n = 8 \cdot (N^4+1)$, for each of the N vertices in G. G' also has a component, called an H-shaped graph, which is shown in Fig. 4.3. The H-shaped graph is formed by combining a rectangle $R(m, n)$, where $m = 2 \cdot N^4$ and $n = 36 \cdot (N^4+1)$, with four rectangles $R(m, n)$, where $m = N^4/2$ and $n = 12 \cdot (N^4+1)$, in the manner shown in the figure. (We assume, without loss of generality, that N is even.)

From each of the N rectangle components we add a sheaf of $2N^2$ edges connecting distinct degree 2 vertices in the rectangle to $2N^2$ distinct degree 2 vertices in the middle

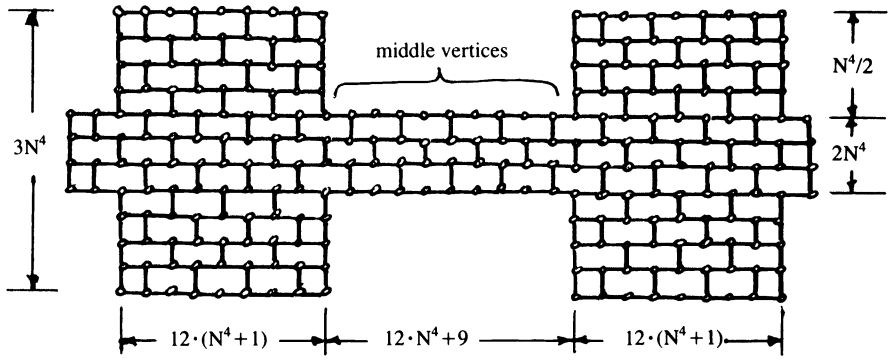


FIG. 4.3. *The H-shaped graph used in the proof of* NP*-completeness.*

of the H-shaped graph. In addition, for each edge $\{u, v\}$ of the original graph G we add an edge connecting a degree 2 vertex in the rectangle component corresponding to the vertex u with a degree 2 vertex in the rectangle component corresponding to the vertex v. These are all the edges in the graph G' that connect the components. It should be noted that, since the edges added to connect the components are only incident to vertices that formerly had degree two, the resulting graph has degree three.

Choose the integer k' to be $3N^4+N^3+k-1$. We show that (G, k) is a positive instance of the problem Min Cut into Equal Size Subsets if and only if (G', k') is a positive instance of Modified Cutwidth.

First, let (G', k') be a positive instance of Modified Cutwidth. Then, there is a linear layout L' of G' such that $\text{mcw}(G', L') \leqq k'$. As we have shown previously any linear layout of a rectangle $R(m, n)$, where $m \geqq 3$ and $n \geqq 4m+8$, must have modified cutwidth at least $m-1$. It follows that the separate components of G' cannot overlap heavily under the layout L'. That is, each $2N^4$ by $8 \cdot (N^4+1)$ rectangle has modified cutwidth $2N^4-1$ and, in fact, has $2N^4-1$ edges passing over every vertex in its "middle" under any linear layout. Thus, if the middle vertices of two such rectangles overlapped in the layout L', then L' would have modified cutwidth at least $4N^4-2$. Since $4N^4-2$ is greater than k' (we are assuming that N is large, without loss of generality), this is a contradiction. So, separate components cannot overlap heavily. In fact, one can identify a layout of the original graph G from the layout of G' by assigning each vertex v to the position occupied by the corresponding rectangle component.

Notice that the H-shaped component contains a copy of the rectangle $R(m, n)$ at both ends, where $m = 3N^4$ and $n = 12(N^4+1)$. Therefore, these rectangles at each end of the H-shaped graph have modified cutwidth at least $3N^4-1$ under the linear layout L'. Since G' has modified cutwidth $k' = 3N^4+N^3+k-1$ under the layout L, at most N^3+k additional edges can pass over either of these rectangles at the ends of the H-shaped graph. It follows that half of the rectangles components corresponding to vertices in the original graph G are positioned to the right of the H-shaped graph and the other half are positioned to the left. That is, if $N/2+1$ of the rectangle components are placed on one side of the H-shaped graph, then there would be $(N/2+1) \cdot 2N^2 = N^3+2N^2$ edges passing over one of the rectangles at the end of the H-shaped graph. Since at most N^3+k edges can pass over, as we have seen, and k is less than $2N^2$, without loss of generality, this is a contradiction. So, half of the rectangle components must be placed to the left of the H-shaped graph by the layout L' and the other half must be placed to the right of the H-shaped graph.

So, the vertices of G are also partitioned into two equal size subsets, say V_L and V_R, consisting of those vertices whose corresponding rectangle component lies to the left or to the right of the H-shaped graph, respectively. There can be at most k edges connecting vertices in V_L with vertices in V_R. That is, in the graph G', when the layout L' assigns half of the rectangle components to positions to the left of the H-shaped graph and the other half to positions to the right of the H-shaped graph, there are $3N^4-1+(N/2)2N^2 = 3N^4+N^3-1$ edges passing over vertices in the ends of the H-shaped graph. Thus, since $k' = 3N^4+N^3+k-1$, it follows that at most k more edges can pass over these vertices. So, at most k edges connect vertices in the rectangle components to the left of the H-shaped graph with vertices in the rectangle components to the right of the H-shaped graph. This is equivalent to saying that there are k edges connecting vertices in V_L with vertices in V_R in the graph G.

Conversely, let (G, k) be a positive instance of the Min Cut into Equal Sized Subsets problem. Let V_1 and V_2 be the sets in a partition of the vertices of G so that

$|V_1| = |V_2|$ and at most k edges have one endpoint in V_1 and one endpoint in V_2. Lay out the graph G' so that all the rectangle components that correspond to vertices in V_1 are to the left of the H-shaped graph and all the rectangle components that correspond to vertices in V_2 are to the right of the H-shaped graph. Furthermore, these rectangle components are laid out in such a way that they do not overlap. With such a layout it is straightforward to show that G' has modified cutwidth $k' = 3N^4 + N^3 + k - 1$. The basic structure of this layout is shown in Fig. 4.4. So, (G', k') is a positive instance of Modified Cutwidth. □

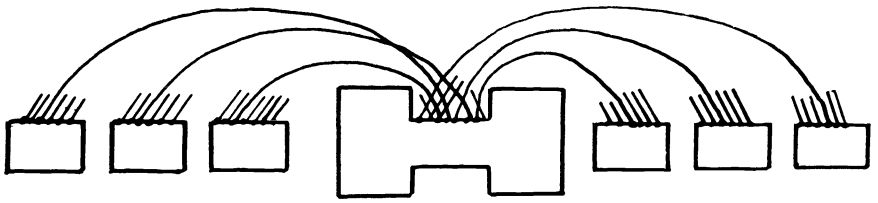


FIG. 4.4. *An illustration of the basic structure in a layout of the graph G' constructed in the proof of Theorem* 4.1.

COROLLARY 4.1. *The Topological Bandwidth problem is* NP-*complete even when restricted to graphs with maximum vertex degree three.*

The corollary follows immediately from Theorem 2.2. The following result can be shown by the same construction as described in the proof of Theorem 4.1, except the value of k' needs to be changed to $3N^4 + N^3 + k + 1$.

COROLLARY 4.2. *The Min Cut Linear Arrangement problem is* NP-*complete even when restricted to graphs with maximum vertex degree three.*

COROLLARY 4.3. *The Search Number problem is* NP-*complete even when restricted to graphs with maximum vertex degree three.*

Corollary 4.3 follows from the fact that cutwidth and search number are identical for graphs with maximum vertex degree three [15]. Similarly, it follows from Theorem 2.3 that the problem of determining, given a graph G and an integer k, whether the node search number of G is at most k is NP-complete, even when the graph G has maximum vertex degree three.

5. Characterization of graphs with topological bandwidth 2. Clearly a graph has topological bandwidth one if and only if it is a simple chain. What kind of graphs have topological bandwidth two? We shall now provide an answer to this question. It should perhaps be noted that no characterization is yet known for graphs having bandwidth two, although a linear time algorithm to decide if a graph has bandwidth 2 has been described [6]. First, we shall give a characterization of trees with topological bandwidth two.

LEMMA 3.1. *Let T be a tree. The following statements are equivalent*:

1. $\text{tb}(T) \leqq 2$,
2. *T does not contain a subtree that is a homeomorphic image of one of the trees shown in Fig.* 5.1, *and*
3. *T has maximum vertex degree four and there is a path P in T which contains all degree four vertices and is such that all degree* 3 *vertices in T are either on the path P or are adjacent to a degree* 3 *vertex on P.*

Proof. $(1 \to 2)$ It is easily seen that none of the trees described in Fig. 5.1 have topological bandwidth two. So, if T is a tree with topological bandwidth 2, then T cannot have a subtree that is a homeomorphic image of any one of these trees.

(2→3) Suppose that T did not contain a path that contained all degree 4 nodes. It would then follow that T contains a subtree that is a homeomorphic image of the tree (c) in Fig. 5.1. If no path P exists which contains all degree 4 vertices and is such that every degree 3 vertex is either on the path P or adjacent to a degree 3 vertex on the path, then T would contain a subtree that is a homeomorphic image of one of the trees (b), (d), (e), or (f). Finally, if T contains a vertex with degree 5 or larger, then it contains a copy of tree (a) in Fig. 5.1.

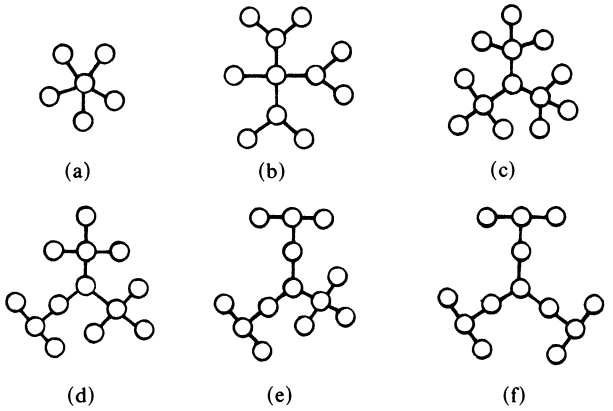


FIG. 5.1. *Forbidden subtrees for any tree with topological bandwidth two.*

(3→1) Let T have maximum vertex degree 4 and have a path P containing all degree 4 vertices and such that all degree 3 vertices are either on the path P or are adjacent to a degree 3 vertex in P. Let the sequence of vertices in the path P be $a_1, a_2, \cdots, a_m$ ($m \geqq 1$) so that, for all i ($1 \leqq i \leqq m$), $\{a_i, a_{i+1}\}$ is an edge in T. We describe a homeomorphic image of T and a bandwidth 2 layout of this tree informally. If a_i is a degree 4 vertex, then it must necessarily be attached to two simple chains of vertices that are not part of P. These simple chains can be laid out to fit into consecutive empty spaces between a sufficiently large number of degree 2 vertices added into the edges $\{a_{i-1}, a_i\}$ and $\{a_i, a_{i+1}\}$. If a_i is a degree 3 vertex, then either it is attached to a single vertex at the end of a simple chain or it is joined by an edge to a vertex in a simple chain. The first case is handled in the same way as specified for degree 4 vertices. In the latter case, the degree 3 vertex adjacent to a_i is laid out next to a_i and the ends of the chain are inserted into consecutive open spaces made available by inserting a sufficiently large number of degree 2 vertices into $\{a_{i-1}, a_i\}$ and $\{a_i, a_{i+1}\}$. In this way we see that the tree T has topological bandwidth two. □

Let us now turn to biconnected graphs. In the following we see that there is a close relationship between topological bandwidth 2 and cutwidth 3 in biconnected graphs. Let reduce (G) be the reduction of a graph G. Then, for any biconnected graph G, $\text{cw}(G) = 3$ if and only if $\text{tb}(\text{reduce}(G)) = 2$. That is, let $\text{cw}(G) = 3$. Since $G' = \text{reduce}(G)$ has no vertices with degree less than 3, $\text{mcw}(G') \leqq \text{cw}(G) - 2$. By Theorem 2.1, it follows that $\text{tb}(G') \leqq \text{cw}(G') - 1$. Consequently, since $\text{cw}(G) = \text{cw}(G')$, $\text{tb}(G') \leqq 2$. Conversely, let $\text{tb}(\text{reduce}(G)) \leqq 2$. So, there is a homeomorphic image G' of reduce (G) such that $b(G') \leqq 2$. It is easily seen that any graph with bandwidth at most 2 has cutwidth at most 3, so $\text{cw}(G') \leqq 3$. Since $\text{cw}(G) = \text{cw}(G')$, we have $\text{cw}(G) \leqq 3$.

A biconnected graph G is *outerplanar* if it has a planar embedding in which a single face includes all of its vertices. The edges of the face are called *sides*; the

remaining edges are called *chords*. A chord $\{x, y\}$ in a biconnected outerplanar graph is a *depth one chord* if there is a path from x to y through sides and vertices that are incident to no other chords.

LEMMA 5.2. *Let G be a biconnected graph.* $\text{tb}(G) \leqq 2$ *if and only if G is outerplanar, has at most 2 chords of depth one, and, if $\{x, y\}$ and $\{x, z\}$ are distinct chords in G, then y and z are connected by a side.*

Proof. Let $\text{tb}(G) = 2$. Then there is a homeomorphic image G' of G and a linear layout L of G' such that $b(G', L) = 2$. Let A, B be the first and last vertices of G' under the layout L, respectively. There must be two vertex disjoint paths connecting A and B, since G is biconnected. Since G' has bandwith 2 under the layout L, the vertices in these two paths must alternate: one occupying the odd numbered positions and the other occupying the even numbered positions. All additional edges in G' must connect consecutive odd and even numbered vertices. Consequently, G' is outer planar. That is, place all the vertices along a line in the order of their appearance in the linear layout L, draw the edges in one path above this line, draw the edges in the other path below this line, and draw all additional edges on the line. It follows that this is an outerplanar representation of G'. All the vertices are on the external face. Furthermore, all of the chords connect consecutive vertices along the line, so they do not cross and there can be at most two of depth one. Furthermore, if $\{x, y\}$ and $\{x, z\}$ are chords in G', then y and z must be successive vertices of one or the other of these two paths and, consequently, y and z are connected by a side.

Conversely, let G be a biconnected outer planar graph with at most two depth one chords and satisfying the property that, if $\{x, y\}$ and $\{x, z\}$ are chords in G, then y and z are connected by a side. Consider an arbitrary outerplanar representation of G. If G has no chords, then clearly G has bandwidth 2. Similarly, if G has only one chord, then it is easy to see that it has bandwidth 2. So, assume that G has at least two chords. Consequently, there are two distinct depth one chords in G. Let A and B be vertices that are spanned by these different depth one chords. (A vertex x is spanned by a depth one chord, if x is one of the vertices encountered in a path along the sides from one end of the chord to the other that does not enter any vertex incident to another chord.) We construct a linear layout of a homeomorphic image of G in which vertex A is the leftmost vertex and vertex B is the rightmost vertex. Let P_1 and P_2 be the two paths along the sides from A to B. We add degree 2 vertices to these sides so that (1) the path P_1 involves vertices in the even numbered positions, (2) the path P_2 involves vertices in the odd numbered positions, and (3) all of the chords connect consecutive even and odd numbered positions. This is always possible, since all chords connect vertices in P_1 with vertices in P_2 and we can fill in as many degree 2 vertices as required to make the chords connect consecutive numbered vertices. Furthermore, there can be no conflicting demands made by the chords, since, if $\{x, y\}$ and $\{x, z\}$ are distinct chords, then y and z are joined by a side. That is, y and z must be consecutive vertices in the path P_1 or the path P_2. So, we see that there is a homeomorphic image of the graph G with bandwidth 2 and, consequently, G has topological bandwidth 2. □

In [15] biconnected graphs with cutwidth 3 were characterized as outerplanar graphs satisfying the "collinear chord" property. Although all topological bandwidth 2 graphs satisfy this collinear chord property, biconnected outerplanar graphs with the collinear chord property do not necessarily have topological bandwidth 2. They do, as we have seen, when the graph is reduced.

We have characterized biconnected graphs and trees having topological bandwidth 2; now we turn to a general characterization of graphs with topological bandwidth 2.

Let us say that a biconnected component of a graph G is *simple*, if it consists of a single edge; otherwise, it is *nonsimple.*

Let G be a connected graph with topological bandwidth 2. So, there is a homeomorphic image G' of G and a linear layout L of G' such that $b(G', L) = 2$. Let A, B be the first and last vertices of G' under the layout L, respectively. Since G, and hence G', is connected, there is a path P connecting A and B. Let $C' = C'_1, C'_2, \cdots, C'_m$ ($m \geqq 1$) be the sequence of biconnected components in G' such that (1) C'_1 contains the first vertex $a_0 = A$, (2) C'_m contains the last vertex $a_m = B$, (3) for all i ($1 \leqq i < m$), C'_i and C'_{i+1} share an articulation point a_i, and (4) for all i ($1 \leqq i \leqq m$) C'_i shares at least two vertices with the path P. Oberve that from the leftmost vertex of a nonsimple biconnected component C'_i to the rightmost vertex of C'_i there can only be vertices from C'_i. That is, since L is a bandwidth 2 layout and there are two vertex disjoint paths connecting any distinct pair of vertices in C'_i, there are no available spaces for vertices not in C'_i between its leftmost and rightmost vertices. In fact, the vertices in the sequence C' can be assumed, without any loss of generality, to be laid out in the order of their appearance in the sequence C'.

In what way can these biconnected components be joined? For each i ($1 \leqq i \leqq m$), let a_{i-1} and a_i be the *connection points* of the component C'_i. What conditions must these connection points satisfy? Let us call a sequence $v_1, v_2, \cdots, v_r$ ($r \geqq 2$) of vertices in a biconnected outer planar graph an *end-region* if, for all i ($1 \leqq i \leqq r$), $\{v_i, v_{i+1}\}$ is a side and $\{v_1, v_r\}$ is a depth one chord. Every outerplanar biconnected graph with at least one chord has at least two distinct end-regions. As we have seen in Lemma 5.2, if a biconnected outerplanar graph has topological bandwidth 2, then there can be at most two end-regions.

First, observe that if C'_i has a chord, then the connection points a_{i-1} and a_i must be in distinct end-regions. That is, there are two vertex disjoint paths, say P_1 and P_2, connecting a_{i-1} and a_i which traverse the sides of C'_i. We have seen, in the proof of Lemma 5.2, that chords must connect vertices in path P_1 to vertices in the path P_2. If a_{i-1} and a_i are not in distinct end-regions, then there is a chord connecting vertices in the same path. This is impossible. So, connection points must be in distinct end-regions.

Furthermore, if C'_i is a nonsimple component and a_{i-1} or a_i is incident to a chord of C'_i, then the end-regions defined by this chord must contain at most three vertices. (In fact, it contains exactly three vertices.) Without loss of generality, assume a_{i-1} is incident to a chord. In this case, a_{i-1} cannot be the first vertex in the layout (which must have degree two), as it has degree at least 4. That is, it is incident to two sides of C'_i, a chord of C'_i, and an edge in the preceding component C'_{i-1}. Clearly, in any bandwidth 2 layout, two of the adjacent vertices must be placed before a_{i-1} and the other two after it. The chord incident to a_{i-1} must, in fact, connect a_{i-1} with the vertex to its right and there must be sides of C'_i that connect (1) a_{i-1} with the vertex on its left and (2) the vertex on the left of a_{i-1} with the vertex on the right of a_{i-1}. Consequently, the end-region defined by the chord incident to a_{i-1} contains a_{i-1}, the vertex to its left, and the vertex to its right. So, the end-region has three vertices. See Fig. 5.2 for an example.

Observe that the nonsimple components of G' correspond to nonsimple components of G and the articulation points in these nonsimple components are also articulation points in the original graph G. That is, corresponding to the sequence C' of biconnected components in the graph G' there is a sequence C of biconnected components in the graph G. The rules for connection points that we have just stated for G' must also hold in G. That is, (1) connection points must be in distinct end-regions,

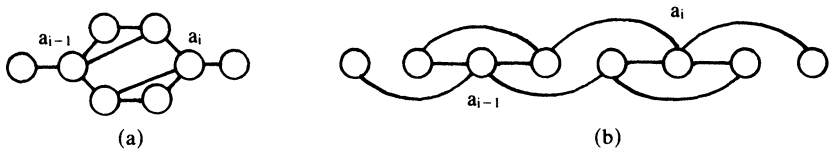


FIG. 5.2. (a) *Three biconnected components joined by the articulation points* a_{i-1} *and* a_i. (b) *A bandwidth* 2 *layout of these components.*

and (2) if a connection point is incident to a chord, then this chord defines an end-region with at most three vertices. Connection points that satisfy these two properties are called *valid* connection points.

Let G be a graph formed by taking a sequence of biconnected graphs $C_1, C_2, \cdots, C_m$, such that C_i has topological bandwidth 2, for all i $(1 \leqq i \leqq m)$, and coalescing a vertex a_i in C_i with a vertex in C_{i+1}. It follows that G has topological bandwidth 2 if and only if there exists a choice of vertices a_0 from C_1 and a_m from C_m such that, for all i $(1 \leqq i \leqq m)$, a_{i-1} and a_i are valid connection points for C_i.

A graph G with topological bandwidth 2 may be, however, more than a chain of biconnected graphs. It may be a chain $C = C_1, C_2, \cdots, C_m$ $(m \geqq 1)$ of biconnected graphs together with various attached graphs. Let us call a vertex in a graph *simple* if it is not part of any nonsimple biconnected component and *semisimple* if it is part of a simple component and a nonsimple component.

What kind of graphs can be attached to the vertices of a chain of biconnected components? To which vertices can such graphs be attached?

Let G be a connected graph with topological bandwidth 2. There is a homeomorphic image G' of G and a linear layout L of G' such that $b(G', L) = 2$. Let A and B be the first and last vertices in the layout L. There must be a path P connecting A and B. When all of the vertices along the path P and all of the edges incident to these vertices are deleted from G, the resulting graph must have bandwith one, although it need not be connected. That is, in deleting the vertices in P from G we delete at least one vertex from every two successive vertices. So, the remainder must have bandwidth one. Consequently, the remainder must be at most a collection of simple chains.

As we have seen earlier, there must be a chain C' of biconnected components in G' and, therefore, a chain $C = C_1, C_2, \cdots, C_m$ $(m \geqq 1)$ of biconnected components in G such that, for all i $(1 \leqq i < m)$, C_i and C_{i+1} share an articulation point a_i and, for some choice of vertices a_0 in C_1 and a_m in C_m, and for all i $(1 \leqq i \leqq m)$, a_{i-1} and a_i are valid connection points for C_i. (If C_i is a simple component, we shall agree that its two vertices always form a valid pair of connection points.)

If a_i is a simple articulation point, so C_i and C_{i+1} are single edges, then any one of the following types of attachment can be made to a_i:

Type (1). A single edge e is attached by joining the ends of the edge by new edges to a_i. (Thus a cycle of three vertices is attached by coalescing one of its vertices with a_i.)

Type (2). A simple chain is attached by joining one of its vertices with a new edge to a_i.

Type (3). A simple chain is attached by coalescing one of its vertices with a_i.

It is easy to see that these are the only possible types of attachments, since if a_i and its incident edges are deleted, then the result must have topological bandwidth one. Furthermore, any graph attachment to a_i cannot be attached to any other vertex of the chain C or it would not be an attachment, but would be part of a biconnected component in the chain. Also, note that it is not possible to have a chain, instead of

just a single edge, attached as described in Type (1). If a chain of more than one edge were attached by joining its endpoints by new edges to a_i, then the result cannot have topological bandwidth 2. So, the only permissible attachments are of Types (1)-(3).

There may also be attachments to vertices in the nonsimple biconnected components of the chain C, but these are even more restricted, as we shall see. As we know, in any bandwidth 2 layout, from the leftmost vertex of a nonsimple biconnected component to its rightmost vertex there can only be vertices that are part of this component. Consequently, the only possible attachments to a nonsimple component are simple chains attached by coalescing one of the ends of the chain to one of the two leftmost vertices in this component or to one of the two rightmost vertices in this component. That is, one chain may be attached to one of the two leftmost vertices and one chain may be attached to one of the two rightmost vertices. Furthermore, if the connection point on the right (left) end of the component is not the rightmost (leftmost) vertex in the component, then the chain can only be attached to the rightmost (leftmost) vertex. This follows from the fact that the connection points must already have an edge connecting to vertices in another component.

Thus, the problem of identifying where chains can be attached to nonsimple components becomes that of identifying which vertices can be the two leftmost and which vertices can be the two rightmost vertices in a layout of this component. Clearly, the connection points of a component need to be one of these extreme vertices, since they need to be connected by an edge to vertices in components before or after the current component. Furthermore, the other vertices at the left or right end of the layout must be vertices joined to these connection points by a side of the outerplanar component. Note that a connection point can always be chosen to be the leftmost (rightmost) vertex of the component, except when it is incident to a chord, as we have seen. In this case the leftmost (rightmost) vertex of the layout of this component must be the vertex in the end-region defined by this chord and it must not be incident to any chord.

So, we have seen that (1) only simple chains can be attached to vertices in nonsimple components of a topological bandwidth 2 graph, (2) these chains must be attached to either the connection points or vertices connected to these connection points by a side and which are in the end-region containing the corresponding connection point, and (3) there can be at most one chain attached per end-region. It should be noted that no chain can be attached to a connection point incident to chord in a nonsimple biconnected component, since then this vertex would have degree at least five, which is impossible for a bandwidth 2 graph. Also, note that a simple chain attachment is possible to the leftmost (rightmost) two vertices only when C_{i-1} (C_{i+1}) is a simple component. That is, only then can the vertices in the chain be inserted into spaces made available by the insertion of an arbitrary large number of degree two vertices into the edge of this component. Finally, note that if a nonsimple component has four or more vertices, then the simple chains attached to this component must be attached to distinct vertices. That is, a single vertex in such a case cannot be both one of the leftmost two vertices and one of the rightmost two vertices of this component.

Thus, we have shown the following characterization of graphs with topological bandwidth 2:

THEOREM 5.1. *Let G be a graph. The topological bandwidth of G is at most 2 if and only if*:

(1) *all nodes in G have degree at most 4*;

(2) *there is a chain of biconnected components $C = C_1, C_2, \cdots, C_m$ ($m \geqq 1$) in G such that, for all i ($1 \leqq i < m$), C_i and C_{i+1} share an articulation point a_i, and one can*

choose vertices a_0 and a_m in C_1 and C_m, respectively, such that, for all i $(1 \leqq i \leqq m)$, C_i has topological bandwidth 2 and the vertices a_{i-1} and a_i are valid connection points for C_i; and

(3) *G can be obtained from C by attaching various graphs such that all of the following conditions are satisfied:*

(a) *for all i $(1 \leqq i \leqq m)$, if a_i is simple, then attachments of Types* (1)-(3) *may be attached to a_i,*

(b) *for all i $(1 \leqq i \leqq m)$, if a_i is semisimple, then simple chains may be attached to either a_i or to a vertex connected to a_i by a side of the nonsimple outerplanar component provided that this vertex is in the end-region containing a_i, and*

(c) *at most two simple chains can be attached to a nonsimple component: one to each end-region. If the component has at least four vertices, then chains must be attached to distinct vertices. All attachments are of the kind indicated in* (a) *or* (b).

To illustrate the theorem we describe some graphs with topological bandwidth two in Fig. 5.3 and some graphs with topological bandwidth greater than two in Fig. 5.4.

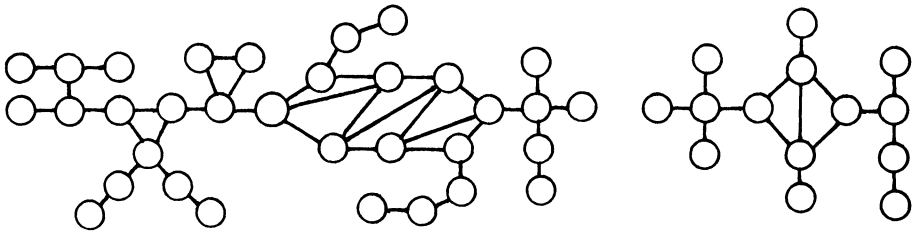


FIG. 5.3. *Some graphs with topological bandwidth two.*

Theorem 5.1 suggests a linear time algorithm to decide if a graph has topological bandwidth two. Basically, the algorithm would follow the following steps:

1. Find the biconnected components of G,

2. See if a chain of these biconnected components can be formed that satisfies all of the properties indicated in Theorem 5.1. If it is not possible to construct such a chain of components, then stop and answer "no"; otherwise, if there is such a chain of components such that G is obtained from the chain by adding the indicated types of attachments, then stop and answer "yes."

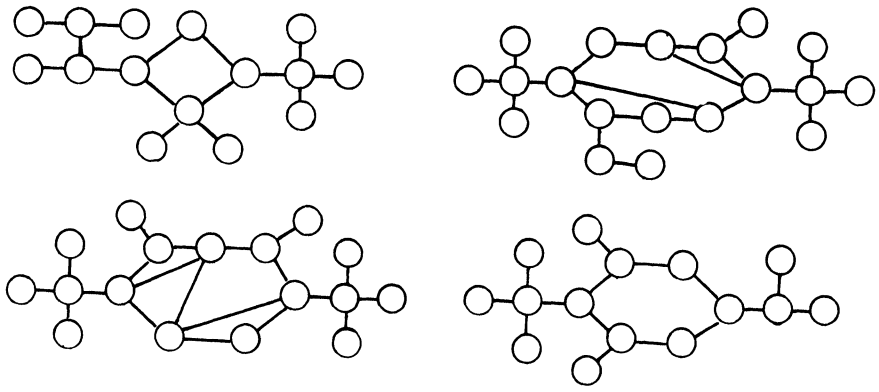


FIG. 5.4. *Some graphs with topological bandwidth greater than two.*

This algorithm works in linear time, since the biconnected components can be found in an amount of time bounded by some constant times the number of edges in G and the number of edges in any graph with topological bandwidth two must be at most two times the number of vertices, i.e. each vertex must have degree at most four. Furthermore, the location of the chain of components and the verification of the indicated properties, including the test for each nonsimple component satisfying the outerplanar property, can be done in linear time. We shall not give the details of such an algorithm here; however, the central idea should already be clear.

Topological bandwidth three seems to be much more difficult to characterize. However, it should be noted that a modification of the dynamic programming algorithms described in [9], [24], which show that the problem of deciding if a graph G with n vertices has bandwidth k can be decided in $O(n^k)$ steps, will suffice to show that the problem of deciding if a graph G with n vertices has topological bandwidth k can be decided in $O(n^k)$ steps. That is, the improved dynamic programming algorithm given in [9] can be modified to incorporate the possibility of degree two vertices being inserted into the edges of G without an order of magnitude increase in the running time of the algorithm. Thus, for all $k \geqq 3$, it is possible to recognize graphs with topological bandwidth k in $O(n^k)$ steps.

Acknowledgments. Charles Simonson, currently a graduate student at Northwestern University, suggested a technique for showing $\text{tb}(G) \leqq \text{mcw}(G)+1$, which, after several modifications, we have used to prove Theorem 2.1. We gratefully acknowledge the communication of errors in earlier versions of our Theorems 3.1-2 by F. R. K. Chung and D. S. Johnson. We are also grateful to R. L. Graham for communicating the term "topological bandwidth" for the subject of this paper during a visit to Northwestern University.

REFERENCES

[1] I. ARANY, L. SZODA AND W. F. SMITH, *An improved method for reducing the bandwidth of sparse symmetric matrices*, Proc. 1971 IFIP Congress, pp. 1246-1250.

[2] P. Z. CHINN, J. CHVATALOVA, A. K. DEWDNEY AND N. E. GIBBS, *The bandwidth problem for graphs and matrices*, J. Graph Theory, 6 (1982), pp. 223-254.

[3] F. R. K. CHUNG, *Some problems and results in labelings of graphs*, in The Theory and Applications of Graphs, G. Chartrand, ed., John Wiley, New York, 1981, pp. 255-263.

[4] ———, *On the cutwidth and the topological bandwidth of a tree*, Technical Report, Bell Laboratories, Murray Hill, NJ, 1982; this Journal, 6 (1985), pp. 268-277.

[5] M.-J. CHUNG, F. S. MAKEDON, I. H. SUDBOROUGH AND J. TURNER, *Polynomial time algorithms for the min cut problem on degree restricted trees*, SIAM J. Comput., 14 (1985), pp. 158-177.

[6] M. R. GAREY, R. L. GRAHAM, D. S. JOHNSON AND D. KNUTH, *Complexity results for bandwidth minimization*, SIAM J. Appl. Math., 34 (1978), pp. 477-495.

[7] M. R. GAREY AND D. S. JOHNSON, *Computers and Intractability: A Guide to the Theory of* NP-*Completeness*, W. H. Freeman, San Francisco, 1979.

[8] F. GAVRIL, *Some* NP-*complete problems on graphs*, Proc. 11th Annual Conference on Information Sciences and Systems, Johns Hopkin Univ., Baltimore, MD, 1977, pp. 91-95.

[9] E. M. GURARI AND I. H. SUDBOROUGH, *Improved dynamic programming algorithms for the bandwidth minimization problem and the Min Cut Linear Arrangement problem*, J. Algorithms, to appear.

[10] A. LAPAUGH, *Recontamination does not help*, Technical Report, Dept. Computer Science, Princeton Univ., Princeton, NJ, 1982.

[11] T. LENGAUER, *Upper and lower bounds on the complexity of the min cut linear arrangement problem on trees*, this Journal, 3 (1982), pp. 99-113.

[12] ———, *Black-white pebbles and graph separation*, Acta Inform., 16 (1981), pp. 465-475.

[13] W. LIN AND A. B. SHERMAN, *Comparative analysis of the Cuthill-McKee ordering algorithms for sparse matrices*, SIAM J. Numer. Anal., 13 (1976), pp. 198-213.

[14] F. MAKEDON, *Layout problems and their complexity*, Ph.D. Thesis, Electrical Engineering and Computer Science Dept., Northwestern Univ., Evanston, IL, 1982.

[15] F. MAKEDON AND I. H. SUDBOROUGH, *Minimizing width in linear layouts*, in Proc. 10th International Colloquium on Automata, Languages, and Programming, Lecture Notes in Computer Science 154, Springer-Verlag, New York, 1983, pp. 478-490.

[16] N. MEGIDDO, L. HAKIMI, M. R. GAREY, D. S. JOHNSON AND C. H. PAPADIMITRIOU, *The complexity of searching a graph*, Proc. 22nd Annual IEEE Symposium on Foundations of Computer Science, 1981, pp. 376-385.

[17] B. MONIEN AND I. H. SUDBOROUGH, *On eliminating nondeterminism from Turing machines that use less than logarithm worktape space*, Theoret. Comput. Sci., 21 (1982), pp. 237-253.

[18] ———, *Bandwidth constrained* NP*-complete problems*, Proc. 11th Annual ACM Symposium on Theory of Computing, 1981, pp. 207-217.

[19] T. OHTSUKI, H. MORI, E. S. KUH, T. KASHIWABARA AND T. FUJISAWA, *One-dimensional logic gate assignments and interval graphs*, IEEE Trans. Circuits Systems, CAS-26 (1979), pp. 675-684.

[20] C. H. PAPADIMITRIOU AND L. KIROUSIS, personal communication.

[21] ———. *The* NP*-completeness of the bandwidth minimization problem*, Computing, 16 (1976), pp. 237-267.

[22] R. R. REDZIEJOWSKI, *On arithmetic expressions and trees*, Comm. ACM, 12 (1969), pp. 81-84.

[23] A. L. ROSENBERG AND I. H. SUDBOROUGH, *Bandwidth and pebbling*, Computing, 31 (1983), pp. 115-139.

[24] J. B. SAXE, *Dynamic programming algorithms for recognizing small bandwidth graphs in polynomial time*, this Journal, 1 (1980), pp. 363-369.

[25] L. J. STOCKMEYER, private communication to M. R. Garey and D. S. Johnson, cited in [7, p. 201].

[26] I. H. SUDBOROUGH, *Bandwidth constraints on problems complete for polynomial time*, Theoret. Comput. Sci., 26 (1983), pp. 25-52.

[27] I. H. SUDBOROUGH AND J. TURNER, *On computing the width and black/white pebble demand of trees*, Technical Report, Electrical Engineering and Computer Science Dept., Northwestern Univ., Evanston, 1983.

[28] S. TRIMBERG, *Automating chip layout*, IEEE Spectrum, 1982, pp. 38-45.

[29] A. WEINBERGER, *Large scale integration of* MOS *complex logic: A layout method*, IEEE J. Solid State Circuits, SC-2 (1967), pp. 182-190.

[30] H. YOSHIZAWA, H. KAWANISHI AND K. KANI, *A hueristic procedure for ordering* MOS *arrays*, Proc. Design Automation Conference, 1975, pp. 384-393.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 3, July 1985

011

CODING STRINGS BY PAIRS OF STRINGS*

F. R. K. CHUNG†, R. E. TARJAN†‡, W. J. PAUL† AND R. REISCHUK§

Abstract. Let $X, Y \subset \{0,1\}^*$. We say Y codes X if every $x \in X$ can be obtained by applying a short program to some $y \in Y$. We are interested in sets Y that code X robustly in the sense that even if we delete an arbitrary subset $Y' \subset Y$ of size k, say, the remaining set of strings $Y \setminus Y'$ still codes X. In general, this can be achieved only by making in some sense more than k copies of each $x \in X$ and distributing these copies on different strings Y. Thus if the strings in X and Y have the same length, then $\# Y \geqq (k+1)\# X$.

If we allow coding of X by Y in a way that every $x \in X$ is obtained from strings $x, z \in Y$ by application of a short program, then we can do better.

Let $Y = \{\bigoplus_{x \in S} x | S \subset X\}$ where $\bigoplus$ denotes bitwise sum mod 2. Then $\# Y = 2^{\# X}$. Yet Y codes X robustly for $k = 2^{\# X - 1} - 1$. This paper explores the limitations of coding schemes of this nature.

1. Robust coding of strings by strings. For strings $x, y \in \{0,1\}^*$, we denote by $K(x|y)$ the Kolmogorov complexity of x given y [P], [ZL]. We say *y codes x* if $K(x|y) = O(\log |x|)$. We deliberately leave the implicit constant in the O-notation undefined. Let $X, Y \subset \{0,1\}^*$. We say *Y 1-codes X* if for all $x \in X$ there is $y \in Y$ such that y codes x. We say *Y codes X k-robustly* if for all $Y' \subset Y$ with $\# Y' \leqq k$ the set of strings $Y \setminus Y'$ still 1-codes X.

Assume that the strings $x \in X$ are of the same length and sufficiently irregular, that the strings in Y are longer than the strings in X by a factor α, and that there are β times more strings in Y than in X. Then one would intuitively expect every $y \in Y$ to code at most α strings $x \in X$, and most strings $x \in X$ are coded by at most $\alpha\beta$ strings $y \in Y$. This is more or less confirmed by Lemma 1.

LEMMA 1. *Let $p \gg \alpha \log np$. Let $X = \{x_1, \cdots, x_n\} \subset \{0,1\}^p$, $Y = \{y_1, \cdots, y_{\beta n}\} \subset \{0,1\}^{\alpha p}$ and $K(x_1 \cdots x_n) \geqq np$ (i.e. $x_1 \cdots x_n$ is a random string). Then*

(a) *Each of $y \in Y$ codes at most α strings $x \in X$.*

(b) *Each of at least $n/2$ strings $x \in X$ is coded by at most $2\alpha\beta$ strings $y \in Y$.*

Proof. Let $\{i_1, \cdots, i_s\} \subset \{1, \cdots, n\}$. Then

(1) $sp - O(s \log n) \leqq K(x_{i_1} \cdots x_{i_s})$ because $x_1 \cdots x_n$ is random [P, fact 5].

Suppose $y \in Y$ codes $x_{i_1}, \cdots, x_{i_s}$. Then

(2) $K(x_{i_1} \cdots x_{i_s}) \leqq \sum (K(x_{i_j}|y) + O(\log K(x_{i_j}|y)) + K(y) \leqq O(s \log p) + \alpha p$.

For $s = \alpha + 1$, (1) and (2) imply $(\alpha+1)p - O(\alpha \log n) \leqq \alpha p + O(\alpha \log p)$. Hence $p - O(\alpha \log np) \leqq 0$. This proves (a).

Suppose (b) is false. Then

$$\begin{aligned} \alpha\beta n &\geqq \sum_j \#\{x | y_j \text{ codes } x\} && \text{by (a)} \\ &= \sum_i \#\{y | y \text{ codes } x_i\} \\ &> (n/2)2\alpha\beta = \alpha\beta n && \text{by assumption.} \end{aligned}$$

□

Clearly, it makes sense to say that, for every $x \in X$, certain strings $y \in Y$ carry specific information about x—namely those strings y that code x. By Lemma 1, if the strings in X are messy, then every string y carries specific information about a small number of strings in X. Moreover, if one deletes from Y all strings carrying specific

* Received by the editors September 22, 1983 and in final form April 4, 1984.

† AT&T Bell Laboratories, Murray Hill, New Jersey 07974.

‡ Part of this research was done while the second author was visiting the University of Bielefeld.

§ IBM Research Laboratory San Jose, California 95193.

information about a particular string $x \in X$, then the resulting set of strings does not 1-code $\{x\}$ any more. Thus we have:

COROLLARY 1. *If under the hypotheses of Lemma* 1, Y *1-codes* X *k-robustly, then* $2\alpha\beta > k$.

2. Simple coding of strings by pairs of strings. For $y, z \in \{0,1\}^p$, let $y \oplus z \in \{0,1\}^p$ be the string whose ith bit is the mod 2 sum of the ith bits of y and z for $1 \leqq i \leqq p$. For $1 \leqq i \leqq p$, let $e_i \in \{0,1\}^p$ be the string which has 1 in the ith position and 0's in all other positions. Let $E_p = \{e_1, \cdots, e_p\}$. Let $\mathbf{0} \in \{0,1\}^p$ be the string consisting of p 0's.

Let $X, Y \subset \{0,1\}^p$. We say Y *simply 2-codes* X if for all $x \in X$ there are two strings $y, z \in Y$ such that $x = z \oplus y$. We say Y simply 2-codes X k-robustly if for all $Y' \subset Y$ with $\# Y' \leqq k$ the set of strings $Y \setminus Y'$ simply 2-codes X.

Example 1. $X = E_p$, $Y = \{y_1, \cdots, y_{p+1}\}$, with $y_i = e_i$ for $i \leqq p$ and $Y_{p+1} = \mathbf{0}$.

Intuition suggests that in this example for $i \leqq p$, the string y_i carries specific information about e_i and about no other strings in X.

Example 2. $X = E_p$, $Y = \{y_1, \cdots, y_{p+1}\}$, with $y_i = \bigoplus_{j \neq i} x_j$ for $i \leqq p$ and $y_{p+1} = \bigoplus_{j=1}^{p} x_i$.

Is there still a reasonable way to attribute to every string $y \in Y$ specific information about a small number of strings $x \in X$? Motivated by this question, we consider for arbitrary X, $Y \subset \{0,1\}^p$ the following edge-labelled graph $G(X, Y) = (V, E, L)$: $V = Y$ is the vertex set. For all $y, z \in Y$, there is an edge $\{y, z\} \in E$ iff $y \oplus z = x$ for some $x \in X$. $L: E \to X$ is a mapping that labels every edge $e = \{y, z\}$ with $L(e) = y \oplus z$. For X, Y, as in Examples 1 and 2, we get the graph of Fig. 1.

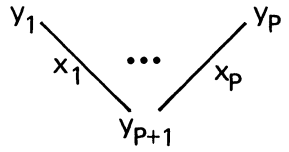


FIG. 1

Transform the edge labelling $L: E \to X$ into a node labelling by the following rule:

(*) For every edge $e = \{y, z\}$, put label $L(e)$ on node y or on node z.

There are many ways to do this, and in general, nodes may get more than one label. Thus the resulting node labelling is a mapping from Y to the power set of X. We will use the letter L both for edge and node labellings.

If an edge labelling L has been transformed by Rule (*) into a node labelling L', then for every $x \in X$, the set of strings $Y' = \{y | x \in L(y)\}$ has the property of a set of strings each of which carries specific information about x: $Y \setminus Y'$ does not simply 2-code $\{x\}$. In analogy with the case of 1-coding, we want each $y \in Y$ to carry specific information about only a small number of strings $x \in X$. Thus we are interested in node labellings L that minimize

$$\max_{y \in Y} \# L(y).$$

For edge-labelled graphs, $G = (V, E, L)$, let

$$l(G) = \min_{L'} \max_{v \in V} \# L'(v),$$

where the minimum is taken over all node labellings L' that can be obtained from L by Rule (*). For the graph G of Fig. 1, we have $l(G) = 1$, which is obtained by the node labelling $L'(y_i) = \{x_i\}$ for $i \leqq p$ and $L'(y_{p+1}) = \varnothing$.

3. Transformation of labellings for simple 2-coding. We define the labelled p-dimensional cube $C_p := G(E_p, \{0,1\}^p)$. If $Y \subset \{0,1\}^p$, then $G(E_p, Y)$ is a subgraph of C_p.

Any node labelling of C_p has to distribute $p2^{p-1}$ occurrences of labels among 2^p nodes. As for every node v in C_p, different edges incident with v have different labels, we find $l(C_p) \geqq p/2$. This shows that the function $l(\cdot)$ is unbounded. As pointed out in the abstract, coding $X = \{x_1, \cdots, x_n\}$ by $Y = \{\bigoplus_{i \in I} x_i | I \subset \{1, \cdots, n\}\}$ works for arbitrary $X \subset \{0,1\}^p$. Thus in the case of simple 2-coding, Lemma 1 and Corollary 1 do not hold, and one has better robust coding schemes than in the case of 1-coding. However, we have

LEMMA 2. *For all p and m, if $y \subset \{0,1\}^p$ and $\# y \leqq m$, then $l(G(E_p, Y)) \leqq \log m$.*

Proof. The proof is by induction on p. For $p = 1$, this is easily verified. Suppose the lemma holds for p. Let $Y \subset \{0,1\}^{p+1}$. For $i = 0, 1$, let $Y_i = \{y \in Y | y_{p+1} = i\}$ and $m_i = \# Y_i$. Then $l(G(E_{p+1}, Y_i)) \leqq \log m_i$ for $i = 0, 1$ by the induction hypothesis. Assume $m_0 \leqq m_1$. For any edge $\{y, z\}$ with $y \in Y_0$, $z \in Y_1$, put the edge label e_{p+1} of edge $\{y, z\}$ on y. This gives

$$l(G(E_{p+1}, y)) \leqq \max\{1 + l(G(E_{p+1}, Y_0)), l(G(E_{p+1}, Y_1))\}$$

$$\leqq \max\left\{1 + \log \frac{m}{2}, \log m\right\}. \qquad \square$$

COROLLARY 2. *Let $Y \subset \{0,1\}^p$, $\# Y = m$. For at least $p/2$ strings $e_i \in E_p$, there is a set $Y_i \subset Y$ such that $\# Y_i \leqq (2m \log m)/p$ and $Y \setminus Y_i$ does not simply 2-code $\{e_i\}$.*

Proof. Assume the corollary is false. Let L be the node labelling of $G(E_p, Y)$ constructed in the proof of Lemma 2. Then

$$m \log m \geqq \sum_{y \in Y} \# L(y) = \sum_i \#\{y | e_i \in L(y)\}$$

$$> (p/2)(2m \log m)/p. \qquad \square$$

COROLLARY 3. *Let $Y \subset \{0,1\}^p$, $\# Y = m$, and let Y simply 2-code E_p k-robustly. Then $(2m \log m)/p > k$.*

4. General 2-coding and the associated graphs. Let $x, y, z \in \{0,1\}^*$. We say y and z 2-code x if $K(x|yz) = O(\log |x|)$. Let $X, Y \subset \{0,1\}^*$. We say *Y 2-codes X* if for all $x \in X$, there are $y, z \in Y$ such that y and z 2-code x. We say *Y 2-codes X k-robustly* if for all $Y' \subset Y$ with $\# Y' \leqq k$, the set of strings $Y \setminus Y'$ 2-codes X.

With $X, Y \subset \{0,1\}^*$, we associate again an edge-labelled graph $G(X, Y) = (Y, E, L)$: for each $y, z \in Y$ there is an edge $\{y, z\} \in E$ iff y and z 2-code some $x \in X$. For each edge $e = \{y, z\} \in E$, we set $L(e) = \{x \in X | y$ and z 2-code $x\}$. Thus L is now a mapping from E into the power set of X. For $E' \subset E$, let

$$L(E') = \bigcup_{e \in E'} L(e).$$

The following lemma exhibits a graph theoretic property of the graphs $G(x, y)$ and their subgraphs,

LEMMA 3. *Let $X = \{x_1, \cdots, x_n\} \subset \{0,1\}^p$, $Y = \{Y_1, \cdots, Y_{bn}\} \subset \{0,1\}^{ap}$ and $G(X, Y) = (Y, E, L)$. Let $K(x_1, \cdots, x_n) \geqq np$. Then*

$$\# L(E) \leqq \# Y \frac{a}{1 - O(\log(p \# Y))/p}.$$

Proof. Let $d = \# L(E)$ and let $L(E) = \{x_{i_1}, \cdots, x_{i_d}\}$. Then

(3) $dp - O(d \log n) \leqq K(x_{i_1} \cdots x_{i_d})$.

The string $x_{i_1} \cdots x_{i_d}$ can be specified in the following way:

- The binary representations of n and b.
- For each $j \in \{1, \cdots, d\}$ the binary representation of two indices k and 1 such that $K(x_{i_j} | y_k y_l) = O(\log p)$ and a program that produces x_{i_j} from $y_k y_l$.
- The bits of $y_1 \cdots y_{bn}$.

Thus

(4) $K(x_{i_1} \cdots x_{i_d}) \leqq O(d \log bn) + O(d \log p) + abnp.$

(3) and (4) imply the lemma. □

Two cases are particularly simple:

(i) $O(\log bnp)/p < c < 1$ for some fixed c. Then $\# L(E) = O(\# Y)$.

(ii) $a = 1$ and $\# Y/(1 - O(\log p \# Y)/p) < \# Y + 1$. Then $\# L(E) \leqq \# Y$.

We now give an example of an edge-labelled graph G such that $\# L(E) \leqq \# Y$ holds for all subgraphs (Y, E, L) of G, yet $G \neq G(X, Y)$ for any X, Y, to which case (ii) applies (if p is large enough).

Let G_1 be a single edge with label x_1. For $i \geqq 1$, let G_i^1, G_i^2 be two copies of G_i. Connect every vertex of G_i^1 with every vertex of G_i^2 with an edge labelled x_{i+1}. Call the resulting graph G_{i+1}. By induction on i, one easily verifies that $\# L(E) \leqq \# V - 1$ for any subgraph (V, E, L) of G_i.

Suppose G_8 is a subgraph of $G(X, Y)$. Consider any node y in G_8. Then $K(x_i|y) > 2p/3 - O(\log p)$ for some $i \in \{5, \cdots, 8\}$. Otherwise one gets the contradiction

$$4p - O(\log p) \leqq K(x_5 \cdots x_8) \leqq \sum_{i=5}^{8} (K(x_i|y) + O(\log p)) + K(y)$$
$$\leqq \frac{11p}{3} + O(\log p).$$

Consider in G_8 the subgraph drawn in Fig. 2. For all $j \in \{1, \cdots, 5\}$, we have

$$K(z_j|yx_i) \leqq K(yx_iz_j) - K(yx_i) + O(\log p)$$
$$\leqq K(yz_j) + K(x_i|yz_j) - K(yx_i) + O(\log p)$$
$$\leqq K(y) + K(z_j|y) - K(y) - K(x_i|y) + O(\log p)$$
$$\leqq p - \frac{2p}{3} + O(\log p). \qquad \text{[ZL]}$$

This gives the contradiction

$$4p - O(\log p) \leqq K(x_1 \cdots x_4)$$
$$\leqq K(yx_i) + \sum_{j=1}^{5} K(z_j|yx_i) + \sum_{j=1}^{4} K(x_j|z_jz_{j+1}) + O(\log p)$$
$$\leqq \left(2 + \frac{5}{3}\right)p + O(\log p).$$

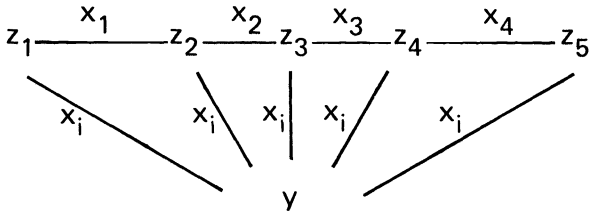


FIG. 2

5. Transforming edge labellings into node labellings. For sets V, V', let $V \otimes V' = \{\{v, v'\} | v \in V, v' \in V'\}$.

THEOREM 1. *Let $G = (V, E, L)$ be an edge-labelled graph, let $\# V = n$ and for all $V' \subset V$, let $\# L((V' \otimes V') \cap E) \leqq \# V'$. Then $l(G) \leqq \alpha\sqrt{n}$ where $\alpha = 2\sqrt{6}$.*

Proof. The proof is by induction on n. The theorem is true for $n \leqq \alpha$. Let $n > \alpha$. Find a node $u \in V$ such that $\# L((\{u\} \otimes V) \cap E) \geqq \alpha\sqrt{n}$ (if no such node exists, the nodes of G can be trivially labelled in the desired way). Let E_1 be a smallest set of edges adjacent to u such that $\# L(E_1) \geqq \alpha\sqrt{n}$. By hypothesis we have $\alpha\sqrt{n} - 1 \leqq \# E_1 \leqq \alpha\sqrt{n}$. Let V_1 be the set of end points of edges in E_1 other than u.

Let $V_2 = V \setminus (V_1 \cup \{u\})$. Let $E_2 = (V_1 \otimes V_2) \cap E$ and $E_3 = (u \otimes V_2) \cap E$ (see Fig. 3). Ignoring labels on edges in E_1 and E_2, we can label the nodes in V_1 with

$$\alpha\sqrt{\# V_1} \leqq \alpha\sqrt{\alpha\sqrt{n}} \leqq \alpha\sqrt{n} - 2$$

labels per node. By hypothesis, every edge in E has at most 2 labels. Thus putting labels on edges in E_1 on the endpoint of these edges in V_1 gives at most 2 extra labels per node in V_1.

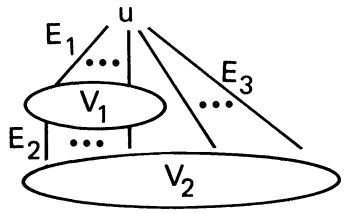


FIG. 3

Ignoring labels on edges in E_2 and E_3, we can label the nodes of V_2 with

$$\alpha\sqrt{\# V_2} \leqq \alpha\sqrt{n - \alpha\sqrt{n}} + 1 \leqq \alpha\left(\sqrt{n} - \frac{\alpha\sqrt{n} - 1}{2\sqrt{n}}\right)$$

$$\leqq \alpha\sqrt{n} - \frac{\alpha^2}{2} + 1 \leqq \alpha\sqrt{n} - 11$$

labels per node. Putting labels on edges in E_3 to the endpoints of these edges in V_2 gives at most 2 extra labels per node in V_2.

Now for every label x on an edge e in $V_1 \otimes V_2$ that has already been put by the above operations on the endpoint of e in V_1, delete label x from edge e. We continue to use the letter L for the modified edge labelling.

The theorem follows if we establish

LEMMA 4. *For evey node $w \in V_2$, we have*

$$\# L((w \otimes V_1) \cap E) \leqq 9.$$

Proof. Assume the lemma is false for node w. Let $V_3 \subset V_1$ be a smallest set of nodes such that $\# L((w \otimes V_3) \cap E) \geqq 10$ (Fig. 4). We make three observations:

(i) $\# V_3 \geqq 9$.

(ii) Let $V_4 \subset V_3$ and $z \in V_3 \setminus V_4$. Then

$$L(\{z, u\}) \setminus L(\{V_4 \otimes u\}) \neq \varnothing,$$

$$L(\{z, w\}) \setminus L(\{V_4 \otimes w\}) \neq \varnothing$$

by the minimality of V_1 and V_3.

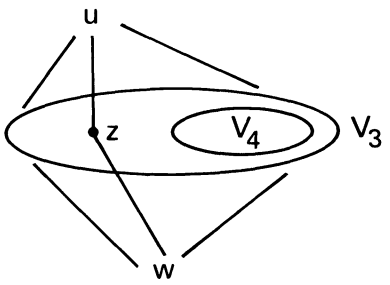


FIG. 4

(iii) Let $V_4 \subset V_3$, $\# V_4 \leqq 2$. Then $\# L(u \otimes V_4) \leqq 3$ and $\# L(w \otimes V_4) \leqq 3$. By (ii) we have

(5) $L(\{w, z\}) \subset L(\{w, u\} \otimes V_4)$ for at most 3 nodes $z \in V_3 \setminus V_4$. Similarly

(6) $L(\{u, z\}) \subset L(\{w, u\} \otimes V_4)$ for at most 3 nodes $z \in V_3 \setminus V_4$.

By (i) there is $z \in V_3 \setminus V_4$ such that (5) and (6) both do not hold for z.

But $L(\{u, z\}) \cap L(\{w, z\}) = \varnothing$, because labels from this intersection have already been deleted from the edge $\{w, z\}$. Thus

$$\# L((\{z\} \cup V_4) \otimes \{u, w\}) \geqq \# L(V_4 \otimes \{u, w\}) + 2.$$

Starting with $V_4 = \varnothing$ and carrying out this construction 3 times gives a set of 3 nodes z_1, z_2, z_3 such that

$$6 \leqq \# L(\{z_1, z_2, z_3\} \otimes \{u, w\}) \leqq 5. \qquad \square$$

COROLLARY 4. *Let* $X = \{x_1, \cdots, x_n\} \subset \{0, 1\}^p$, $Y = \{y_1, \cdots, y_{bn}\} \subset \{0, 1\}^p$, *let* $K(x_1 \cdots x_n) \geqq np$, *let* $(1 - O(\log p \# Y)/p) \leqq 1 + 1/\# Y$, *and suppose* Y 2*-codes* X k*-robustly. Then* $4\sqrt{6 \# Y} > k$.

THEOREM 2. *Let* $G = (V, E, L)$ *be an edge-labelled graph,* $\# V = n \gg 1$, *and for all* $V' \subset V$, *let* $\# L((V' \otimes V') \cap E) \leqq c \# V'$. *Then* $l(G) \leqq 4\ cn^{1-\varepsilon}$ *where* $\varepsilon < 1/(12c)$.

Proof. By hypothesis, every edge has at most $2c$ labels. We show $l(G) \leqq 2n^{1-\varepsilon}$ if every edge has at most 1 label.

For every node $v \in V$ and any edge label l that occurs on at least $n^{\varepsilon} + 1$ of the edges adjacent to v, put label l on v and delete it from the edges adjacent to v. By this at most $n^{1-\varepsilon}$ labels are put on every node.

Next, for each $v \in V$, partition the edges adjacent to v into $\tau = n^{\varepsilon}$ classes $E_v^1, \cdots, E_v^{\tau}$ such that in every class E_v^i, every label occurs on at most one edge of E_v^i. Partition E into classes $E^{i,j}$, $1 \leqq i \leqq j \leqq n^{\varepsilon}$, by $\{u, v\} \in E^{i,j}$ if $[u, v] \in E_v^i \cap E_u^j$. For all i, j, let $G^{i,j} = (V, E^{i,j}, L^{i,j})$ where $L^{i,j}$ is L restricted to $E^{i,j}$. Then in $G^{i,j}$ for every node v, all edges adjacent to v have different labels. We will show $l(G^{i,j}) \leqq n^{1-3\varepsilon}$.

For every vertex v that is adjacent to at most $n^{1-3\varepsilon}$ edges, put all labels occurring on these edges on v. Delete v and its adjacent edges from $G^{i,j}$. Continue this process as long as possible. If finally all of $G^{i,j}$ is deleted, we are done. Otherwise we are left with an edge-labelled graph $G' = (V', E', L')$ with at most n nodes. Every node v has at least $n^{1-3\varepsilon}$ neighbors and the edges joining v with its neighbors have all different labels. We will derive a contradiction from this.

We consider the adjacency matrix A' of G' and use the following fact [H].

For natural numbers m, n, j, k, let $z(m, n, j, k)$ be the smallest number z' such that every $m \times n$ matrix with z' ones contains a $j \times k$ minor μ that consists of ones

only. Then $z(m, n, j, k) \leqq 1 + km + (j-1)^{1/k} m^{1-1/k} n$. In particular,

$$z(n, n, 9c^3, 2c) \leqq 1 + 2cn + (9c^3)^{1/(2c)} n^{2-1/(2c)} \leqq n^{2-6\varepsilon}$$

if $6\varepsilon < 1/(2c)$ and n is large enough.

Let μ' be a $9c^3 \times (2c)$ minor of A' that consists only of ones. Every one in μ' corresponds to an edge $e \in E^{i,j}$. Replace each one in μ' by the label $L^{i,j}(e)$ of the corresponding edge. Call the resulting matrix μ. Every label occurs in each row and column of μ at most once. We make the following observation.

If R is a set of at most $2c$ rows of μ, then at most $4c^2$ different labels occur in R. Each label occurs in at most $2c$ more rows of μ. Thus there is a row r of R consisting only of labels that are not yet in R. Starting with $R =$ an arbitrary row of M and repeating this process $2c$ times gives a $(2c+1) \times (2c)$ minor M'' of μ that contains $4c^2 + 2c$ different labels. The rows and columns of μ' correspond to a set V' of $4c+1$ vertices of μ. Thus

$$2c(2c+1) \leqq \# L((V' \otimes V') \cap E) \leqq c(4c+1). \qquad \square$$

COROLLARY 5. *Let* $X = \{x_1, \cdots, x_n\} \subset \{0,1\}^p$, $Y = \{Y_1, \cdots, Y_{bn}\} \subset \{0,1\}^{ap}$ *and* $p \gg \log bnp$. *Suppose* $K(x_1 \cdots x_n) \geqq pn$ *and* Y *2-codes* X *k-robustly. Then* $16a(bn)^{1-\varepsilon} > k$ *for* $\varepsilon < 1/(24a)$ *and* n *large enough.*

6. A lower bound. We want to establish lower bounds for $l(G)$ where G satisfies the property in Theorem 2.

THEOREM 3. *For* $c \geqq 2$, *there exists* $G = (V, E, L)$, $\# V = n$, *having an edge labelling satisfying*

$$\# L((V' \otimes V') \cap E) \leqq c \# V' \quad \textit{for all } V' \subset V$$

and

$$l(G) \geqq c' n^a \quad \textit{where } a < \frac{1}{2} - \frac{1}{4c-2}.$$

Proof. Let $\delta = (\frac{1}{2} - 1/(4c-2) - a)/2$. Let $\alpha = \frac{1}{2} - 1/(4c-2) - \delta = (c-1)/(2c-1) - \delta$ and let $G = (V, E)$ be a random graph with n nodes and $n^{1+\alpha}$ edges, where all such graphs are equally likely. We first show that with high probability,

$$(**) \qquad \#(V' \otimes V') \cap E \leqq c \# V' \quad \text{for all } V' \subset V \quad \text{with } \# V' \leqq n^{\alpha}.$$

The probability that for any set V' of cardinality $j \leqq n^{\alpha}$, $(**)$ does not hold, is at most

$$\begin{aligned} W_j &= \binom{n}{j} \binom{\binom{j}{2}}{cj} \binom{\binom{n}{2} - cj}{n^{1+\alpha} - cj} \Bigg/ \binom{\binom{n}{2}}{n^{1+\alpha}} \\ &\leqq \left(\frac{ne}{j}\right)^j \left(\frac{ej^2}{2cj}\right)^{cj} \frac{n^{1+\alpha} \cdots (n^{1+\alpha} - cj + 1)}{\binom{n}{2} \cdots \left(\binom{n}{2} - cj + 1\right)} \\ &\leqq \left(\frac{ne}{j}\right)^j \left(\frac{ej}{2c}\right)^{cj} (2.1 n^{-1+\alpha})^{cj} \\ &\leqq (c_2 j^{1-1/c} n^{-1+\alpha+1/c})^{cj}. \end{aligned}$$

For $2c+2 \leqq j \leqq \log n$, we estimate

$$W_j \leqq (\log^2 n \cdot n^{-1/2-1/(4c-2)-\delta+1/c})^{cj} \leqq (n^{-1/2-1/6+1/2})^{2.6} = n^{-2}.$$

For $\log n < j \leqq n^\alpha$, we have

$$\begin{aligned} W_j &\leqq (c_2 n^{\alpha(1-1/c)-1+\alpha+1/c})^{cj} \\ &= (c_2 n^{(\alpha(2c-1)-c+1)/c})^{cj} \\ &= (c_2 n^{-\delta(2c-1)/c})^{cj} \leqq (n^{-c_3})^{\log n} \leqq n^{-2}. \end{aligned}$$

Hence the probability that (**) does not hold is at most

$$\sum_{j=2c+2}^{n^\alpha} W_j \leqq n^\alpha n^{-2} \leqq n^{-1}.$$

Next we make use of the fact that with probability $1-o(n^{-1})$, the degree of every node in G is bounded by $3n^\alpha$ [ER]. Therefore there exists a graph G with n nodes, $n^{1+\alpha}$ edges, such that the degree of every node in G is bounded by $3n^\alpha$ and (**) holds for G.

Let L be any edge labelling of G, which labels every edge with exactly 1 label $l \in \{1, \cdots, n^\alpha\}$. Let $V' \subset V$. Then $\#L((V' \otimes V') \cap E) \leqq \min\{n^\alpha, \#((V' \otimes V') \cap E\}) \leqq c\#V'$.

Suppose we choose L randomly in such a way that edges are labelled independently, and such that for each edge, each label is equally likely. Let v be any node of G, let d be the degree of v and let l be any label. Then the probability that j or more edges adjacent to v have label l is at most

$$\binom{d}{j}\left(\frac{1}{n^\alpha}\right)^j \leqq \left(\frac{de}{j}\right)^j \left(\frac{1}{n^\alpha}\right)^j \leqq \left(\frac{3n^\alpha e}{j}\right)^j \left(\frac{1}{n^\alpha}\right)^j = \left(\frac{3e}{j}\right)^j = O(n^{-3})$$

if $j \geqq \log n$. Therefore the probability that $\log n$ or more edges adjacent to the same node as G have the same label is at most $n \cdot n^\alpha \cdot O(n^{-3}) = O(n^{-1})$. Hence there is a labelling L such that for every l and V label, l occurs on at most $\log n$ edges adjacent to node V. No matter how we transform L into a node labelling L', we have $\sum_v \#L'(v) \geqq n^{1+\alpha}/\log n$. This proves the theorem. □

7. Simple 2-coding revisited. If Y is a subset of $\{0,1\}^p$ of size m, then $G(E_p, Y)$ may have up to $m \log m$ labels. This means the number of pairs in Y that code some e_i grows faster than the size of Y. But at least for the obvious example to demonstrate this, the $(\log m)$-dimensional subcube, one notices that for $\log m \ll p$, only a small subset of the e_i can be coded by many pairs. Thus there is hope that, disregarding a small subset of $\{e_1, \cdots, e_p\}$, the remaining e_i have a much smaller number of pairs which simply 2-code them.

For $X \subset \{0,1\}^p$ and $1 \leqq i \leqq p$, define $r(X, i)$ as the number of edges in $G(E_p, X)$ with label i.

For $1 \leqq k \leqq p$ define

$$r_k(X) = \min_{\substack{D \subset \{1,\cdots,p\} \\ |D| \geqq k}} \sum_{i \in D} r(X, i)$$

and

$$\rho_k(X) = \min_{\substack{D \subset \{1,\cdots,p\} \\ |D| \geqq k}} \max_{i \in D} r(X, i),$$

and for $m \in IN$,

$$r_k(m) = \max_{|X| \leqq m} r_k(X)$$

and

$$\rho_k(m) = \max_{|X| \leqq m} \rho_k(X).$$

One checks easily that for $m \leqq 2^p$, $\rho_p(m) = \lfloor m/2 \rfloor$, whereas from Lemma 2 it follows that $r_p(m) = \theta(m \log m)$.

Let $\ln(x)$ denote the natural logarithm of x and $\ln^k(x) = [\ln(x)]^k$.

THEOREM 4. *There are constants $\alpha > 0$ and C, $h \geqq 1$ such that for all $1 \leqq k < p$ and $m/(p-k) \geqq C/\alpha$,*

$$\rho_k(m) \leqq \alpha \frac{m}{p-k} \ln^3\left(\alpha \frac{m}{p-k} + h\right).$$

COROLLARY 6. *For any $\varepsilon > 0$ and $m = O(p)$,*

$$\rho_{(1-\varepsilon)p}(m) = O(1).$$

Theorem 4 follows from the following:

LEMMA 5. *There are constants $\beta > 0$, $h \geqq 1$ such that for any $X \subset \{0,1\}^p$,*

$$\sum_{i=1}^{p} \frac{r(X,i)}{\ln^3(r(X,i)+h)} \leqq \beta |X|.$$

Proof of Theorem 4. Assume

$$\rho_k(m) > r := \alpha \frac{m}{p-k} \ln^3\left(\alpha \frac{m}{p-k} + h\right).$$

Then there exists $X \subset \{0,1\}^p$ of size m such that $r(X,i) := r_i > r$ holds for more than $p-k$ labels $i \in \{1, \cdots, p\}$. Define

$$F(x) = \frac{x}{\ln^3(x+h)}.$$

Later it will be shown that for appropriate $h \geqq e^2$, $F(x)$ is monotonically increasing for $x \geqq 0$. Hence,

$$\sum_{i=1}^{p} F(r_i) \geqq \sum_{i} \sum_{r_i > r} F(r_i) > (p-k)F(r)$$

$$= \alpha m \frac{\ln^3(\alpha(m/(p-k))+h)}{\ln^3(\alpha(m/(p-k)) \ln^3(\alpha(m/(p-k))+h)+h)}.$$

For an appropriate $C \geqq 1$,

$$x + h \geqq \ln^3(x+h)$$

holds for all $x \geqq C$. Thus if $\alpha m/(p-k) \geqq C$, then

$$\ln^3\left(\frac{\alpha m}{p-k} \ln^3\left(\frac{\alpha m}{p-k} + h\right) + h\right) \leqq \ln^3\left(\frac{\alpha m}{p-k}\left(\frac{\alpha m}{p-k} + h\right) + h\right)$$

$$\leqq \ln^3\left(\left(\frac{\alpha m}{p-k} + h\right)^2\right) = 8 \ln^3\left(\frac{\alpha m}{p-k} + h\right).$$

Therefore, $\sum_{i=1}^{p} F(r_i) > \alpha m/8$. But this contradicts Lemma 5 if $\alpha/8 \geqq \beta$. □

Proof of Lemma 5. Define $h = e^2 \approx 7.389$ and $\gamma = 0.16$.

$$g(x) = \gamma \frac{x}{\ln^3 (x+h)}, \quad \text{for } x \geqq 0$$

and

$$f(n) = 1 + \sum_{m=2}^{n} \frac{1}{m \ln^2 (m)}, \quad \text{for } n \in \mathbb{N}, \quad n \geqq 1.$$

We will show in the Appendix:

(g1) $0 \leqq g(x) \leqq 0.16x$ for all $x \geqq 0$,

(g2) $g'(x) \geqq 0$ for all $x \geqq 0$,

(g3) $g''(x) \leqq 0$ for all $x \geqq 0$,

(f1) $1 \leqq f(n) \leqq f(n+1) \leqq 5$ for all $n \geqq 1$.

Now let $X \subset \{0, 1\}^p$. Lemma 5 follows from the following:

PROPOSITION. *If* $n = |\{i | r_i > 0\}|$, *then*

$$\sum_{i=1}^{p} g(r_i) \leqq f(n)|X|.$$

Proof. The proof is by induction on n. Define $r = \max_{1 \leqq i \leqq p} r_i$. For each i, the edges with label i are a matching. Hence, $|X| \geqq 2r$. For all $1 \leqq h \leqq n_0 = 98$, we get

$$\sum_{i=1}^{p} g(r_i) \leqq ng(r) \leqq 98\gamma \frac{r}{\ln^3 (r+h)} \leqq \frac{49\gamma |X|}{\ln^3 (r+h)} \leqq \frac{49\gamma}{2^3}|X| \leqq |X| \leqq f(n)|X|.$$

Thus, the claim holds for all $n \leqq n_0$. Now assume

(7.1) $$n+1 > n_0 = 98,$$

and the claim is true for all $n' \leqq n_0$. We may assume that $r_1 \geqq r_2 \geqq \cdots \geqq r_{n+1} > r_{n+2} = \cdots = r_p = 0$. Define for $l \in \{0, 1\}$,

$$X^l = \{x \in X | x_{n+1} = l\},$$

and for $1 \leqq i \leqq n$ r_i^l as the number of edges in $G(E_p, X^l)$ with label i, this means we cut X in dimension $n+1$. Obviously,

(7.2) $$X = X^0 \cup X^1, \qquad r_i = r_i^0 + r_i^1 \quad \text{for } 1 \leqq i \leqq n.$$

$D = D^0 \cap D^1$ and $d^l = |D^l|$. One can check easily

(7.3) $$|X^l| \geqq \max\{r_{n+1}, d^l + 1\} \quad \text{for } l = 0, 1.$$

Define $\Delta g(x, y) = g(x) + g(y) - g(x+y)$. Now

$$\sum_{i=1}^{p} g(r_i) = \sum_{i=1}^{n+1} g(r_i) = \sum_{i=1}^{n} g(r_1^0) + \sum_{i=1}^{n} g(r_i^1) - \sum_{i \in D} \Delta g(r_i^0, r_i^1) + g(r_{n+1}).$$

Applying the induction hypothesis to X^0 and X^1 gives

(7.4) $$\sum_{i=1}^{p} g(r_i) \leqq |X^0| f(d^0) + |X^1| f(d^1) - \sum_{i \in D} \Delta g(r_i^0, r_i^1) + g(r_{n+1}).$$

The idea of the proof is as follows: if D is large, then $\sum_{i \in D} \Delta g(r_0^0, r_i^1)$ is large enough to compensate the term $g(r_{n+1})$; otherwise one of the d^l must be relatively small, such

that the difference between $|X^l|f(n+1)$ and $|X^l|f(d^l)$ is bigger than $g(r_{n+1})$. We have to distinguish several cases. First, we state some more properties of f and g which will be proved in the appendix.

(g4) $\Delta g(x, y) \geqq 0$ for all $x, y \geqq 0$,

(g5) $\Delta g(x, y) \leqq \Delta g(x, z)$ for all $0 \leqq x$ and $0 \leqq y \leqq z$,

(g6) $\Delta g(1, 1) \geqq 0.0298\gamma$,

(g7) $$\Delta g(x, y) \geqq 1.4 \frac{g(x)}{\ln (x+h)}$$ for all $0 \leqq x \leqq y$ and $y \geqq 3h$.

Define $\delta f(n, m) = f(n) - f(m)$ for $1 \leqq m \leqq n$. Then

(f2) $$\delta f(n, m) \geqq \frac{1}{4} \frac{1}{\ln^2 (m+h)}$$ for all $16 \leqq m \leqq \frac{2}{3}n$.

Case 1. $\exists l$ with $d^l \leqq 2/3n$. Assume $l = 1$. Then (7.4) yields

$$\sum_{i=1}^{p} g(r_i) \leqq |X^0|f(d^0) + |X^1|f(d^1) + g(r_{n+1})$$
$$\leqq (|X^0| + |X^1|)f(n+1) + g(r_{n+1}) - |X^1|\,\delta f(n+1, d^1).$$

If $d^1 \geqq 16$ and $d^1 \geqq r_{n+1}$, we get

$$g(r_{n+1}) - |X^1|\,\delta f(n+1, d^1) \leqq g(r_{n+1}) - d^1 \frac{1}{4} \frac{1}{\ln^2 (d^1+h)} \quad \text{by (7.3) and (f2)}$$
$$\leqq g(r_{n+1}) - \gamma \frac{d^1}{\ln^3 (d^1+h)} \quad \text{since } \frac{1}{4} \geqq \frac{\gamma}{\ln (d^1+h)}$$
$$= g(r_{n+1}) - g(d^1) \leqq 0 \quad \text{by (g2)}.$$

If $15 \geqq d^1 \geqq r_{n+1}$, we get

$$g(r_{n+1}) - |X^1|\,\delta f(n+1, d^1) \leqq g(15) - d^1 \sum_{j=d^1+1}^{n+1} \frac{1}{\ln^2 j} \leqq g(15) - \frac{15}{16 \ln^2 16} < 0.$$

If $16 \leqq d^1 \leqq r_{n+1}$, we have

$$g(r_{n+1}) - |X^1|\,\delta f(n+1, d^1) \leqq g(r_{n+1}) - \frac{1}{4} \frac{r_{n+1}}{\ln^2 (d^1+h)} \leqq g(r_{n+1}) - \gamma \frac{r_{n+1}}{\ln^3 (r_{n+1}+h)} = 0.$$

If $1 \leqq d^1 \leqq 15$ and $d^1 \leqq r_{n+1}$, we have

$$g(r_{n+1}) - |X^1|\,\delta f(n+1, d^1) \leqq \gamma \frac{r_{n+1}}{\ln^3 (r_{n+1}+h)} - r_{n+1} \frac{1}{(d^1+1) \ln^2 (d^2+1)}$$
$$\leqq \frac{r_{n+1}}{\ln^2 (d^1+1)} \left(\frac{\gamma}{\ln (d^1+1)} - \frac{1}{d^1+1} \right) < 0,$$

because $\ln (d^1+1)/(d^1+1) > \gamma$ for all $d^1 \in \{1, \cdots, 15\}$. Finally, if $d^1 = 0$, then

$$g(r_{n+1}) - |X^1|\,\delta f(n+1, d^1) \leqq \gamma \frac{r_{n+1}}{\ln^3 (r_{n+1}+h)} - r_{n+1} \frac{1}{2 \ln^2 2}$$
$$\leqq r_{n+1} \left(\frac{\gamma}{8} - \frac{1}{2 \ln^2 2} \right) < 0.$$

Thus, $\sum_{i=1}^{p} g(r_i) \leqq |X| f(n+1)$. We now assume

$$d^l \geqq \tfrac{2}{3}n \quad \text{for } l = 0, 1. \tag{7.5}$$

Case 2. $r_{n+1} \leqq c_1 n \ln^3 (n+h)$, where $c_1 = 0.0099$. By (7.1), $n \geqq 98 \geqq e^{c_1^{-1/3}} - h$. Thus $\ln (n+h) \geqq c_1^{-1/3}$ and

$$c_1 n \ln^3 (n+h) \geqq n. \tag{7.6}$$

This implies

$$g(r_{n+1}) \leqq g(c_1 n \ln^3 (n+h)) = \gamma \frac{c_1 n \ln^3 (n+h)}{\ln^3 (c_1 n \ln^3 (n+h) + h)} \leqq \gamma c_1 n.$$

From (7.5) follows $|D| \geqq n/3$. Thus

$$\begin{aligned}
\sum_{i=1}^{p} g(r_i) &\leqq |X^0| f(d^0) + |X^1| f(d^1) - \sum_{i \in D} \Delta g(r_i^0, r_i^1) + g(r_{n+1}) \\
&\leqq |X| f(n+1) - \sum_{i \in D} \Delta g(1, 1) + g(r_{n+1}) \quad \text{by (g5)} \\
&\leqq |X| f(n+1) - \frac{n}{3} 0.0298 \gamma + \gamma c_1 n \quad \text{by (g6)} \\
&\leqq |X| f(n+1) \quad \text{since } \frac{0.0298}{3} \geqq c_1.
\end{aligned}$$

Let us now assume

$$r_{n+1} \geqq c_1 n \ln^3 (n+h). \tag{7.7}$$

From (7.1) it follows that

$$r_{n+1} \geqq n \geqq n_0 \geqq 98 \geqq 6h. \tag{7.8}$$

For $1 \leqq i \leqq h$, define $z_i = \min \{r_i^0, r_i^1\}$ and $v_i = \max \{r_i^0, r_i^1\}$. We have

$$v_i \geqq \frac{r_i}{2} \geqq \frac{r_{n+1}}{2} \geqq 3h. \tag{7.9}$$

Case 3.

$$\sum_{i \in D} g(z_i) \geqq \frac{1}{8} \frac{r_{n+1}}{\ln^2 (r_{n+1} + h)}.$$

Then

$$\begin{aligned}
\sum_{i \in D} g(r_i^0, r_i^1) &= \sum \Delta g(z_i, v_i) \\
&\geqq \sum 1.4 \frac{g(z_i)}{\ln (z_i + h)} \quad \text{by (7.9) and (g7)} \\
&\geqq 1.4 \frac{\sum g(z_i)}{\ln (\sum g(z_i) + h)} \quad \geqq 1.4 \frac{(1/8) \ln^2 (r_{n+1}/(r_{n+1}+h))}{\ln ((1/8)(r_{n+1}/\ln^2 (r_{n+1}+h)) + h)} \\
&\geqq 0.175 \frac{r_{n+1}}{\ln^3 (r_{n+1} + h)} \quad \geqq g(r_{n+1}), \quad \text{since } 0.175 \geqq \gamma.
\end{aligned}$$

Hence in (7.4),

$$\sum_{i=1}^{p} g(r_i) \leqq (|X^0|+|X^1|)f(n+1)+g(r_{n+1})-\sum_{i\in 0}\Delta g(r_i^0, r_i^1) \leqq |X|f(n+1).$$

It remains the case that

$$\sum_{i\in D} g(z_i) \leqq \frac{1}{8}\frac{r_{n+1}}{\ln^2(r_{n+1}+h)}.$$

Define for $l=0, 1$, $B^l=\{i|r_i^l > r_i^{1-l}\}$ and $b^l=|B^l|$. Since $b^0+b^1 \leqq n$, we may assume $b^1 \leqq n/2$.

If we remove from $G(E_p, X^1)$ edges with labels not in B^1, the remaining graph consists of some connected components $G(E_p, Y^1), \cdots, G(E_p, Y^u)$ where $\bigcup_{1\leqq j\leqq u} Y^j = X^1$. Let us denote by y_i^j the number of edges in $G(E_p, Y^j)$ with label i. Each such graph contains only labels from B^1. Hence by the induction hypothesis,

$$\sum_{i=1}^{p} g(y_i^j) \leqq |Y^j| f\left(\frac{n}{2}\right)$$

and

$$\sum_{i\in B^1} g(r_i^1) \leqq \sum_{i\in B^1}\sum_{j=1}^{u} g(y_i^j) \quad \text{since } r_i^1 = \sum_{j=1}^{u} y_i^j$$

$$\leqq \sum_{j=1}^{u} |Y^j| f\left(\frac{n}{2}\right) = |X^1| f\left(\frac{n}{2}\right).$$

Thus we can conclude

$$\sum_{j=1}^{p} g(r_i) \leqq \sum_{i=1}^{n} g(r_i^0)+\sum_{i\in B^1} g(r_i^1)+\sum_{i\notin B^1} g(r_i^1)+g(r_{n+1})$$

$$\leqq |X^0|f(n+1)+|X^1|f(n+1)-|X^1|\,\delta f\left(n+1, \frac{n}{2}\right)$$

$$+\sum_{i=1}^{n} g(z_i)+g(r_{n+1}) \qquad \text{since } r_i^1 = z_i \text{ for } i\notin B^1$$

$$\leqq |X|f(n+1)-\frac{1}{4}\frac{r_{n+1}}{\ln^2(n/2+h)}+\frac{1}{8}\frac{r_{n+1}}{\ln^2(r_{n+1}+h)}$$

$$+\gamma\frac{r_{n+1}}{\ln^3(r_{n+1}+h)} \qquad \text{by (f2)}$$

$$= |X|f(n+1)-\frac{r_{n+1}}{\ln^2(r_{n+1}+h)}\left[\frac{1}{4}-\frac{1}{8}-\frac{\gamma}{\ln(r_{n+1}-h)}\right]$$

$$\leqq |X|f(n+1).$$

This completes the proof of the Proposition and Theorem 4. □

For $Y\subset\{0, 1\}^p$ and $Q\subset\{1, \cdots, p\}$, let $G^Q(E_p, Y)$ denote the subgraph of $G(E_p, Y)$ that has the same set of nodes, but only edges with labels in Q.

The previous result can then be stated as follows. For any $\varepsilon, \mu > 0$, there is a constant $A(\varepsilon, \mu)$ such that for any $Y\subset\{0, 1\}^p$ of size at most μp, one can find a set $Q\subset\{1, \cdots, p\}$ of size at least $(1-\varepsilon)p$ such that in $G^Q(E_p, Y)$ the occurrence of each label is bounded by $A(\varepsilon, \mu)$, and hence $G^Q(E_p, Y)$ has less than $A(\varepsilon, \mu)p$ edges.

This does not necessarily imply that in $G^Q(E_p, Y)$ the labelled edges are distributed in a nice uniform manner such that every node gets about the same number of labels. There might exist a neighborhood of nodes in $G^Q(E_p, y)$ where each node has a high degree (increasing with p), and some of them might have to accept many labels. It will be shown that the structure of the cube excludes such cases. Define

$$l_k(Y) = \min_{\substack{Q\subset\{1,\cdots,p\} \\ |Q|\geqq k}} \min_{\substack{\text{transformation } L \\ \text{for } G^Q(E_p,Y)}} \max_{v\in G^Q(E_p,Y)} \#L(v)$$

and

$$l_k(m) = \max_{|Y|\leqq m} l_k(Y).$$

Obviously, for $n-(\log p)/2 \leqq k \leqq n$, it holds that $l_k(p) = \theta(\log p)$.

THEOREM 5. *For any $\varepsilon, \mu > 0$ there exists a constant $R(\varepsilon, \mu)$ such that*

$$l_{(1-\varepsilon)p}(\mu p) \leqq R(\varepsilon, \mu), \quad \textit{for any } p.$$

Proof. From Corollary 6, we know that there is a constant $A = A(\varepsilon/2, \mu)$ such that $l_{(1-\varepsilon)p}(\mu p)$, $(\mu p) \leqq A$ for all p.

Let $R = R(\varepsilon, \mu) > 10A/\varepsilon g(1)$. If the theorem is false, then there exists $p \in \mathbb{N}$ and $Y \subset \{0,1\}^p$, $|Y| \leqq \mu p$ such that for any $Q \subset \{1, \cdots, p\}$ of size at least $(1-\varepsilon)p$ and any transformation L of labels to nodes for $G^Q(E_p, Y)$, we find a node v with $\#L(v) > R$.

By Corollary 6, for the given Y there exists a set $U \subset \{1, \cdots, p\}$ of size $(1-\varepsilon/2)p$ such that $G^U(E_p, Y)$ has less than Ap edges. Among all transformations of labels in $G^u(E_p, Y)$, choose L that minimizes the function

$$F(L) := \sum_{v\in Y} \max\{0, \#L(v) - R\}.$$

By assumption, for L and also any restriction $\tilde{L}$ of L to a graph $G^Q(Ep, Y)$ where Q is a subset of U of size $(1-\varepsilon)p$, $F(L)$ and $F(\tilde{L})$ are positive. L defines an orientation of the edges in $G^U(E_p, Y)$: edge $\{v, v'\}$ is changed into the directed edge (v, v') iff L assigns the label of $\{v, v'\}$ to v'. Let us call this directed graph H.

Let $Z \subset Y$ be the set of all nodes from which there is a path of length $\geqq 0$ in H to a node v with $\#L(v) > R$, and let $\bar{H}$ be the subgraph of H induced by Z. By assumption, Z is nonempty, since there is at least one node that gets more than R labels. Notice that for $z \in Z$, $\#L(z)$ equals the indegree of z in $\bar{H}$.

CLAIM 1. *Each node of Z has indegree at least R in $\bar{H}$.*

Proof. Assume $z \in Z$ has indegree less than R, and let $z = z_0, z_1, \cdots, z_l$ be a path in $\bar{H}$ from z to a node z_l with indegree bigger than R. By definition of Z, such a path must exist.

Change L into $\bar{L}$ by assigning for $0 \leqq i < l$ the label on edge $\{z_i, z_{i+1}\}$ to node z_i instead of z_{i+1}. Since in a cube all edges adjacent to a node have different labels, we have $\#\bar{L}(z_0) \leqq R$, $\#\bar{L}(z_l) = \#L(z) - 1 \geqq R$ and $\#\bar{L}(z) = \#L(z)$ for all remaining $z \in Y$. Hence

$$F(L) > F(\bar{L}),$$

which contradicts the minimality of L. □

Therefore, we now conclude that $\bar{H}$ has at least $R|Z|$ edges.

Since $\bar{H}$ is a subgraph of H, and H has the same number of edges as $G^U(E_p, Y)$, we know that $R|Z| \leqq Ap$. Hence

$$|Z| \leqq \frac{A}{R} p.$$

On the other hand, $G(E_p, Z)$ must have at least $\varepsilon p/2$ different labels; otherwise, deleting this set of labels from U would yield a subset Q of $\{1, \cdots, p\}$ of size at least $(1-\varepsilon)p$ such that L restricted to $G^Q(E_p, Y)$ does not assign more than R labels to any node. From the Proposition in the proof of Lemma 5, it follows that

$$|Z| \geqq \frac{1}{f(n)} \sum_{i=1}^{p} g(r_i),$$

where r_i = number of edges in $G(E_p, Z)$ with label i and n = number of $r_i > 0$.

Since g is monotonic and f is bounded by five, we get

$$|Z| \geqq \frac{1}{5} \cdot \frac{\varepsilon}{2} p \cdot g(1) = \frac{\varepsilon}{10} g(1) p.$$

Combining the two inequalities for $|Z|$ gives

$$\frac{\varepsilon}{10} g(1) \leqq \frac{A}{R}.$$

Hence

$$R \leqq \frac{10A}{\varepsilon g(1)}.$$

This contradicts the definition of R. □

COROLLARY 7. *If $Y \subset \{0, 1\}^p$, $\# Y = O(p)$ and Y simply 2-codes E_p, then Y 2-codes E_p $O(1)$-robustly.*

8. Problems. (i) How good are the bounds of Theorems 1 and 2?

(ii) Consider 3-coding or more general r-coding for $r \geqq 3$. Now $G(x, y)$ becomes a hypergraph, and a result analogous to Lemma 3 holds. Are there, even in the case of simple 3-coding, any nontrivial bounds on $l(G(x, y))$?

9. Appendix. Proof of Properties (g1)–(g7) and (f1)–(f2). Let $h = e^2$, let $\gamma = 0.16$ and for $x \geqq 0$ let

$$g(x) = \gamma \frac{x}{\ln^3 (x+h)}.$$

(g1) is obvious. To prove (g2) we get

$$g'(x) = \gamma \frac{\ln^3 (x+h) - x3 \ln^2 (x+h)/(x+h)}{\ln^6 (x+h)} = \gamma \frac{1}{\ln^3 (x+h)} \left[1 - \frac{3x}{(x+h) \ln (x+h)}\right].$$

Let $\varphi(x) := (x+h) \ln (x+h) - 3x$.

Then for $x \geqq 0$, $g'(x) \geqq 0 \Leftrightarrow \varphi(x) \geqq 0$. We have $\varphi'(x) = \ln (x+h) - 2$ and $\lim_{x \to \infty} \varphi(x) = \infty$, and hence $x = 0$ is the only minimum of φ for $x \geqq 0$. Since $\varphi(0) = 2e^2$, we get $\varphi(x) \geqq 0$ for all $x \geqq 0$, and $g'(x) \geqq 0$ for all $x \geqq 0$.

$$\begin{aligned} g''(x) = \gamma \Bigg(& \frac{-3}{\ln^4 (x+h)} \frac{1}{x+h} \left[1 - \frac{3x}{(x+h) \ln (x+h)}\right] \\ & - \frac{1}{\ln^3 (x+h)} \left[\frac{3(x+h) \ln (x+h) - 3x(\ln (x+h) + 1)}{(x+h)^2 \ln^2 (x+h)}\right] \Bigg) \\ = & -\gamma \frac{3}{\ln^5 (x+h)(x+h)^2} [(x+h) \ln (x+h) - 3x + h \ln (x+h) - x] \\ = & -\gamma \frac{3}{\ln^5 (x+h)(x+h)^2} [(x+2h) \ln (x+h) - 4x]. \end{aligned}$$

Let $\varphi(x) := (x+2h)\ln(x+h) - 4x$. Then for $x \geqq 0$, $g''(x) \leqq 0 \Leftrightarrow \varphi(x) \geqq 0$,

$$\varphi'(x) = \ln(x+h) + \frac{x+2h}{x+h} - 4,$$

$$\varphi''(x) = \frac{1}{x+h} + \frac{(x+h)-(x+2h)}{(x+h)^2} = \frac{x}{(x+h)^2}.$$

Since $\varphi'(0) = 0$, $\varphi''(x) \geqq 0$ for $x \geqq 0$ and $\lim_{x\to\infty} \varphi(x) = \infty$, $x = 0$ is the only minimum of $\varphi(x)$. From $\varphi(0) = 4h \geqq 0$ it follows that

(g3) $$g''(x) \leqq 0 \quad \text{for all } x \geqq 0.$$

Define $\Delta g(x, y) = g(x) + g(y) - g(x+y)$. Calculation proves (g6):

$$\Delta g(1, 1) = 2\gamma\left[\frac{1}{\ln^3(1+h)} - \frac{1}{\ln^3(2+h)}\right] \geqq 0.0298\gamma.$$

Assume $0 \leqq x$ and $0 \leqq y \leqq z$. Since for all $t \in [y, z]$, $g'(x+t) \leqq g'(t)$ by (g3), we can conclude that $g(x+z) - g(x+y) \leqq g(z) - g(y)$. This yields $g(x) + g(y) - g(x+y) \leqq g(x) + g(z) - g(x+z)$, or

(g5) $$\Delta g(x, y) \leqq \Delta g(x, z) \quad \text{for all } 0 \leqq x \text{ and } 0 \leqq y \leqq z.$$

For Δg we can show the bound for $0 \leqq x \leqq y$:

$$\Delta g(x, y) = g(x) + g(y) - g(x+y) \geqq g(x) - x \sup_{z\in[y,x+y]} g'(z) = g(x) - xg'(y).$$

This yields

$$\Delta g(x, y) \geqq g(x) - x\gamma \frac{1}{\ln^3(y+h)}\left(1 - \frac{3y}{(y+h)\ln(y+h)}\right)$$

$$= g(x)\left[1 - \frac{\ln^3(x+h)}{\ln^3(y+h)}\left(1 - \frac{3y}{(y+h)\ln(y+h)}\right)\right].$$

Since $\ln(x+h) \leqq \ln(y+h)$ and $0 \leqq 3y \leqq (y+h)\ln(y+h)$ (see proof of (g2)), $\Delta g(x, y) \geqq 0$ follows from $g(x) \geqq 0$. The case $x > y$ follows from $\Delta g(x, y) = \Delta g(y, x)$. This proves (g4). If $x + h \geqq (y+h)^{2/3}$, we get $\ln(x+h) \geqq (2/3)\ln(y+h)$ and

$$\Delta g(x, y) \geqq g(x)\frac{3y}{y+h}\frac{1}{\ln(y+h)} \geqq g(x)\frac{3y}{y+h}\frac{2/3}{\ln(x+h)} = \frac{2}{3}\frac{3y}{y+h}\frac{g(x)}{\ln(x+h)}.$$

If $y \geqq 3h$ then $\Delta g(x, y) \geqq \frac{3}{2}g(x)/\ln(x+h)$. If on the other hand $x + h \leqq (y+h)^{2/3}$, we can bound $\Delta g(x, y)$ by

$$\Delta g(x, y) \geqq g(x)\left[1 - \frac{\ln^3(x+h)}{\ln^3(y+h)}\right]$$

$$\geqq g(x)\left[1 - \left(\frac{2}{3}\right)^3\right]$$

$$\geqq 0.7g(x)$$

$$\geqq 1.4\frac{g(x)}{\ln(x+h)} \quad \text{since } \ln(x+h) \geqq 2.$$

Therefore we have shown (g7):

$$\Delta g(x, y) \geqq 1.4 \frac{g(x)}{\ln (x+h)}, \quad \text{for all } 0 \leqq x \leqq y \text{ and } y \geqq 3h.$$

For $n \in \mathbb{N}$, $n \geqq 1$, define

$$f(n) = 1 + \sum_{m=2}^{n} \frac{1}{m \ln^2 m}.$$

Then

$$\begin{aligned} f(n) &\leqq 1 + \sum_{m=2}^{\infty} \frac{1}{m \ln^2 m} = 1 + (\log_2 e)^2 \sum_{m=2}^{\infty} \frac{1}{m(\log_2 m)^2} \\ &= 1 + (\log_2 e)^2 \sum_{i=1}^{\infty} \sum_{2^i \leqq m < 2^{i+1}} \frac{1}{m(\log_2 m)^2} \\ &\leqq 1 + (\log_2 e)^2 \sum_{i=1}^{\infty} 2^i \frac{1}{2^i \cdot i^2} = 1 + (\log_2 e)^2 \frac{\pi^2}{6} \leqq 5. \end{aligned}$$

Thus (f1), $1 \leqq f(n) \leqq 5$, holds for all $n \geqq 1$. Define $\delta f(n, m) = f(n) - f(m)$ for $1 \leqq m \leqq n$. For $16 \leqq m \leqq 2/3n$,

$$\begin{aligned} \delta f(n, m) &= \sum_{j=m+1}^{n} \frac{1}{j \ln^2 j} \geqq \sum_{j=m+1}^{\lfloor 3m/2 \rfloor} \frac{1}{j \ln^2 j} \\ &\geqq \lceil m/2 \rceil \frac{1}{\lceil 3m/2 \rceil \ln^2 \lceil 3m/2 \rceil} \geqq \frac{1}{3} \frac{1}{\ln^2 \lceil 3m/2 \rceil}. \end{aligned}$$

Since $m \geqq 16$,

$$\lceil 3m/2 \rceil \leqq \left(\frac{3}{2} + \frac{1}{20}\right) m \leqq 1.6m \leqq (16+h)^{0.15} m \leqq (m+h)^{1.15} \leqq (m+h)^{\sqrt{4/3}}.$$

Hence

$$\ln^2 \left\lceil \frac{3m}{2} \right\rceil \leqq \ln^2 (m+h)^{\sqrt{4/3}} = \frac{4}{3} \ln^2 (m+h).$$

This proves

$$\text{(f2)} \qquad \delta f(n, m) \geqq \frac{1}{4} \frac{1}{\ln^2 (m+h)} \quad \text{for all } 16 \leqq m \leqq \frac{2}{3} n.$$

REFERENCES

[H] C. HYLTEN-CAVALLIUS, *On a combinatorial problem*, Colloq. Math., 6 (1958), pp. 59-65.

[P] W. J. PAUL, *On heads versus tapes*, Proc. 22nd Symposium on Foundations of Computer Science, 1981.

[ZL] A. ZVONKIN AND L. LEVIN, *The complexity of finite objects and the development of the concepts of information and randomness by means of the theory of algorithms*, Russian Math. Surveys, 6, 1970.

[ER] P. ERDÖS AND A. RENYI, *On the evolution of random graphs*, in P. Erdös, The Art of Counting, MIT Press, Cambridge, MA, 1973.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 3, July 1985

012

CONVERGENCE OF A NONLINEAR SHARPENING TRANSFORMATION FOR DIGITAL IMAGES*

CAROLYN R. JOHNSON†

Abstract. Various transformations, including Fourier and Laplacian, have been used for enhancement and restoration of digitized grey level images. A simple nonlinear transformation for sharpening digital images, which depends on local operations, was introduced by Kramer and Bruckner (Pattern Recognition, 7 (1975, pp. 53-58)). A simplified proof of the Kramer-Bruckner pointwise convergence theorem for iterates of the sharpening transformation is given.

AMS(MOS) subject classifications: Primary, 68G10, 68E99; secondary, 65K05, 54C30

1. Introduction. Graded patterns are matrices with integer entries. Such patterns are produced by high speed scanning devices such as densitometers. These digitized images arise in areas such as land use study, planetary observations, fingerprint analysis, X-ray diagnosis, and optical character recognition.

DEFINITION. Let N_1, N_2, and N_3 be positive integers. A *graded pattern* (or *digitized grey level image*) is an $N_1 \times N_2$ matrix with entries from $\{0, 1, 2, \cdots, N_3\}$.

Graded patterns often become distorted or "fuzzy" due to noise or electronic interference during data collection and transmission. Generally, graded patterns are subjected to an image enhancement procedure before being interpreted. These procedures attempt to restore the pattern to its original state. A simple nonlinear transformation for sharpening graded patterns which depends on local operations was introduced by Kramer and Bruckner (1975). Basically the transformation replaces the value of each entry of a graded pattern by the largest or smallest value in its neighborhood.

2. Definition of transformation. Rather than describe the transformation in terms of graded patterns, it is best described in terms of real valued functions F on finite sets X. Of particular interest is the case where X is the ordered pairs of an $N_1 \times N_2$ matrix, that is

$$X = \{(i, j): 1 \leqq i \leqq N_1, 1 \leqq j \leqq N_2\},$$

and F is the function on X which assigns to the ijth pair the value a_{ij}, $F((i, j)) = a_{ij}$, where $A = [a_{ij}]$ is the graded pattern.

The definition of the sharpening transformation requires the notion of a neighborhood system for X, $\mathbf{N} = \{N(x): x \in X\}$. Associate with every point $x \in X$ a unique nonempty subset $N(x)$ of X such that $x \in N(x)$. We require that the neighborhoods satisfy the symmetry condition that if $x \in N(y)$, then $y \in N(x)$ for all elements x and y of X. Associate with every real valued function F on X two other functions F^* and F_*, the local maximum and local minimum functions respectively; that is,

$$F^*(x) = \max \{F(y): y \in N(x)\} \quad \text{and} \quad F_*(x) = \min \{F(y): y \in N(x)\}$$

for all elements x of X.

If X is a finite set with a neighborhood system and F is a real valued function on X, then the sharpening transformation S is defined by

$$S(F)(x) = (SF)(x) = \begin{cases} F^*(x) & \text{if } F^*(x) - F(x) \leqq F(x) - F_*(x), \\ F_*(x) & \text{otherwise.} \end{cases}$$

* Received by the editors July 22, 1981, and in final revised form April 6, 1984.
† Bell Laboratories, Holmdel, New Jersey 07733.

Define $S^0F = F$, and for all positive integers n, $S^{n+1}F = S(S^nF)$ is given by

$$S^{n+1}F(x) = \begin{cases} (S^nF)^*(x) & \text{if } (S^nF)^* - S^nF(x) \leqq S^nF(x) - (S^nF)_*(x), \\ (S^nF)_*(x) & \text{otherwise.} \end{cases}$$

Kramer and Bruckner proved the pointwise convergence of the sequence $\{S^nF\}$. We provide a simplified proof based on the cardinality of the range.

Remark. A point $x \in X$ is called a local maximum of F if $F(x) = F^*(x)$. Dually, if $F(x) = F_*(x)$ we say that x is a local minimum of F.

It is immediate that the pointwise convergence of the sequence $\{S^nF\}$ is equivalent to the assertion that there exists a positive integer N such that for each $x \in X$, x is either a local minimum or a local maximum of S^NF.

3. Convergence.

THEOREM (Kramer-Bruckner). *If X is a finite set with a neighborhood system and F is a real valued function on X, then for every element x of X, there exists an integer N such that for all $n \geqq N$, $S^nF(x) = S^NF(x)$.*

Proof. If F is constant on X then $F^*(x) = F_*(x) = F(x)$ for all x in X, and the result is immediate. If the cardinality of $F(X)$, $|F(X)|$, equals two, then every point of X is a local maximum or local minimum of F, and again the result follows. The proof proceeds by induction on the cardinality of $F(X)$. Assume that $|F(X)| \geqq 3$, and that the result is true for all functions F and finite sets X such that $|F(X)| \leqq n$. Let $|F(X)| = n+1$.

Let $u = \max\{F(x): x \in X\}$ and $\ell = \min\{F(x): x \in X\}$. Define $\mathbf{U}(F) = \{x \in X: F(x) = u\}$ and $\mathbf{L}(F) = \{x \in X: F(x) = \ell\}$. Note that if $F(x) = u = F^*(x)$, then $S^kF(x) = (S^{k-1}F)^*(x) = u$ for all positive integers k; and similarly, if $F(x) = \ell = F_*(x)$, then $S^kF(x) = (S^{k-1}F)_*(x) = \ell$ for all positive integers k. This implies that

$$\mathbf{U}(F) \subset \mathbf{U}(SF) \subset \mathbf{U}(S^2F) \subset \cdots \subset \mathbf{U}(S^nF) \subset \cdots \subset X, \quad \text{and}$$

$$\mathbf{L}(F) \subset \mathbf{L}(SF) \subset \mathbf{L}(S^2F) \subset \cdots \subset \mathbf{L}(S^nF) \subset \cdots \subset X.$$

Because X is finite there exists an integer N such that

$$\mathbf{U}(S^NF) = \mathbf{U}(S^{N+k}F) \quad \text{and} \quad \mathbf{L}(S^NF) = \mathbf{L}(S^{N+k}F)$$

for all nonnegative integers k. Let $\mathbf{U} = \mathbf{U}(S^NF)$, $\mathbf{L} = \mathbf{L}(S^NF)$, and let x be an element of $X - (\mathbf{U} \cup \mathbf{L})$.

CLAIM. *Either $N(x) \cap \mathbf{U} = \varnothing$ or $N(x) \cap \mathbf{L} = \varnothing$.*

Suppose that $N(x) \cap \mathbf{U}$ and $N(x) \cap \mathbf{L}$ are nonempty. Then there are points y_1 and y_2 such that $y_1 \in N(x)$ and $S^NF(y_1) = u$, and $y_2 \in N(x)$ and $S^NF(y_2) = \ell$. But then

$$S^{N+1}F(x) \begin{cases} (S^NF)^*(x) = u, & \text{or} \\ (S^NF)_*(x) = \ell, \end{cases}$$

which implies that $x \in \mathbf{U} \cup \mathbf{L}$, a contradiction.

We now consider three cases.

Case 1. Suppose $N(x) \cap \mathbf{U} \neq \varnothing$. Then there is a $y \in N(x)$ such that $S^NF(y) = u$, and $S^NF(x) \neq u$, so $S^NF(x) = (S^{N-1}F)_*(x) < u$. Because $(S^{N+k}F)^*(x) = u$ for all $k = 0, 1, 2, \cdots$ and x not in $\mathbf{U}$, this means that $S^{N+k}F(x) = (S^{N+k-1}F)_*(x)$ for all nonnegative integer k. Then

$$S^NF(x) = (S^{N-1}F)_*(x) \geqq (S^NF)_*(x) \geqq (S^{N+1}F)_*(x) \geqq \cdots > \ell.$$

So there is an integer N_1 such that $S^{N_1}F(x) = S^{N_1+k}F(x)$ for all nonnegative integers k.

Case 2. If $N(x) \cap \mathbf{L}$ is nonempty then an argument symmetric to Case 1 shows that there is an integer N_2 such that $S^{N_2}F(x) = S^{N_2+k}F(x)$ for all nonnegative integers k.

Case 3. Suppose that both $N(x) \cap \mathbf{U}$ and $N(x) \cap \mathbf{L}$ are empty. Let $Y = X - (\mathbf{U} \cup \mathbf{L})$, $\tilde{F} = F|_Y$, the restriction of F to Y, and for all $y \in Y$, let $N_Y(y) = N(y) \cap Y$ be the neighborhood system for Y. Then $|\tilde{F}(Y)| < n$, so by the inductive hypothesis there is an integer N_3 such that

$$S^{N_3}\tilde{F}(x) = S^{N_3+k}\tilde{F}(x)$$

for all nonnegative integers k. But $N(x) \cap \mathbf{U} = \varnothing = N(x) \cap \mathbf{L}$ implies that $N(x) \cap Y = N(x)$ so $S\tilde{F}(x) = SF(x)$. Moreover, $S^k\tilde{F}(x) = S^kF(x)$ since $(S^k\tilde{F})_*(x) = (S^kF)_*(x)$ and $(S^k\tilde{F})^*(x) = (S^kF)^*(x)$ for all integers k. Thus, $S^{N_3}F(x) = S^{N_3}\tilde{F}(x) = S^{N_3+k}\tilde{F}(x) = S^{N_3+k}F(x)$ for all nonnegative integers k, and the result is true for all x in X.

Remark. The Kramer-Bruckner claim of pointwise convergence of $\{S^nF\}$ to the function P is then established by letting $P = S^{N_*}F$, where N_* is the maximum of the integers given by the Theorem for the individual elements of X.

4. Example. We now provide an illustration of how the sharpening transformation S can be applied to character recognition. Let X be an 8×8 matrix and let F be the function from X into $\{0, 1, 2, \cdots, 7\}$ which represents the graded pattern B with eight grey levels shown in Fig. 1. We distorted the pattern by introducing noise, $n(x)$, as follows. The noise was determined by first generating a random number r from $\{0, 1, \cdots, 9\}$ and then using the following rule:

if $r = 0$ or 9 then $n(x) = 0$

if $r = 1$ or 5 then $n(x) = 1$

if $r = 2$ or 6 then $n(x) = -1$

if $r = 3$ or 7 then $n(x) = 2$

if $r = 4$ or 8 then $n(x) = -2$.

0	7	7	7	7	0	0	0
0	7	0	0	7	0	0	0
0	7	0	0	7	0	0	0
0	7	7	7	7	7	0	0
0	7	0	0	0	7	0	0
0	7	0	0	0	7	0	0
0	7	0	0	0	7	0	0
0	7	7	7	7	7	0	0

B

1	7	6	7	7	1	0	0
2	6	2	1	7	1	0	0
0	6	2	0	5	0	1	2
1	6	6	7	7	7	1	2
2	7	1	0	0	7	0	0
0	7	0	0	0	6	2	0
0	7	0	1	2	6	0	1
0	7	7	7	5	7	2	2

T

1	7	7	7	7	0	0	0
0	7	1	0	7	0	0	0
0	6	0	0	7	0	2	2
0	7	7	7	7	7	0	2
0	7	0	0	0	7	0	0
0	7	0	0	0	7	0	0
0	7	0	0	0	7	0	2
0	7	7	7	7	7	0	2

ST

0	7	7	7	7	0	0	0
0	7	0	0	7	0	0	0
0	7	0	0	7	0	2	2
0	7	7	7	7	7	0	2
0	7	0	0	0	7	0	0
0	7	0	0	0	7	0	0
0	7	0	0	0	7	0	2
0	7	7	7	7	7	0	2

S^2T

FIG. 1

The distorted pattern is represented by T in Fig. 1, where $T: X \to \{0, 1, 2, \cdots, 7\}$ is given by

$$T(x) = \begin{cases} F(x) + n(x) & \text{if } 0 \leqq F(x) + n(x) \leqq 7, \\ F(x) & \text{otherwise.} \end{cases}$$

Crucifix neighborhoods were used; that is, the neighborhood of the ijth entry consisted of the ijth entry and those entries immediately above, below, right, and left of the entry. After two applications of the sharpening transformation S, the limit S^2T was reached and found to be very close to the original pattern, as shown in the figure.

REFERENCE

H. P. KRAMER AND J. B. BRUCKNER, *Iterations of a nonlinear transformation for enhancement of digital images*, Pattern Recognition, 7 (1975), pp. 53–58.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 3, July 1985

013

DISJUNCTIVE PROGRAMMING AND A HIERARCHY OF RELAXATIONS FOR DISCRETE OPTIMIZATION PROBLEMS*

EGON BALAS†

Abstract. We discuss a new conceptual framework for the convexification of discrete optimization problems, and a general technique for obtaining approximations to the convex hull of the feasible set. The concepts come from disjunctive programming and the key tool is a description of the convex hull of a union of polyhedra in terms of a higher dimensional polyhedron. Although this description was known for several years, only recently was it shown by Jeroslow and Lowe to yield improved representations of discrete optimization problems. We express the feasible set of a discrete optimization problem as the intersection (conjunction) of unions of polyhedra, and define an operation that takes one such expression into another, equivalent one, with fewer conjuncts. We then introduce a class of relaxations based on replacing each conjunct (union of polyhedra) by its convex hull. The strength of the relaxations increases as the number of conjuncts decreases, and the class of relaxations forms a hierarchy that spans the spectrum between the common linear programming relaxation, and the convex hull of the feasible set itself. Instances where this approach has advantages include critical path problems in disjunctive graphs, network synthesis problems, certain fixed charge network flow problems, etc. We illustrate the approach on the first of these problems, which is a model for machine sequencing.

AMS(MOS) subject classification. 90C10

1. Introduction. Most discrete optimization problems are solved by some kind of enumerative procedure. These procedures use relaxations of the feasible set, and of the subsets into which the latter is broken up, in order to derive bounds on the objective function value on these subsets. Their efficiency depends crucially on the strength of these bounds, which in turn hinges on the strength of the relaxation used. The most commonly used relaxation is the linear program obtained by removing the integrality conditions, sometimes amended with cutting planes. However, some integer programming problems have more than one formulation, and the various formulations may give rise to linear programming relaxations of varying strengths. This has been known for a long time about the simple plant location problem, for which the disaggregation of the capacity constraints involving the 0-1 variables produces a considerably stronger linear program than the aggregated one. To the disaggregation of the capacity constraints, Rardin and Choe [16] have recently added a disaggregation of the flow variables of fixed charge network flow problems, either from arc into path flows, or from single commodity into multicommodity flows, which often yields a stronger linear program than the one in the original variables.

Approaching the problem from another standpoint, that of mixed integer representability of various functions and sets, Jeroslow and Lowe [13] have recently shown how certain mixed integer formulations using a larger number of variables than the common formulation, give rise to stronger linear programming relaxations. Their approach essentially uses disjunctive programming, and our work is closely related to theirs.

* Received by the editors August 3, 1983, and in final revised form May 21, 1984. The research underlying this report was supported by the National Science Foundation under grants ECS-8205425 and ECS-8218181 and the U.S. Office of Naval Research under contract N00014-82-K-0329 NR047-607. This work was presented at the SIAM Second Conference on the Applications of Discrete Mathematics, held at Massachusetts Institute of Technology, Cambridge, Massachusetts, June 27-29, 1983.

† Graduate School of Industrial Administration, Carnegie-Mellon University, Schenley Park, Pittsburgh, Pennsylvania 15213.

Disjunctive programming is optimization over disjunctive sets. A *disjunctive set* is a set defined by inequalities connected to each other by the operations of conjunction ($\wedge$, juxtaposition, "and") or disjunction ($\vee$, "or"). Since inequalities define halfspaces, a disjunctive set can also be viewed as a collection of halfspaces joined together by the operations of intersection ($\cap$) or union ($\cup$). A disjunctive program is than a problem of the form $\min\{cx|x\in F\}$, where F is a disjunctive set.

Any integer or mixed integer program can be stated as a disjunctive program, usually in more than one way. Conversely, any bounded disjunctive program can be stated as a pure or mixed integer 0-1 program. This is not always true, though, of an unbounded disjunctive program: the set $x_j \leqq 0 \vee x_j \geqq 1$, for instance, cannot be represented by the use of integer variables unless x_j is bounded.

Besides this—not too important—difference in the domain of applicability of the two problem classes, it is often convenient to view integer programming problems as disjunctive programs. Apart from the fact that this is the most natural and straightforward way of stating many problems involving logical conditions (dichotomies, implications, etc.), the disjunctive programming approach seems to be fruitful both theoretically and practically. On the theoretical side, it provides some neat structural characterizations which offer new insights. On the practical side, it produces a variety of cutting planes, including facets of the convex hull of feasible points, which are hard to obtain by other means. In some cases, like set covering and partitioning, these cutting planes have been shown to be considerably stronger than those derived by other means, and have been successfully used in algorithms. In this paper we show that disjunctive programming also provides strong relaxations of an integer program. For background on disjunctive programming, see the surveys [4], [12], [17].

In this paper we introduce a general framework in which various linear programming relaxations can be classified, ranked, strengthened at a given computational cost, and viewed from a unifying perspective. In fact, we provide a family of relaxations of a (pure or mixed) integer 0-1 program (P) whose members form a hierarchy in terms of their strength, or tightness. The members of this hierarchy span the whole spectrum between the usual linear programming relaxation and the convex hull of the feasible set of (P). This is obtained by viewing (P) as a disjunctive program and making use of the rich variety of representations available for the latter. Our main tool is the operation of taking the convex hull of various disjunctive sets.

The paper is organized as follows. Section 2 discusses some basic properties of disjunctive sets and their equivalent forms, and describes a procedure for systematically generating these forms from each other. Section 3 deals with characterizations of the convex hull of a disjunctive set, and their relationship to mixed integer representations of such a set. Section 4 introduces the hull relaxation of a disjunctive set, which gives rise to the hierarchy of relaxations mentioned earlier. Section 5 illustrates these concepts and procedures on the disjunctive graph formulation of the machine sequencing problem.

2. Disjunctive sets and their equivalent forms. We denote a halfspace by

$$H^+ = \{x\in\mathbb{R}^n | ax \geqq a_0\},$$

where $a\in\mathbb{R}^n$, $a_0\in\mathbb{R}$. While the intersection of a finite collection of halfspaces, i.e., a set of the form

$$P = \bigcap_{i\in M} H_i^+ = \{x\in\mathbb{R}^n | a^i x \geqq a_{i0},\ i\in M\}$$

is known as a polyhedron, we call the union of a finite collection of halfspaces, i.e.,

a set of the form

$$D = \bigcup_{i \in M} H_i^+ = \left\{ x \in \mathbb{R}^n \,\middle|\, \bigvee_{i \in M} (a^i x \geqq a_{i0}) \right\},$$

an *elementary* disjunctive set.

A disjunctive set F can be expressed in many different forms, that are logically equivalent and can be obtained from each other by considering F as a logical expression whose statement forms are inequalities, and applying the rules of propositional calculus. Among these equivalent forms, the two extreme ones are the *conjunctive normal form* (CNF)

$$F = \bigcap_{i \in T} D_i,$$

where each D_i is an elementary disjunctive set, and the *disjunctive normal form* (DNF)

$$F = \bigcup_{i \in Q} P_i,$$

where each P_i is a polyhedron.

The usual statement of most discrete optimization problems is in the form of an intersection of elementary disjunctions, that is in CNF. We give a few examples.

The feasible set of a mixed integer 0-1 program given by the constraints

$$a^i x \geqq b_i,\ i \in M, \quad 0 \leqq x_j \leqq 1, j \in N, \quad x_j \leqq 0 \vee x_j \geqq 1, j \in I \subset N,$$

is in CNF, and can be written as $F = \bigcap_{i \in T} D_i$, with $T = M \cup N_1 \cup N_2 \cup I$ (where $N_1 = N_2 = N$), and D_i defined as $\{x | a^i x \geqq b_i\}$ for $i \in M$; $\{x | x_i \geqq 0\}$ for $i \in N_1$; $\{x | -x_i \geqq -1\}$ for $i \in N_2$; and $\{x | -x_i \geqq 0 \vee x_i \geqq 1\}$ for $i \in I$.

The DNF of the same set is $F = \bigcup_{S \subseteq I} P_S$, where P_S is the set of those x satisfying $a^i x \geqq b_i$, $i \in M$; $0 \leqq x_j \leqq 1$, $j \in N$; $x_j \geqq 1$, $j \in S$; and $-x_j \geqq 0$, $j \in I \setminus S$.

Similarly, the feasible set of a linear complementarity problem given by

$$a^i x + b^i y = c^i,\ i \in M, \quad x_j \geqq 0, y_j \geqq 0, j \in N, \quad x_j \leqq 0 \vee y_j \leqq 0, j \in N,$$

is in CNF, and so is the feasible set of the machine sequencing problem [1]

$$\begin{aligned} &t_j - t_i \geqq d_i, && (i,j) \in Z, \\ &t_i \geqq 0, && i \in V, \\ &t_j - t_i \geqq d_i \vee t_i - t_j \geqq d_j, && (i,j), (j,i) \in W, \end{aligned}$$

where each inequality of Z defines a precedence relation between two jobs, and each disjunctive pair $(i,j), (j,i) \in W$ states the condition that jobs i and j cannot overlap.

On the other hand, the feasible set of the set covering problem defined by the $m \times n$ matrix $A = (a_{ij})$, $a_{ij} \in \{0,1\}$, $\forall i, j$, can be stated in CNF either in the same way as shown for the general mixed integer program, or else by letting $T = M (= \{1, \cdots, m\})$ and $F = \bigcap_{i \in T} D_i$, with $D_i = \{x | \bigvee_{j \in N_i} (x_j \geqq 1)\}$, $i \in T$, where $N_i = \{j \in N | a_{ij} = 1\}$. The DNF of the same problem, on the other hand, is $F = \bigcup_{C \in \mathscr{C}} \{x | x_j \geqq 1, j \in C\}$, where $\mathscr{C}$ is the set of all covers.

Although the CNF and the DNF are the two extremes of the spectrum of equivalent forms of a disjunctive set, they share a property *not* common to all forms: each of them is an *intersection of unions of polyhedra.* We will say that a disjunctive set that has this property is in *regular form* (RF). Thus the RF is

$$F = \bigcap_{j \in T} S_j, \tag{2.1}$$

where for $j \in T$,

$$S_j = \bigcup_{i \in Q_j} P_i, \qquad P_i \text{ a polyhedron}, \quad i \in Q_j. \tag{2.2}$$

The CNF is the RF in which every S_j is elementary, i.e., every polyhedron P_i is a halfspace. The DNF, on the other hand, is the RF in which $|T| = 1$. Notice that if F is in the RF given by (2.1), (2.2), each S_j is in DNF. A disjunctive set S_j in the DNF (2.2) will be called *improper* if $S_j = P_i$ for some $i \in Q_j$, *proper* otherwise. Any disjunctive set S_j such that $|Q_j| = 1$ is improper. If S_j is improper then it is convex (and polyhedral).

Next we define an operation which, when applied to a disjunctive set in RF, results in another RF with one less conjuncts, i.e., an operation which brings the disjunctive set closer to the DNF. There are several advantages to having a disjunctive set in DNF, i.e., expressed as a union of polyhedra; beyond this, the motivation for the basic step introduced here will become clearer below when we discuss relaxations of disjunctive sets.

THEOREM 2.1. *Let F be the disjunctive set in* RF *given by* (2.1), (2.2). *Then F can be brought to* DNF *by $|T|-1$ applications of the following basic step, which preserves regularity*:

For some $k, l \in T$, $k \neq l$, bring $S_k \cap S_l$ to DNF, *by replacing it with*

$$S_{kl} = \bigcup_{\substack{i \in Q_k \\ j \in Q_l}} (P_i \cap P_j). \tag{2.3}$$

Proof. S_{kl} is the DNF of $S_k \cap S_l$. Indeed, by the distributivity of $\cup$ and $\cap$, we have

$$S_k \cap S_l = \left(\bigcup_{i \in Q_k} P_i \right) \cap \left(\bigcup_{j \in Q_l} P_j \right) = \bigcup_{\substack{j \in Q_k \\ j \in Q_l}} (P_i \cap P_j) = S_{kl}.$$

The set F given by (2.1), (2.2) is the intersection of $|T|$ unions of polyhedra. Every application of the basic step replaces the intersection of p unions of polyhedra (for some positive integer p) by the intersection of $p-1$ unions of polyhedra. Regularity is thus preserved, and after $|T|-1$ basic steps F becomes a single union of polyhedra, i.e., is in DNF. □

Remark 2.1.1. Deleting repetitions, (2.3) can be written as

$$S_{kl} = \left(\bigcup_{i \in Q_k \cap Q_l} P_i \right) \cup \left(\bigcup_{\substack{i \in Q_k \setminus Q_l \\ j \in Q_l \setminus Q_k}} (P_i \cap P_j) \right). \tag{2.3'}$$

Remark 2.1.2. If $S_k = P_{i_0}$ for some $i_0 \in Q_k$, i.e., S_k is improper, then

$$S_{kl} = \begin{cases} P_{i_0} & \text{if } i_0 \in Q_l, \\ \bigcup_{j \in Q_l} (P_{i_0} \cap P_j) & \text{otherwise.} \end{cases} \tag{2.4}$$

Every basic step reduces by one the number of conjuncts S_j in the RF to which it is applied. On the other hand, it is also of interest to know the effect of a basic step on the number of polyhedra whose unions are the conjuncts of the RF. When the basic step is applied to a pair of conjuncts S_k, S_l that are both proper disjunctive sets, namely unions of polyhedra indexed by Q_k and Q_l, respectively, then the set S_{kl} resulting from the basic step is the union of p polyhedra, where

$$p = |Q_k \setminus Q_l| \times |Q_l \setminus Q_k| + |Q_k \cap Q_l|.$$

This is to be compared with the number of polyhedra in the unions defining S_k and S_l, which is $|Q_k|+|Q_l|$. Obviously, more often than not a basic step applied to a pair of proper disjunctive sets results in *an increase in the number of polyhedra* whose union is taken. On the other hand, when one of the two disjunctive sets, say S_k, is improper, then S_{kl} is the union of *at most as many polyhedra* as S_l.

Given a disjunctive set in CNF with t conjuncts, where the ith conjunct is the union of q_i halfspaces, and given the same disjunctive set in DNF, as the union of q polyhedra, we have the bounding inequality

$$q \leqq q_1 \times \cdots \times q_t.$$

Because performing a basic step on a pair S_k, S_l such that S_k is improper, results in a set S_{kl} that is the union of no more polyhedra than is S_l, it is often useful to carry out a parallel basic step, defined as follows:

For F given by (2.1), (2.2), and $S_k = P_{i_0}$ for some $i_0 \in Q_k$ (i.e., S_k improper), replace $\bigcap_{j\in T} S_j$ by $\bigcap_{j\in T\setminus\{k\}} S_{kj}$, where each S_{kj} is defined by (2.4).

Note that if some of the basic steps of Theorem 2.1 are replaced by parallel basic steps, the total number of steps required to bring F to DNF remains the same.

Next we turn to the operation of taking the convex hull of a disjunctive set, which plays a central role in the construction of the family of relaxations that we are about to introduce.

3. The convex hull of a disjunctive set. We have two characterizations of the convex hull of a disjunctive set, each of which requires the set to be in DNF. The first one is described by the following two theorems.

THEOREM 3.1 [3], [4], [12]. *Let*

$$F = \bigcup_{i\in Q} P_i, \qquad P_i = \{x \in \mathbb{R}^n \mid A^i x \geqq a_0^i\}, \quad i \in Q, \tag{3.1}$$

where each A^i is an $m_i \times n$ matrix, each a_0^i is an m_i-vector, and Q is an arbitrary index set. Let $Q^* = \{i \in Q \mid P_i \neq \varnothing\}$, and let

$$\mathscr{C}(Q^*) = \left\{ x \in \mathbb{R}^n \;\middle|\; \begin{array}{l} \alpha x \geqq \alpha_0 \text{ for all } (\alpha, \alpha_0) \in \mathbb{R}^{n+1} \text{ such that } \alpha = u^i A^i, \\ \alpha_0 \leqq u^i a_0^i,\ i \in Q^*, \text{ for some } u^i \in \mathbb{R}^{m_i},\ u^i \geqq 0,\ i \in Q^* \end{array} \right\}.$$

Then cl conv $F = \mathscr{C}(Q^*)$.

For the next theorem we need a definition. An inequality $\alpha x \geqq \alpha_0$ is said to define (or induce) a facet of a polyhedron P of dimension n, if $\alpha x \geqq \alpha_0$ for all $x \in P$, and $\alpha x = \alpha_0$ for n affinely independent points $x \in P$.

THEOREM 3.2 [3], [4]. *Let the set F defined by* (2.1) *be full-dimensional, and let Q be finite. Then the inequality $\alpha x \geqq \alpha_0$, where $\alpha_0 \neq 0$, defines a facet of* cl conv F *if and only if $\alpha \neq 0$ is a vertex of*

$$F^{\#} = \{y \in \mathbb{R}^n \mid y = u^i A^i,\ i \in Q^* \text{ for some } u^i \geqq 0 \text{ such that } u^i a_0^i \geqq \alpha_0,\ i \in Q^*\}$$

for some fixed α_0.

Analogous results are known for the cases where F is less than full-dimensional and/or $\alpha_0 = 0$ (see [3]).

This characterization can be used to derive strong cutting planes whenever Q is small or, although Q is large, the special structure of the polyhedra P_i makes it easy to find vertices of $F^{\#}$. Such cutting planes have been derived in [2], [4], [5], [7], [15] and have been successfully used to solve, for instance, set covering [6] and set partitioning [10] problems. For related theoretical developments see also [9], [11].

The second characterization expresses the convex hull of a disjunctive set as the projection into $\mathbb{R}^n$ of a higher dimensional polyhedron. It is this second characterization that we are going to use extensively in this paper. Since this result is from an unpublished technical report, we provide a proof here. As before, we denote $Q^* = \{i \in Q | P_i \neq \emptyset\}$.

THEOREM 3.3 [3]. *Let F be given by* (3.1), *and let $\mathcal{S}(Q^*)$ be the set of all those $x \in \mathbb{R}^n$ such that there exist vectors $(y^i, y_0^i) \in \mathbb{R}^{n+1}$, $i \in Q^*$, satisfying*

$$\begin{aligned} & x - \sum_{i \in Q^*} y^i = 0, \\ & A^i y^i - a_0^i y_0^i \geqq 0, \qquad i \in Q^*. \\ & y_0^i \geqq 0, \\ & \sum_{i \in Q^*} y_0^i = 1. \end{aligned} \tag{3.2}$$

Then cl conv $F = \mathcal{S}(Q^*)$.

Proof. If P denotes the set of those $x \in \mathbb{R}^n$, $(y^i, y_0^i) \in \mathbb{R}^{n+1}$, $i \in Q^*$, satisfying the constraints of (3.2), then $\mathcal{S}(Q^*)$ is by definition the projection of P into the subspace of the x variables. Let w be a vector of variables associated with the constraints (3.2), and let $(\alpha, \alpha_0) \in \mathbb{R}^{n+1}$ and $u^i \in \mathbb{R}^{m_i}$, $i \in Q^*$, be the components of w associated with the $n+1$ equations and the $|Q^*|$ systems of inequalities, respectively, of (3.2). Define the polyhedral cone

$$W = \{w | -\alpha + u^i A^i = 0,\ \alpha_0 - u^i a_0^i \leqq 0,\ u^i \geqq 0,\ i \in Q^*\}.$$

Then the projection $\mathcal{S}(Q^*)$ of P (see [8, Thm. 2]) is the set of those $x \in \mathbb{R}^n$ that satisfy the inequality

$$(\alpha \cdot I_n + \sum_{i \in Q^*} u^i \cdot 0_{m_i}^n) x \geqq \alpha_0 \cdot 1 + \sum_{i \in Q^*} u^i \cdot 0_{m_i}^i \tag{3.3}$$

for every extreme direction vector w of W. Here I_n is the identity matrix of order n, and $0_{m_i}^n$ is the $m_i \times n$ zero matrix. Rewriting (3.3) in the simpler form $\alpha x \geqq \alpha_0$ and noticing that x satisfies (3.3) for every extreme direction vector w of W if and only if it satisfies (3.3) for every $w \in W$, we conclude that $\mathcal{S}(Q^*)$ is the set of those $x \in \mathbb{R}^n$ such that $\alpha x \geqq \alpha_0$ for every $(\alpha, \alpha_0) \in \mathbb{R}^{n+1}$ for which there exist vectors u^i, $i \in Q^*$, that together with (α, α_0) satisfy the constraints of W. But this is precisely the set $\mathcal{C}(Q^*)$, hence from Theorem 3.1, $\mathcal{S}(Q^*) = \mathcal{C}(Q^*) =$ cl conv F. □

In order to use this characterization of the convex hull, one needs to know which P_i are nonempty. This inconvenience is considerably mitigated by the fact, to be shown below, that the information in question becomes irrelevant if the systems $A^i y^i \geqq a_0^i$ satisfy a condition that is often easy to check. Let $(3.2)_Q$ be the constraint set obtained from (3.2) by substituting Q for Q^*, and let $\mathcal{S}(Q)$ be the set obtained from $\mathcal{S}(Q^*)$ by the same substitution.

For each $P_i = \{y \in \mathbb{R}^n | A^i y \geqq a_0^i\}$, define the cone $C_i = \{y \in \mathbb{R}^n | A^i y \geqq 0\}$. If $P_i \neq 0$, then C_i is the recession cone of P_i, i.e.,

$$C_i = \{y | x + \lambda y \in P_i, \forall x \in P_i, \lambda \geqq 0\}.$$

For arbitrary sets $S_i \subset \mathbb{R}^n$, $i \in M$, we denote

$$\sum_{i \in M} S_i = \left\{ x \in \mathbb{R}^n \,\middle|\, x = \sum_{i \in M} y^i \text{ for some } y^i \in S_i, i \in M \right\}.$$

THEOREM 3.4. *$\mathcal{S}(Q)=\mathcal{S}(Q^*)$ if and only if*

$$C_k \subseteq \sum_{i\in Q^*} C_i \quad \forall k \in Q\backslash Q^*. \tag{3.4}$$

Proof. For $k\in Q\backslash Q^*$, $A^k y - a_0^k y_0 \geqq 0$, $y_0 \geqq 0$ implies $y_0 = 0$. Therefore

$$\mathcal{S}(Q)=\mathcal{S}(Q^*)+\sum_{k\in Q\backslash Q^*} C_k$$

and from Theorem 3.3,

$$\mathcal{S}(Q)=\text{cl conv } F+\sum_{k\in Q\backslash Q^*} C_k.$$

But condition (3.4) holds if and only if

$$\left(\sum_{k\in Q\backslash Q^*} C_k\right) \subseteq \text{cl conv } F$$

hence $\mathcal{S}(Q)=\mathcal{S}(Q^*)$ if and only if (3.4) holds. □

COROLLARY 3.5. *If for every $i\in Q$, some subset of the set of inequalities $A^i y^i \geqq a_0^i$ defines a bounded nonempty polyhedron, then $\mathcal{S}(Q)=\mathcal{S}(Q^*)$.*

Thus the disjunctive program $\min\{cx|x\in F\}$, where F is given by (3.1), is equivalent to the linear program $\min\{cx|x\in\mathcal{S}(Q^*)\}$. Furthermore, there is a 1-1 correspondence between vertices of the polyhedra P_i, $i\in Q^*$, and basic solutions of the system (3.2). More specifically [3]:

(i) If $\bar{x}$ is a vertex of P_i for some $i\in Q^*$, then the vector with components $(\bar{y}^i, \bar{y}_0^i)=(\bar{x}, 1)$, $(\bar{y}^k, \bar{y}_0^k)=(0,0)$, $k\in Q\backslash\{i\}$, together with $\bar{x}$, is a basic solution of the system (3.2).

(ii) If $\hat{x}$ together with $(\hat{y}^k, \hat{y}_0^k)$, $k\in Q$, is a basic solution of (3.2), then $(\hat{y}^i, \hat{y}_0^i)=(\hat{x}, 1)$ for some $i\in Q^*$, $(\hat{y}^k, \hat{y}_0^k)=(0,0)$ for $k\in Q\backslash\{i\}$, and $\hat{x}$ is a vertex of P_i.

Thus all basic solutions of the system (3.2) (or $(3.2)_Q$) satisfy the condition $y_0^i\in\{0,1\}$, $i\in Q$. On the other hand, a solution of (3.2) (or $(3.2)_Q$) satisfying this condition need not be basic. It is then natural to ask the question, what do such solutions represent? The next theorem addresses this issue.

We denote by $\mathcal{S}_I(Q)$ the set of those $x\in\mathbb{R}^n$ for which there exist vectors $(y^i, y_0^i)\in\mathbb{R}^{n+1}$, $i\in Q$, satisfying the constraints of $(3.2)_Q$ and the condition $y_0^i=0$ or 1, $i\in Q$; i.e.,

$$\mathcal{S}_I(Q) := \{x\in\mathcal{S}(Q)|y_0^i\in\{0,1\},\ i\in Q\}.$$

THEOREM 3.6. *Let $F=\cup_{i\in Q} P_i$, $Q^*=\{i\in Q|P_i\neq\emptyset\}$, and $Q^{**}=\{i\in Q^*|P_i\not\subseteq P_j, \forall j\in Q^*\backslash\{i\}\}$. If F satisfies*

$$C_i = C_j \quad \forall i,j\in Q^{**} \tag{3.5}$$

and

$$C_k \subseteq C_i \quad \forall k\in Q\backslash Q^*,\ i\in Q^{**} \tag{3.6}$$

then

$$\mathcal{S}_I(Q)=F.$$

Proof. With or without (3.5) and (3.6), $\mathcal{S}_I(Q)\supseteq F$. Indeed, if $x\in P_i$ for some $i\in Q$, then x together with the vectors $(y^i, y_0^i)=(x,1)$, $(y^k, y_0^k)=(0,0)$, $k\in Q\backslash\{i\}$, satisfies the constraints defining $\mathcal{S}_I(Q)$. It remains to be shown that if (3.5) and (3.6) hold, $\mathcal{S}_I(Q)\subseteq F$.

Suppose (3.5) and (3.6) are satisfied and let $x \in \mathcal{S}_I(Q)$. Then there exists $k \in Q^{**}$, $Q' \subseteq Q^{**}$ and $Q'' \subseteq Q \setminus Q^*$, such that

$$x = y^k + \sum_{i \in Q' \cup Q''} y^i,$$

and x together with the vectors $(y^k, 1)$, $(y^i, 0)$, $i \in Q' \cup Q''$, and $(y^j, y_0^j) = (0, 0), j \in Q \setminus Q' \cup Q'' \cup \{k\}$, satisfies $(3.2)_Q$. But then $y^k \in P_k$ and $y^i \in C_k$ for $i \in Q'$ (from (3.5)) and for $i \in Q''$ (from (3.6)). Thus $x \in P_k$. □

Theorem 3.6 has the following immediate consequence, proved earlier in a different way by Jeroslow and Lowe [13].

COROLLARY 3.7. *If each P_i is nonempty and bounded, then $\mathcal{S}_I(Q) = F$.*

Thus not only is $\mathcal{S}(Q)$ the convex hull of the union of the nonempty, bounded polyhedra P_i, $i \in Q$, but $\mathcal{S}_I(Q)$ is a valid mixed-integer representation of such a union of polyhedra. As Jeroslow and Lowe [10] have recently noticed, this representation is better than the usual one, since its linear programming relaxation is $\mathcal{S}(Q)$, the convex hull of the union, which is often not true of the usual representation. By the latter we mean the representation of $F = \bigcup_{i \in Q} P_i$ as the set $\Delta_I(Q)$ of those $x \in \mathbb{R}^n$ satisfying

$$A^i x - (a_0^i - L^i)\,\delta_i \geqq L^i, \qquad i \in Q,$$

$$\sum_{i \in Q} \delta_i = 1,$$

$$\delta_i \in \{0, 1\}, \qquad i \in Q,$$

where each L^i is a lower bound (vector) on $A^i x$.

If we denote by $\Delta(Q)$ the set obtained from $\Delta_I(Q)$ by relaxing the conditions $\delta_i \in \{0, 1\}$ to $\delta_i \geqq 0$, $i \in Q$, $\Delta(Q)$ is not necessarily the convex hull of F. In other words, while $\mathcal{S}(Q) = \operatorname{conv} \mathcal{S}_I(Q)$ whenever all P_i are nonempty and bounded, for Δ we only have the relation

$$\Delta(Q) \supseteq \operatorname{conv} \Delta_I(Q)$$

which often holds as strict inclusion, as will be illustrated later.

We need one more result before introducing the family of relaxations of a disjunctive set. Namely, we want to use Theorem 3.3 to characterize the convex hull of an elementary disjunctive set.

THEOREM 3.8. *Let $D = \bigcup_{i \in Q} H_i^+ = \{x \in \mathbb{R}^n \mid \bigvee_{i \in Q} (a^i x \geqq a_{i0})\}$. Then*

$$\operatorname{cl conv} D = \begin{cases} \mathbb{R}^n & \text{if } D \text{ is proper,} \\ H_k^+ & \text{if } D \text{ is improper, with } D = H_k^+. \end{cases}$$

Proof. If $D = H_k^+$ for some $k \in Q$, cl conv $D = H_k^+$ since H_k^+ is closed and convex. Suppose now that D is proper, and let $\bar{x}$ be an arbitrary but fixed point in $\mathbb{R}^n$. From Theorem 3.3, $\bar{x} \in$ cl conv D if and only if the system

$$\sum_{i \in Q} y^i = \bar{x},$$

$$a^i y^i - a_{i0} y_0^i \geqq 0, \qquad i \in Q,$$

$$\sum_{i \in Q} y_0^i = 1,$$

$$y_0^i \geqq 0, \qquad i \in Q,$$

has a solution. From the theorem of the alternative, this is the case if and only if the

system

$$
\begin{aligned}
&-u_0^i a^i + v = 0, \qquad i \in Q,\\
&u_0^i a_{i0} - v_0 \geqq 0,\\
&v\bar{x} - v_0 < 0,\\
&u_0^i \geqq 0, \qquad i \in Q,
\end{aligned} \tag{3.7}
$$

where $u_0^i \in \mathbb{R}$, $i \in Q$, $v_0 \in \mathbb{R}$, and $v \in \mathbb{R}^n$, has no solution.

Since D is proper, there exists no $k \in Q$ such that $H_i^+ \subseteq H_k^+$, $\forall i \in Q$; hence there exist no scalars $u_0^i \geqq 0$, $i \in Q$, such that $u_0^i a_0^i = u_0^k a_0^k$, $\forall i \in Q$. Thus (3.7) has no solution for any $\bar{x}$, and hence $\bar{x} \in \text{cl conv } D$ for all $\bar{x} \in \mathbb{R}^n$, i.e., cl conv $D = \mathbb{R}^n$. □

The convex hull of a proper elementary disjunctive set is thus $\mathbb{R}^n$, i.e., replacing such a set with its convex hull is tantamount to throwing away all the constraints that define it. This of course is not true for more general disjunctive sets, as will become clear soon.

The system (3.2) which defines the convex hull of a disjunctive set in DNF is easy to write down, but is unwieldy when the set Q is large; and for a mixed integer program whose feasible set F is expressed as a disjunctive set in DNF, Q tends to be large. Thus an attempt to use Theorem 3.3 to generate the convex hull of the feasible set is in general not too promising.

On the other hand, the feasible set of most discrete optimization problems, when given as a disjunctive set in CNF, has conjuncts that are the unions of small numbers of halfspaces, often only two. Performing some basic steps one obtains a set in RF whose conjuncts are still the unions of small numbers of polyhedra. Note that if a disjunctive set is in the RF given by (2.1), (2.2), each conjunct S_j is in DNF; hence we know how to take its convex hull. Naturally, taking the convex hull of each conjunct is in general not going to deliver the convex hull of the disjunctive set, but can serve as a relaxation of the latter. This takes us to the class of relaxations announced at the beginning of this paper.

4. A hierarchy of relaxations of a disjunctive set. Given a disjunctive set in regular form

$$F = \bigcap_{j \in T} S_j$$

where each S_j is a union of polyhedra, we define the *hull-relaxation* of F, denoted h-rel F, as

$$\text{h-rel } F := \bigcap_{j \in T} \text{cl conv } S_j.$$

The hull-relaxation of F is not to be confused with the convex hull of F: its usefulness comes precisely from the fact that it involves taking the convex hull of each union of polyhedra *before* intersecting them.

Next we relate the hull-relaxation of a disjunctive set to the usual linear programming relaxation of the feasible set of a mixed integer program. Obviously, the hull-relaxation of any disjunctive set is polyhedral, since the intersection of polyhedra is a polyhedron. Suppose now that we have a disjunctive set in CNF,

$$F_0 = \bigcap_{j \in T} D_j,$$

where each D_j is the union of halfspaces. Let $T^* = \{j \in T | D_j \text{ is improper}\}$, $T^{**} = T \setminus T^*$,

and denote

$$P_0 = \bigcap_{j \in T^*} D_j,$$

with $P_0 = \mathbb{R}^n$ if $T^* = \varnothing$. By definition, P_0 is a polyhedron; and it can be viewed as the "polyhedral part" of F_0, i.e., the intersection of those elementary dijunctive sets that are halfspaces. Thus a disjunctive set in CNF can be represented as

$$F_0 = P_0 \cap \left(\bigcap_{j \in T^{**}} D_j \right)$$

where P_0 is a polyhedron and each D_j, $j \in T^{**}$, is a proper elementary disjunctive set.

LEMMA 4.1. h-rel $F_0 = P_0$.

Proof.

$$\text{h-rel } F_0 = \text{h-rel} \left(P_0 \cap \left(\bigcap_{j \in T^{**}} D_j \right) \right) = \text{cl conv } P_0 \cap \left(\bigcap_{j \in T^{**}} \text{cl conv } D_j \right)$$

by the definition of the hull-relaxation. But cl conv $P_0 = P_0$ and from Theorem 3.6, cl conv $D_j = \mathbb{R}^n$ for all $j \in T^{**}$. This yields the equality stated in the lemma. □

When the feasible set of a (pure or mixed integer) 0-1 program is stated in CNF (which is the usual way of stating it), T^* is the index set of all the conjunctive, i.e., ordinary linear constraints, and T^{**} is the index set of the disjunctions $x_j \leqq 0 \vee x_j \geqq 1$. Thus P_0 is the linear programming feasible set, and the hull-relaxation of a (pure or mixed-integer) 0-1 program stated in CNF is *identical to the usual linear programming relaxation.*

The next question we address is what happens if one applied the hull-relaxation to a disjunctive set that is *not* in CNF. Specifically, we look at the effect of a basic step in the sense of relating the hull-relaxation of the RF before the basic step to that of the RF after the basic step.

LEMMA 4.2. *For* $j = 1, 2$, *let*

$$S_j = \bigcup_{i \in Q_j} P_i,$$

where each P_i, $i \in Q_j$, $j = 1, 2$, *is a polyhedron. Then*

$$\text{cl conv } (S_1 \cap S_2) \subseteq (\text{cl conv } S_1) \cap (\text{cl conv } S_2). \tag{4.1}$$

Proof. Certainly $S_1 \cap S_2 \subseteq (\text{cl conv } S_1) \cap (\text{cl conv } S_2)$, and since cl conv $(S_1 \cap S_2)$ is the smallest closed convex set to contain $S_1 \cap S_2$, (4.1) follows. □

THEOREM 4.3. *For* $i = 0, 1, \cdots, t$, *let*

$$F_i = \bigcap_{j \in T_i} S_j$$

be a sequence of regular forms of a disjunctive set, such that

(i) F_0 *is in* CNF, *with* $P_0 = \bigcap_{j \in T_0^*} S_j$;

(ii) F_t *is in* DNF;

(iii) *for* $i = 1, \cdots, t$, F_i *is obtained from* F_{i-1} *by a basic step.*

Then

$$P_0 = \text{h-rel } F_0 \supseteq \text{h-rel } F_1 \supseteq \cdots \supseteq \text{h-rel } F_t = \text{cl conv } F_t.$$

Proof. The first equality holds by Lemma 4.1, since F_0 is in CNF. The last equality holds by the definition of a hull-relaxation, since F_t is in DNF, i.e., $|T_t| = 1$. Each

inclusion holds by Lemma 4.2, since for $k = 1, \cdots, t$, F_k is obtained from F_{k-1} by a basic step. □

For any F_i in the above sequence, we can obtain from the hull-relaxation a mixed-integer programming representation of F_i by using Theorem 3.6. However, this representation requires one 0-1 variable for every polyhedron P_h in the expression

$$F_i = \bigcap_{j \in T_i} S_j, \qquad S_j = \bigcup_{h \in Q_j} P_h, \tag{4.2}$$

which is usually much more than the number of 0-1 variables needed to represent the CNF of the same set.

The next theorem gives a mixed integer representation of F_i which uses the same number of 0-1 variables as F_0.

Let F_0 be the disjunctive set in CNF consisting of those $x \in \mathbb{R}^n$ satisfying

$$\bigvee_{s \in Q_r} (a^s x \geqq a_{s0}), \qquad r \in T_0 \tag{4.3}$$

and let F be the same set in RF obtained from F_0 by some sequence of basic steps, given as the set of $x \in \mathbb{R}^n$ satisfying

$$\bigvee_{i \in Q_j} (A^i x \geqq a_0^i), \qquad j \in T. \tag{4.4}$$

Then every $j \in T$ corresponds to some subset T_{0j} of T_0, with $T_0 = \bigcup_{j \in T} T_{0j}$, such that the disjunction in (4.4) indexed by j is the disjunctive normal form of the set of disjunctions in (4.3) indexed by T_{0j}. In other words, every system $A^i x \geqq a_0^i$, $i \in Q_j$, contains $|T_{0j}|$ inequalities $a^s x \geqq a_{s0}$, one from each disjunction $r \in T_{0j}$ of (4.3), and there are as many elements of Q_j (systems $A^i x \geqq a_0^i$, $i \in Q_j$) as there are ways of choosing them.

Let M_i be the index set of the inequalities $a^s x \geqq a_{s0}$ making up the system $A^i x \geqq a_0^i$. From the above, $|M_i| = |T_{0j}|$ for all $i \in Q_j$.

Consider now the mixed integer program with the following constraint set:

$$\begin{aligned}
&x - \sum_{i \in Q_j} y^i = 0, && j \in T,\\
&A^i y^i - a_0^i y_0^i \geqq 0, \\
&y_0^i \geqq 0, && i \in Q_j, \quad j \in T,\\
&\sum_{i \in Q_j} y_0^i = 1, && j \in T,\\
&\sum_{i \in Q_j | s \in M_i} y_0^i - \delta_s^r = 0, && s \in Q_r, \quad r \in T_0,
\end{aligned} \tag{4.5}$$

$$\begin{aligned}
&\sum_{s \in Q_r} \delta_s^r = 1, && r \in T_0,\\
&\delta_s^r \in \{0, 1\}, && s \in Q_r, \quad r \in T_0.
\end{aligned} \tag{4.6}$$

THEOREM 4.4. *Assume that F satisfies the conditions of Theorem* 3.6. *Then the constraint set* (4.4) *is equivalent to* (4.5), (4.6), *in that for every solution x to* (4.4) *there exist vectors (y^i, y_0^i), $i \in Q_j$, $j \in T$ and scalars δ_s^r, $s \in Q_r$, $r \in T_0$, that together with x satisfy* (4.5), (4.6); *and conversely, the x-component of any solution to* (4.5), (4.6) *is a solution to* (4.4).

Proof. If we write

$$F = \bigcap_{j \in T} S_j, \qquad S_j = \bigcup_{i \in Q_j} P_i, \quad P_i = \{x \in \mathbb{R}^n | A^i x \geqq a_0^i\}, \quad i \in Q_j, \quad j \in T,$$

then for every $j \in T$ the system (4.5) represents $\mathcal{S}(Q_j)$, and from Theorems 3.3–3.4, $\mathcal{S}(Q_j) = \text{cl conv } S_j$, $j \in T$. Further, from Theorem 3.6, $\mathcal{S}_I(Q_j) = S_j$, where $\mathcal{S}_I(Q_j)$ is the set defined by (4.5) and the conditions $y_0^i \in \{0, 1\}$, $i \in Q_j$, $j \in T$. Since the set of those $x \in \mathbb{R}^n$ satisfying (4.4) is

$$F = \bigcap_{j \in T} \mathcal{S}_I(Q_j),$$

it only remains to be shown that the constraints (4.6) enforce the condition $y^i \in \{0, 1\}$, $i \in Q_j$, $j \in T$, and do not exclude any solution to (4.5) that satisfies this condition.

Let $\bar{x} \in F$, and let $(\bar{y}^i, \bar{y}_0^i)$, $i \in Q_j$, $j \in T$ (together with $\bar{x}$) satisfy (4.5) with $\bar{y}_0^i \in \{0, 1\}$, $i \in Q_j$, $j \in T$. Then $\bar{y}_0^i = 1$ for exactly one $i \in Q_j$, say $i(j)$, $\bar{y}_0^i = 0$ for $i \in Q_j \backslash \{i(j)\}$, for every $j \in T$. Now for $r \in T_{0j}$, $j \in T$, let

$$\bar{\delta}_s^r = \begin{cases} 1 & \text{if } s \in M_{i(j)}, \\ 0 & \text{if } s \in Q_r \backslash M_{i(j)}. \end{cases}$$

Then clearly

$$\sum_{i \in Q_j | s \in M_i} \bar{y}_0^i = \bar{\delta}_s^r, \qquad s \in Q_r, \quad r \in T_0.$$

Further, by construction, each system $A^i x \geqq a_0^i$ contains exactly one inequality $a^s x \geqq a_{s0}$ of every disjunction $r \in T_0$ of (4.3), hence

$$\sum_{s \in Q_r} \bar{\delta}_s^r = 1, \qquad r \in T_0.$$

Thus for any solution $\bar{x}$, $(\bar{y}^i, \bar{y}_0^i)$, $i \in Q_j$, $j \in T$, to the system (4.5) amended by $y_0^i \in \{0, 1\}$, $\forall i$, there exists $\bar{\delta}$ which together with $\bar{y}$ satisfies (4.6).

Conversely, let $\hat{x}$, $(\hat{y}^i, \hat{y}_0^i)$, $i \in Q$, $j \in T$, be a solution to the system (4.5) such that $0 < \hat{y}^k < 1$ for some $k \in Q_j$, $j \in T$. From

$$\sum_{i \in Q_j} \hat{y}_0^i = 1,$$

it the follows that $\hat{y}_0^i < 1$ for all $i \in Q_j$; and since there can be no pair $i_1, i_2 \in Q_j$ such that $s \in M_{i_1} \cap M_{i_2}$ for all $s \in Q_r$, $r \in T_0$, it follows that

$$\sum_{i \in Q_j | s \in M_i} \hat{y}_0^i < 1$$

for some $s \in Q_r$, $r \in T_0$; hence some $\hat{\delta}_s^r$ must be fractional in order for (4.6) to be satisfied. □

Theorem 4.4 provides a way of representing any disjunctive set in regular form as the feasible set of a mixed-integer program with the same number of 0–1 variables as would be required to represent the same disjunctive set in CNF.

In order to make best use of the hierarchy of relaxations defined in Theorem 4.3, one would like to know which basic steps result in a strict inclusion as opposed to an equality. The next theorem addresses this question.

THEOREM 4.5. *For $j = 1, 2$, let*

$$S_j = \bigcup_{i \in Q_j} P_i,$$

where each P_i, $i \in Q$, $j = 1, 2$, is a polyhedron. Then

$$\text{cl conv } (S_1 \cap S_2) = (\text{cl conv } S_1) \cap (\text{cl conv } S_2) \tag{4.7}$$

if and only if every extreme point (extreme direction) of $(\text{cl conv } S_1) \cap (\text{cl conv } S_2)$ *is an extreme point (extreme direction) of $P_i \cap P_k$ for some $(i, k) \in Q_1 \times Q_2$.*

Proof. Let T_L and T_R denote the left-hand side and right-hand side, respectively, of (4.7). Then

$$T_L = \text{cl conv}\left(\bigcup_{\substack{i \in Q_1 \\ k \in Q_2}} (P_i \cap P_k) \right).$$

Thus $x \in T_L$ if and only if there exist scalars $\lambda_j \geqq 0$, $j \in V$ and $\mu_l \geqq 0$, $l \in W$, such that $\sum_{j \in V} \lambda_j = 1$ and

$$x = \sum_{j \in V} v_j \lambda_j + \sum_{l \in W} w_l \mu_l,$$

where V and W are the sets of extreme points and extreme direction vectors, respectively, of the union of all $P_i \cap P_k$, $(i, k) \in Q_1 \times Q_2$.

On the other hand, $x \in T_R$ if and only if there exist scalars $\lambda'_j \geqq 0$, $j \in V'$ and $\mu'_l \geqq 0$, $l \in W'$, such that $\sum_{j \in V'} \lambda'_j = 1$ and

$$x = \sum_{j \in V'} v'_j \lambda'_j + \sum_{l \in W'} w'_l \mu'_l,$$

where V' and W' are the sets of extreme points and extreme direction vectors, respectively, of T_R. If the condition of the theorem holds, i.e., if $V' \subseteq V$ and $W' \subseteq W$, then $T_R \subseteq T_L$, and since from Lemma 4.2 $T_L \subseteq T_R$, we have $T_L = T_R$ as claimed. If, on the other hand, $V' \setminus V \neq \emptyset$ or $W' \setminus W \neq \emptyset$, then there exists $x \in T_R \setminus T_L$, hence $T_L \subsetneqq T_R$. □

One immediate consequence of this theorem is

COROLLARY 4.6. *Let*

$$K = \{x \in \mathbb{R}^n | 0 \leqq x_j \leqq 1, j = 1, \cdots, n\},$$

and

$$S_j = \{x \in K | x_j \leqq 0 \vee x_j \geqq 1\}, \qquad j = 1, \cdots, n.$$

Then

$$\text{conv} \bigcap_{j=1}^{n} S_j = \bigcap_{j=1}^{n} \text{conv } S_j.$$

Thus basic steps that replace a set of disjunctive constraints of the form

$$x_j \leqq 0 \vee x_j \geqq 1, \qquad j \in T$$

by a disjunctive constraint of the form

$$\bigvee_{S \subseteq T} (x_j \leqq 0, j \in S, x_j \geqq 1, j \in T \setminus S)$$

before taking the hull-relaxation, do *not* produce a stronger relaxation: taking the convex hull before or after the execution of such basic steps produces the same result. In order to obtain a stronger hull-relaxation, the basic steps to be performed must involve some other constraints.

Next we illustrate on some examples various situations where taking the convex hull before or after a basic step does make a difference.

Example 4.1 (Fig. 4.1). Let $P_1 = \{x \in \mathbb{R}^2 | x_1 = 0, 0 \leqq x_2 \leqq 1\}$, $P_2 = \{x \in \mathbb{R}^2 | x_1 = 1, 0 \leqq x_2 \leqq 1\}$, $P_3 = \{x \in \mathbb{R}^2 | -x_1 + x_2 \geqq 0.5, x_1 \geqq 0, x_2 \leqq 1\}$, $P_4 = \{x \in \mathbb{R}^2 | x_1 - x_2 \geqq 0.5, x_1 \leqq 1, x_2 \geqq 0\}$, and let $F = S_1 \cap S_2$, with $S_1 = P_1 \cup P_2$, $S_2 = P_3 \cup P_4$. Then

$$\text{cl conv } S_1 = \{x \in \mathbb{R}^2 | 0 \leqq x_1 \leqq 1, 0 \leqq x_2 \leqq 1\},$$

$$\text{cl conv } S_2 = \{x \in \mathbb{R}^2 | 0.5 \leqq x_1 + x_2 \leqq 1.5, 0 \leqq x_1 \leqq 1, 0 \leqq x_2 \leqq 1\},$$

and

$$(\text{cl conv } S_1) \cap (\text{cl conv } S_2) = \text{cl conv } S_2.$$

On the other hand, $S_1 \cap S_2 = (P_1 \cup P_3) \cap (P_2 \cup P_4)$ (since $P_1 \cap P_4 = P_2 \cap P_3 = \varnothing$), and

$$\text{cl conv } (S_1 \cap S_2) = \{x \in \mathbb{R}^2 | 1 \leqq x_1 + 2x_2 \leqq 2,\ 0 \leqq x_1 \leqq 1\}.$$

Here (4.1) holds as strict inclusion, because the vertices (0.5, 0) and (0.5, 1) of $(\text{cl conv } S_1) \cap (\text{cl conv } S_2)$ are not vertices of either $P_1 \cap P_3$ or $P_2 \cap P_4$, although the first one is a vertex of P_4, and the second one a vertex of P_3.

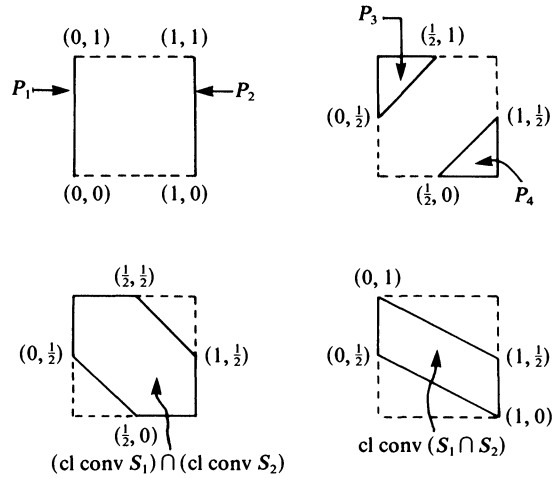


FIG. 4.1

Example 4.2 (Fig. 4.2). Let $P_1 = \{x \in \mathbb{R}^2 | x_1 = 0,\ x_2 \geqq 0\}$, $P_2 = \{x \in \mathbb{R}^2 | x_1 = 1,\ x_2 = 0\}$, $P_3 = \{x \in \mathbb{R}^2 | x_1 = 0,\ x_2 = 0\}$, $P_4 = \{x \in \mathbb{R}^2 | x_1 = 1,\ x_2 \geqq 0\}$, and let $F = S_1 \cap S_2$, with $S_1 = P_1 \cup P_2$, $S_2 = P_3 \cup P_4$. Then

$$\begin{aligned} \text{cl conv } S_1 = \text{cl conv } S_2 &= \{x \in \mathbb{R}^2 | 0 \leqq x_1 \leqq 1,\ x_2 \geqq 0\} \\ &= (\text{cl conv } S_1) \cap (\text{cl conv } S_2) \end{aligned}$$

whereas

$$\begin{aligned} \text{cl conv } (S_1 \cap S_2) &= \text{cl conv } ((P_1 \cup P_3) \cap (P_2 \cup P_4)) \\ &= \{x \in \mathbb{R}^2 | 0 \leqq x_1 \leqq 1,\ x_2 = 0\}. \end{aligned}$$

Here (4.1) holds as strict inclusion because (0, 1) is an extreme direction vector of $(\text{cl conv } S_1) \cap (\text{cl conv } S_2)$, but not of $P_1 \cap P_3$ or $P_2 \cap P_4$.

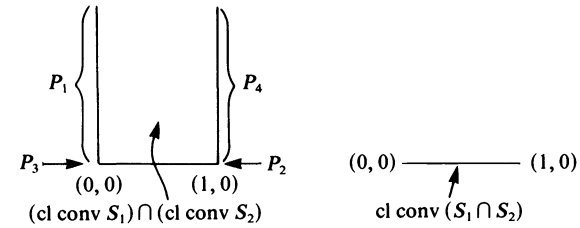


FIG. 4.2

It is an important practical problem to identify typical situations when it is useful to perform some basic step, i.e., to intersect two conjuncts of a RF before taking their convex hull. The usefulness of such a step can be measured in terms of the gain in strength of the hull-relaxation versus the price one has to pay in terms of the increase in size. Since the convex hull of an elementary disjunctive set is $\mathbb{R}^n$, i.e., taking the convex hull of such sets does not constrain the problem at all, one should intersect each elementary disjunctive set S_j in the given RF with some other conjunct S_k before taking the hull-relaxation. This can be done at no cost (in terms of new variables) if S_k is improper. Often intersecting a single improper conjunct S_k with each proper disjunctive set S_j appearing in the same RF before taking the hull-relaxation can substantially strengthen the latter without much increase in problem size. As to which improper conjunct S_k to select, a general principle that one can formulate is that the more restrictive is S_k with respect to each S_j, the better suited it is for the purpose. The next example illustrates this.

Example 4.3. Consider the 0-1 program

$$\text{(P)} \qquad \min\{z = -x_1 + 4x_2 \mid -x_1 + x_2 \geqq 0;\ x_1 + 4x_2 \geqq 2;\ x_1, x_2 \in \{0, 1\}\}$$

illustrated in Fig. 4.3.

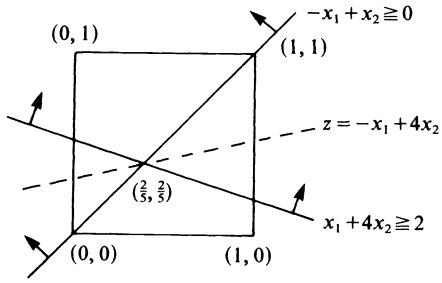


FIG. 4.3

The usual linear programming relaxation gives the optimal solution $\bar{x}_1 = \bar{x}_2 = \frac{2}{5}$, with a value of $\bar{z} = \frac{6}{5}$. This of course corresponds to taking the hull-relaxation of the CNF of the feasible set of (P), which contains as conjuncts the improper disjunctive sets corresponding to each of the inequalities of (P) (including $0 \leqq x_1 \leqq 1$, $0 \leqq x_2 \leqq 1$) and the two proper disjunctive sets $S_1 = \{x \in \mathbb{R}^2 \mid x_1 \leqq 0 \vee x_1 \geqq 1\}$, $S_2 = \{x \in \mathbb{R}^2 \mid x_2 \leqq 0 \vee x_2 \geqq 1\}$. If P_0 is the intersection of all the improper disjunctive sets, the hull-relaxation of the CNF of (P) is $F_0 = P_0 \cap \text{conv}\, S_1 \cap \text{conv}\, S_2$.

Let us write $K = \{x \in \mathbb{R}^2 \mid 0 \leqq x_1 \leqq 1,\ 0 \leqq x_2 \leqq 1\}$, and $P_0 = P_{01} \cap P_{02}$, with $P_{01} = \{x \in K \mid -x_1 + x_2 \geqq 0\}$, $P_{02} = \{x \in K \mid x_1 + 4x_2 \geqq 2\}$. Now suppose we intersect each of S_1 and S_2 with P_{01} before taking the convex hull, i.e., use the hull relaxation $F_1 = P_{02} \cap \text{conv}\,(P_{01} \cap S_1) \cap \text{conv}\,(P_{01} \cap S_2)$. We find that $\text{conv}\,(P_{01} \cap S_1) = \text{conv}\,(P_{01} \cap S_2) = \{x \in K \mid -x_1 + x_2 \geqq 0\}$, and hence $F_1 = F_0$, i.e., these particular basic steps bring no gain in the strength of the relaxation.

Suppose instead that we intersect S_1 and S_2 with P_{02} before taking the convex hull, i.e., use the hull relaxation $F_2 = P_{01} \cap \text{conv}\,(P_{02} \cap S_1) \cap \text{conv}\,(P_{02} \cap S_2)$. Then $\text{conv}\,(P_{02} \cap S_1) = \{x \in K \mid x_1 + 4x_2 \geqq 2\}$, $\text{conv}\,(P_{02} \cap S_2) = \{x \in K \mid x_2 = 1\}$, and $F_2 = \{x \in K \mid x_2 = 1\}$, which is a stronger relaxation than F_0. Using the relaxation F_2 instead of F_0, i.e., solving $\min\{z = -x_1 + 4x_2 \mid x \in F_2\}$, yields $\hat{x}_1 = \hat{x}_2 = 1$, with $\hat{z} = 3$, which happens to be the optimal solution of (P).

Note that P_{01} cuts off only one vertex of $\operatorname{conv}(S_1 \cap K) = \operatorname{conv}(S_2 \cap K) = K$, whereas P_{02} cuts off two vertices of K. □

When basic steps are used that intersect *proper* disjunctive sets before taking their convex hull, the number of variables in the hull relaxation increases. Especially attractive are those situations where the increase in problem size is mitigated by the presence of some structure that makes it possible to solve the increased linear programs efficiently. This is the case in the machine sequencing problem discussed in the next section, as well as in certain network synthesis and fixed charge network flow problems.

5. An illustration: Machine sequencing via disjunctive graphs. In this section we illustrate the concepts and methods discussed in §§ 1–4 on the example of the following well-known job shop scheduling (machine sequencing) problem: n operations are to be performed on different items using a set of machines, where the duration of operation i is d_i. The objective is to minmize total completion time, subject to (i) precedence constraints between the operations, and (ii) the condition that a machine can process only one item at a time, and operations cannot be interrupted. The problem is usually stated [1] as

$$
\begin{aligned}
&\min\ t_n \\
&\qquad t_j - t_i \geqq d_i, && (i,j) \in Z, \\
(\mathrm{P}) \qquad &\qquad t_i \geqq 0, && i \in V, \\
&\qquad t_j - t_i \geqq d_i \vee t_i - t_j \geqq d_j, && (i,j) \in W^+,
\end{aligned}
$$

where t_i is the starting time of job i (with n the dummy job "finish"), V is the set of operations, Z the set of pairs constrained by precedence relations, and W^+ the set of pairs that use the same machine and therefore cannot overlap in time. It is often useful to represent the problem by a *disjunctive graph* $G = (V, Z, W)$, with vertex set V and two kinds of directed arc sets: conjunctive (or usual) arcs, indexed by Z, and disjunctive arcs, indexed by W. The set W consists of pairs of disjunctive arcs and is of the form $W = W^+ \cup W^-$, with $(i,j) \in W^+$ if and only if $(j,i) \in W^-$. The subset of nodes corresponding to each machine, together with the disjunctive arcs joining them to each other, forms a *disjunctive clique.* A *selection* $S \subset W$ consists of exactly one member of each pair of W: i.e., there are 2^q possible selections, where $q = \frac{1}{2}|W|$: G is illustrated in Fig. 5.1, where the disjunctive arcs are shown by dotted lines. If $\mathcal{S}$ denotes the set of selections, for every $S \in \mathcal{S}$, $G_S = (V, Z \cup S)$ is an ordinary directed graph; and the problem (P(S)) obtained from (P) by replacing the set of disjunctive constraints indexed by W^+ with the set of conjunctive constraints indexed by S is the dual of a longest path (critical path) problem in G_S. Thus solving (P) amounts to finding a selection $S \in \mathcal{S}$ that minimizes the length of a critical path in G_S.

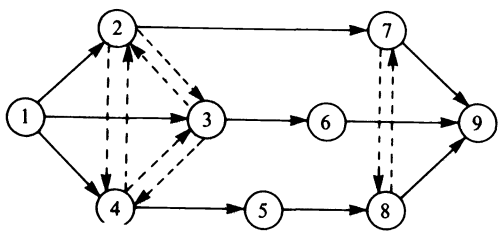


FIG. 5.1

The usual mixed integer programming formulation of (P) represents each disjunction

$$t_j - t_i \geqq d_i \vee t_i - t_j \geqq d_j \tag{5.1}$$

by the constraint set

$$\begin{aligned} & t_j - t_i - (d_i - L_{ij})y_{ij} \geqq L_{ij}, \\ & -t_j + t_i + (d_j - L_{ji})y_{ij} \geqq d_j, \\ & y_{ij} \in \{0, 1\}, \end{aligned} \tag{5.2}$$

where L_{ij} is a lower bound on $t_j - t_i$. Unless one wants to use a very crude lower bound L_{ij}, one has to derive lower and upper bounds, L_k and U_k, respectively, on each t_i, $i \in V$, and set $L_{ij} = L_j - U_i$. L_j can be taken to be the length of a longest path from node 1 (the source) to node j in the (conjunctive) graph $G_\varnothing = (V, Z)$, and U_j the difference between the length of a critical path in G_S for some arbitrary selection $S \in \mathcal{S}$, and the length of a longest path from node j to node n (the sink) in $G_\varnothing$.

The constraint set (5.2) accurately represents (5.1) (amended with the bounds $L_k \leqq t_k \leqq U_k$, $k = 1, 2$), but its linear programming relaxation $(5.2)_L$, obtained by replacing $y_{ij} \in \{0, 1\}$ by $0 \leqq y_{ij} \leqq 1$, has no constraining power, as shown by the next theorem.

THEOREM 5.1. *If the disjunction* (5.1) *is proper, then every* t_i, t_j *that satisfies*

$$L_i \leqq t_i \leqq U_i, \qquad L_j \leqq t_j \leqq U_j \tag{5.3}$$

also satisfies $(5.2)_L$.

Proof. It suffices to show that the four extreme points (L_i, L_j), (L_i, U_j), (U_i, L_j), (U_i, U_j) of the two-dimensional box defined by (5.3) satisfy $(5.2)_L$ for some y_{ij}. We first write $(5.2)_L$ in the form

$$\begin{aligned} & (L_j - U_i)(1 - y_{ij}) + d_i y_{ij} \leqq t_j - t_i \leqq -d_j(1 - y_{ij}) + (U_j - L_i)y_{ij} \\ & 0 \leqq y_{ij} \leqq 1 \end{aligned} \tag{$5.2)_L$}$$

and note that (L_i, U_j) and (L_j, U_i) satisfy (5.2) for $y_{ij} = 1$ and $y_{ij} = 0$, respectively. To show that (L_i, L_j) satisfies $(5.2)_L$ for some y_{ij}, we substitute (L_i, L_j) into $(5.2)_L$ and obtain

$$\frac{d_j - L_i + L_j}{d_j - L_i + U_j} \leqq y_{ij} \leqq \frac{U_i - L_i}{U_i - L_j + d_i}. \tag{5.4}$$

To see that (5.4) is feasible, note that the right-hand side increases with U_i; so (5.4) is feasible if it is for the smallest admissible value of U_i, which is $L_j + d_j$ (for smaller U_i (5.1) becomes improper). Substituting $L_j + d_j$ for U_i we obtain that (5.4) is feasible whenever $L_i + d_i \leqq U_j$, which is a condition for (5.1) to be proper.

An analogous argument shows that (U_i, U_j) satisfies $(5.2)_L$ for some y_{ij}. □

Consider now the mixed integer representation of (5.1) associated with the hull-relaxation of the feasible set of (P). If the latter is given in CNF, as is usually the case, applying the hull-relaxation to this form yields nothing, since the convex hull of the disjunctive set defined by (5.1) is $\mathbb{R}^2$, the space of (t_i, t_j). If we perform a sequence of basic steps of the type defined in § 3 and introduce into each disjunct of (5.1) the lower and upper bounds on t_i and t_j, this replaces every elementary disjunctive set D_{ij}

defined by a pair of constraints (5.1), by a disjunctive set

$$S_{ij} = \left\{ (t_i, t_j) \;\middle|\; \begin{pmatrix} t_j - t_i \geqq d_i \\ L_i \leqq t_i \leqq U_i \\ L_j \leqq t_j \leqq U_j \end{pmatrix} \vee \begin{pmatrix} t_i - t_j \geqq d_j \\ L_i \leqq t_i \leqq U_i \\ L_j \leqq t_j \leqq U_j \end{pmatrix} \right\}.$$

The feasible set of (P) is then of the form

$$F = P_0 \cap \left(\bigcap_{(i,j) \in W^+} S_{ij} \right) \tag{5.5}$$

where P_0 is the polyhedron defined by the inequalities (5.3) and $t_j - t_i \geqq d_{ij}$, $(i, j) \in Z$. Further, we have (since all S_{ij} are bounded, cl conv S_{ij} = conv S_{ij})

$$\text{h-rel } F = P_0 \cap \left(\bigcap_{(i,j) \in W^+} \text{conv } S_{ij} \right),$$

and from Theorem 3.3, the convex hull of S_{ij} is the set of those (t_i, t_j) satisfying the constraints

$$\begin{aligned} & t_k - t_k^1 - t_k^2 = 0, && k = i, j, \\ & t_j^1 - t_i^1 \geqq d_i y_{ij}, && \\ & -t_j^2 + t_i^2 \geqq d_j (1 - y_{ij}), && \\ & L_k y_{ij} \leqq t_k^1 \leqq U_k y_{ij}, && \\ & L_k (1 - y_{ij}) \leqq t_k^2 \leqq U_k (1 - y_{ij}), && k = i, j, \\ & 0 \leqq y_{ij} \leqq 1. && \end{aligned} \tag{5.6}$$

Also, from Corollary 3.7, the set of those (t_i, t_j) satisfying (5.6) and $y_{ij} \in \{0, 1\}$ is S_{ij}, since both disjuncts of S_{ij} are bounded polyhedra; and thus using (5.6) with $y_{ij} \in \{0, 1\}$ for all $(i, j) \in W^+$ is a valid mixed integer formulation of (P). This representation uses the same number of 0-1 variables as the usual one, but introduces two new continuous variables, t_k^1, t_k^2, for every original variable t_k, with associated bounding inequalities $L_k y_{ij} \leqq t_k^1 \leqq U_k y_{ij}$, $L_k (1 - y_{ij}) \leqq t_k^2 \leqq U_k (1 - y_{ij})$. At the price of this increase in the number of variables and constraints, one obtains as the hull-relaxation a linear program whose feasible set is considerably tighter than in the usual formulation, since each constraint set (5.6) defines the convex hull of S_{ij}. It is not hard to see that each of the two points (L_i, L_j) and (U_i, U_j) violates (5.6) unless it is contained in one of the two halfspaces defined by $t_j - t_i \geqq d_i$ and $t_i - t_j \geqq d_j$.

Let us now perform some further basic steps on the regular form (5.5) before taking the hull-relaxation. In particular, let us intersect all S_{ij} such that i and j belong to the same disjunctive clique K. If we denote $T(K) := \cap (S_{ij}: i, j \in K, i \neq j)$, and if $|K| = p$, then

$$T(K) = \{ t \in \mathbb{R}^p \mid t_i - t_j \geqq d_j \vee t_j - t_i \geqq d_i,\ i, j \in K,\ i \neq j,\ L_i \leqq t_i \leqq U_i,\ i \in K \}.$$

Taking the basic steps in question consists of putting $T(K)$ in disjunctive normal form. Let $\langle K \rangle$ denote the subgraph of G induced by K, i.e., the disjunctive clique with node set K. A selection in $\langle K \rangle$, as defined at the beginning of this section, is a set of arcs containing one member of each disjunctive pair. Thus if $\langle K \rangle$ is viewed simply as the complete digraph on K, then a selection is the same thing as a tournament in $\langle K \rangle$. If S_k denotes the kth selection in $\langle K \rangle$ and Q indexes the selections of $\langle K \rangle$, then the

DNF of $T(K)$ is $T(K) = \bigcup_{k \in Q} T_k(K)$, where

$$T_k(K) = \{t \in \mathbb{R}^p \mid t_j - t_i \geqq d_{ij}, (i,j) \in S_k, L_i \leqq t_i \leqq U_i, i \in K\}.$$

It is easy to see that if S_k contains a cycle, then $T_k(K) = \varnothing$. Let $Q^* = \{k \in Q \mid S_k$ is acyclic$\}$. Every selection is known to contain a directed Hamilton path, and for acyclic selections this path is unique. Furthermore, every acyclic selection is the transitive closure of its Hamilton path.

Let P_k denote the directed Hamilton path of the acyclic selection S_k; then S_k is the transitive closure of P_k, and the inequalities $t_j - t_i \geqq d_{ij}$, $(i,j) \in P_k$, obviously imply the remaining inequalities of $T_k(K)$, corresponding to arcs $(i,j) \in S_k \setminus P_k$. Thus a more economical expression for the DNF of T is $T(K) = \bigcup_{k \in Q^*} T_k(K)$, with

$$T_k(K) = \{t \in \mathbb{R}^p \mid t_j - t_i \geqq d_{ij}, (i,j) \in P_k, L_i \leqq t_i \leqq U_i, i \in K\}.$$

Now let M be the index set of the disjunctive cliques in G, and K_m the node set of the mth such clique. Then the RF obtained from (5.5) by performing the basic steps described above is

$$F = P_0 \cap \left(\bigcap_{m \in M} T(K_m) \right), \tag{5.7}$$

and the hull-relaxation of this form is

$$\text{h-rel } F = P_0 \cap \left(\bigcap_{m \in M} \text{conv } T(K_m) \right). \tag{5.8}$$

For $m \in M$, let Q_m^* index the acyclic selections in $\langle K_m \rangle$; and for $k \in Q_m^*$, let S_k^m and P_k^m denote the kth acyclic selection in $\langle K_m \rangle$, and its directed Hamilton path, respectively. Then introducing a continuous variable λ_k^m for every acyclic selection S_k^m and a 0–1 variable y_{ij} for every disjunctive pair of arcs $\{(i,j), (j,i)\}$, and using Theorem 4.4, we obtain the following mixed integer formulation of problem (P) based on the hull-relaxation (5.8):

$$\min t_n$$

$$(\mathcal{P}) \quad \begin{array}{llll}
t_j - t_i & & & \geqq d_{ij}, \quad (i,j) \in Z, \\
t_j - \sum_{k \in Q_m^*} t_j^k & & & = 0, \quad j \in K_m, m \in M, \\
-t_{j(1,k)}^k + t_{j(2,k)}^k & & - d_{j(1,k)} \lambda_k^m & \geqq 0, \\
\quad \ddots \qquad \ddots & & \vdots & \vdots \quad k \in Q_m^*, m \in M, \\
\quad -t_{j(p_k-1,k)}^k & + t_{j(p_k,k)}^k & - d_{j(p_k-1,k)} \lambda_k^m & \geqq 0, \\
t_{j(1,k)}^k & - t_{j(p_k,k)}^k & + (U_{j(p_k,k)} - L_{j(1,k)}) \lambda_k^m & \geqq 0, \\
 & & \sum_{k \in Q_m^*} \lambda_k^m & = 1, \quad m \in M, \\
 & & \sum_{k \mid (i,j) \in S_k} \lambda_k^m + y_{ij} & = 1, \\
 & & \sum_{k \mid (j,i) \in S_k} \lambda_k^m - y_{ij} & = 0, \quad (i,j) \in W^+
\end{array}$$

$$t_j, t_j^k \geqq 0, \forall j, k, \quad \lambda_k^m \geqq 0, \forall k, m, \quad y_{ij} \in \{0,1\}, (i,j) \in W^+.$$

THEOREM 5.2. *Problem* $(\mathcal{P})$ *is equivalent to* (P): *if* t *is a feasible solution to* (P), *there exist vectors* t^k *and scalars* λ_k^m, $k \in Q_m^*$, $m \in M$, *and a vector* y, *satisfying the constraints of* $(\mathcal{P})$; *and conversely, if* t, t^k, λ_k^m, $k \in Q_m^*$, $m \in M$, *and* y *satisfy the constraints of* $(\mathcal{P})$, *then* t *is a feasible solution to* (P).

Proof. $(\mathcal{P})$ is the representation of (P) given in Theorem 4.4, with the set F as defined in (5.9), and with the upper bounding inequalities $-t_j^k + U_j\lambda_k \geqq 0$, $j \in K_m$, replaced by the single inequality $t^k_{j(1,k)} - t^k_{j(p_k,k)} + (U_{j(p_k,k)} - L_{j(1,k)})\lambda_k \geqq 0$, for each $m \in M$. The role of the upper bounding inequalities is to force each t_j^k to 0 when $\lambda_k^m = 0$, and the inequality that replaces them in $(\mathcal{P})$ does precisely that: together with the inequalities associated with the arcs of P_k^m, it defines a directed cycle in $\langle K_m \rangle$ and thus $\lambda_k^m = 0$ forces to 0 all t_j^k, $j \in K_m$. □

The linear programming relaxation of $(\mathcal{P})$ is stronger than the linear programming relaxation of the common mixed integer formulation of (P). Preliminary computational experience on a few small problems indicates that the value of this stronger linear programming relaxation tends to be considerably higher than that of the usual linear programming relaxation. This is illustrated in Table 1 on a small sample of test problems. Problems 1, 2 and 3 are from [1], [15], and [14, p. 138] respectively. Problems 4, $\cdots$, 7 were randomly generated with $d_i \in [1, 5]$.

TABLE 1

Problem	No. of operations	No. of machines	Value of: Usual LP	Value of: Strong LP (rounded)	Value of: IP
1	7	2	18	26	31
2	14	4	8	11	13
3	14	4	20	26	35
4	17	4	8	10	12
5	17	4	7	8	10
6	17	4	7	10	12
7	17	4	8	9	12

On the other hand the linear programming relaxation of $(\mathcal{P})$, unlike that of the usual mixed integer formulation of (P), is not a longest path problem. This disadvantage has to be overcome by finding a solution method that takes advantage of the structure of $(\mathcal{P})$. While this is in general still an unsolved problem, an important aspect of it has been successfully solved. Namely, if $(\mathcal{P})$ is to be solved by projection on the space of the y-variables, i.e., by Benders' partitioning method, then in order to generate the inequalities of the Benders master problem one has to solve the dual of the linear program obtained from $(\mathcal{P})$ for various 0-1 values of y. We have recently found a way of deriving a solution to this problem from a solution to the longest path problem that corresponds to it in the usual formulation of $(\mathcal{P})$. But the discussion of this algorithm is left to another paper.

Acknowledgments. I had useful conversations with Bob Jeroslow and Charlie Blair on the subject matter of this paper.

REFERENCES

[1] E. BALAS, *Machine sequencing via disjunctive graphs: an implicit enumeration algorithm*, Oper. Res., 17 (1969), pp. 941-957.

[2] ———, *Cutting planes from logical conditions*, Nonlinear Programming 2, O. Mangasarian, R. R. Meyer and S. Robinson, eds., Academic Press, New York, 1975, pp. 279-312.

[3] ———, *Disjunctive programming: properties of the convex hull of feasible points*, MSRR No. 348, Carnegie-Mellon Univ., Pittsburgh, PA, July 1974.

[4] E. BALAS, *Disjunctive programming*, Ann. Discr. Math., 5 (1979), pp. 3-51.
[5] ———, *Cutting planes from conditional bounds: a new approach to set covering*, Math. Programming Study, 12 (1980), pp. 12-36.
[6] E. BALAS AND A. HO, *Set covering algorithms using cutting planes, heuristics, and subgradient optimization: a computational study*, Math. Programming Study, 12 (1980), pp. 37-60.
[7] E. BALAS AND R. G. JEROSLOW, *Strengthening cuts for mixed integer programs*, European J. Oper. Res., 4 (1980), pp. 224-234.
[8] E. BALAS AND W. R. PULLEYBLANK, *The perfectly matchable subgraph polytope of a bipartite graph*, Networks, 13 (1983), pp. 495-516.
[9] C. BLAIR, *Facial disjunctive programs and sequences of cutting planes*, Discr. Appl. Math., 2 (1980), pp. 173-180.
[10] R. E. CAMPELLO AND N. MACULAN, *On deep disjunctive cutting planes for set partitioning*, Mathematical Programming, R. W. Cottle, M. L. Kelmanson and B. Korte, eds., North-Holland, Amsterdam, 1984, pp. 69-78.
[11] R. G. JEROSLOW, *Cutting planes for relaxations of integer programs.* MSRR No. 347, Carnegie-Mellon Univ., Pittsburgh, PA, July 1974.
[12] ———, *Cutting plane theory: disjunctive methods*, Ann. Discr. Math., 1 (1977), pp. 293-330.
[13] R. G. JEROSLOW AND J. LOWE, *Modeling with integer variables*, Georgia Institute of Technology, Atlanta, 1982.
[14] J. LENSTRA, *Sequencing by Enumerative Methods*, Mathematisch Centrum, Amsterdam, 1977.
[15] L. NÉMETI, *Das Reihenfolgeproblem in der Fertigungsprogrammierung und Linearplanung mit logischen Bedingungen*, Mathematica, (Cluj), 6 (1964), pp. 87-99.
[16] R. RARDIN AND U. CHOE, *Tighter relaxations of fixed charge network flow problems*, Georgia Institute of Technology, Atlanta, May 1979.
[17] H. D. SHERALI AND C. M. SHETTY, *Optimization with Disjunctive Constraints*, Lecture Notes in Economics and Mathematical Systems 181, Springer-Verlag, New York, 1980.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 3, July 1985

014

ON TRANSPORTATION PROBLEMS WITH UPPER BOUNDS ON LEADING RECTANGLES*

EARL R. BARNES† AND ALAN J. HOFFMAN†

Abstract. For this class of problems, if the given bounds and cost coefficients satisfy certain conditions, an optimal solution can be found by a greedy algorithm. This work was stimulated by an application of linear programming to graph partitioning.

AMS(MOS) subject classifications. 90C08, 05C40

1. Introduction.

Consider the following transportation problem.

$$\text{maximize } \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij} \tag{1.1a}$$

subject to

$$\sum_{j=1}^{n} x_{ij} = a_i, \qquad i = 1, \cdots, m, \tag{1.1b}$$

$$\sum_{i=1}^{m} x_{ij} = b_j, \qquad j = 1, \cdots, n, \tag{1.1c}$$

$$x_{ij} \geqq 0 \quad \text{for all } i \text{ and } j. \tag{1.1d}$$

As usual, there are m origins $1, \cdots, m$, and n destinations $1, \cdots, n$. a_i is the amount of a product available at origin i and b_j is the amount required at destination j. We assume the total availabilities to be equal to the total requirements so that

$$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j.$$

c_{ij} is the negative cost of shipping a unit from origin i to destination j, and x_{ij} is the number of units shipped from i to j. An optimal shipping plan is one which maximizes (1.1a) subject to (1.1b-d).

In addition to these standard conditions, the problems that concern us here satisfy two special conditions. In the first place, the matrix $C = (c_{ij})$ satisfies a "Monge" condition

$$c_{ij} + c_{i+1,j+1} \geqq c_{i,j+1} + c_{i+1,j} \tag{1.2}$$

for $1 \leqq i < m$ and $1 \leqq j < n$.

This attribution to Monge, referring back to [1], is argued in [2]. The principal theorems of [2] show that transportation problems satisfying generalizations of (1.2) can be solved by a greedy algorithm, and that several linear programming problems can be reformulated so that they are "Mongean" transportation problems.

* Received by the editors October 4, 1983, and in revised form June 28, 1984. This work was presented at the SIAM Second Conference on the Applications of Discrete Mathematics, held at Massachusetts Institute of Technology, Cambridge, Massachusetts, June 27-29, 1983.

† IBM Thomas J. Watson Research Center, Yorktown Heights, New York 10598.

Recently we encountered a similar class of transportation problems with additional contraints of the form

$$\sum_{r=1}^{i}\sum_{s=1}^{j} x_{rs} \leqq \gamma_{ij}, \qquad i=1,\cdots,m-1, \quad j=1,\cdots,n-1 \tag{1.3}$$

on the shipments. These can be thought of as capacity constraints restricting the amount that can be shipped from the first i origins to the first j destinations. Problems involving such constraints arose in our work on graph partitioning [3]. We were given a graph G containing n nodes $N=\{1,\cdots,n\}$, with edges connecting certain pairs of nodes. The problem was to partition the nodes into a given number, say k, of disjoint subsets $S_1,\cdots,S_k$, of sizes $m_1 \geqq \cdots \geqq m_k$, respectively, in such a way that the number of edges connecting nodes in distinct subsets was minimized. Such problems arise in laying out logic networks on chips. The logic gates assigned to one chip must be weakly connected to those assigned to other chips because each chip has a limited number of pins for making connections to other chips.

A mathematical formulation of the graph partitioning problem is as follows. Let

$$y_j = \begin{pmatrix} y_{1j} \\ y_{2j} \\ \vdots \\ y_{nj} \end{pmatrix}$$

be an indicator vector for the set S_j, $j=1,\cdots,k$. Thus

$$y_{ij} = \begin{cases} 1 & \text{if } i \in S_j, \\ 0 & \text{if } i \notin S_j. \end{cases}$$

It follows that $\sum_{i=1}^{n} y_{ij} = |S_j| = m_j$, $j=1,\cdots,k$, and $\sum_{j=1}^{k} y_{ij} = 1$, $i=1,\cdots,n$.

Let a_{ij} denote the number of edges connecting nodes i and j, and let A denote the $n \times n$ adjacency matrix (a_{ij}). The number of edges of G having both endpoints in S_j is given by

$$\tfrac{1}{2}\sum_{r\in S_j}\sum_{s\in S_j} a_{rs} = \tfrac{1}{2}\sum_{r=1}^{n}\sum_{s=1}^{n} a_{rs}y_{rj}y_{sj} = \tfrac{1}{2}y_j^T A y_j.$$

It follows that the number of edges not cut by a partition $N = S_1 \cup \cdots \cup S_k$ is

$$E_{nc} = \tfrac{1}{2}\sum_{j=1}^{k} y_j^T A y_j.$$

Let E_c denote the number of edges cut by the partition. Since the total number of edges in G is fixed, minimizing E_c is equivalent to maximizing E_{nc}. Thus our graph partitioning problem is equivalent to solving

$$\begin{aligned} &\text{maximize} \sum_{j=1}^{k} y_j^T A y_j \\ &\text{subject to} \sum_{i=1}^{n} y_{ij} = m_j, \qquad j=1,\cdots,k, \\ &\qquad\qquad \sum_{j=1}^{k} y_{ij} = 1, \qquad i=1,\cdots,n, \\ &\qquad\qquad y_{ij} = 0 \text{ or } 1 \quad \text{for all } i \text{ and } j. \end{aligned} \tag{1.4}$$

There are no practical schemes for solving this problem for large values of n. However, there are heuristic schemes for obtaining approximate solutions and it is often useful to be able to determine how close to optimality a solution obtained in this way is. For this we need a tight upper bound on the maximum in (1.4). This amounts to finding a lower bound on E_c. For this purpose let

$$v_j = \frac{1}{\sqrt{m_j}} y_j, \qquad j = 1, \cdots, k.$$

These vectors form an orthonormal set. Let $\lambda_1 \geqq \cdots \geqq \lambda_n$ denote the eigenvectors of A and let $u_1, \cdots, u_n$ be a corresponding set of orthonormal eigenvectors. Then

$$A = \sum_{i=1}^{n} \lambda_i u_i u_i^T.$$

Substituting this into (1.4) gives

$$\sum_{j=1}^{k} y_j^T A y_j = \sum_{j=1}^{k} m_j v_j^T A v_j = \sum_{i=1}^{n} \sum_{j=1}^{k} \lambda_i m_j (u_i^T v_j)^2 = \sum_{i=1}^{n} \sum_{j=1}^{k} \lambda_i m_j x_{ij},$$

where $x_{ij} = (u_i^T v_j)^2$. Note that

$$\sum_{i=1}^{n} x_{ij} = \|v_j\|^2 = 1, \qquad j = 1, \cdots, k,$$

and

$$\sum_{j=1}^{k} x_{ij} \leqq \|u_i\|^2 = 1, \qquad i = 1, \cdots, n,$$

$$x_{ij} \geqq 0 \quad \text{for all } i \text{ and } j.$$

Thus an upper bound on the maximum in (1.4) can be obtained by solving the linear programming problem

$$\begin{aligned} &\text{maximize} \sum_{i=1}^{n} \sum_{j=1}^{k} \lambda_i m_j x_{ij} \\ &\text{subject to} \sum_{i=1}^{n} x_{ij} = 1, \qquad j = 1, \cdots, k, \\ &\qquad \sum_{j=1}^{k} x_{ij} \leqq 1, \qquad i = 1, \cdots, n, \\ &\qquad x_{ij} \geqq 0 \quad \text{for all } i \text{ and } j. \end{aligned} \tag{1.5}$$

The matrix $(c_{ij}) = (\lambda_i m_j)$ satisfies the Monge condition (1.2). For this matrix this condition is equivalent to the condition

$$(\lambda_i - \lambda_{i+1})(m_j - m_{j+1}) \geqq 0$$

which holds since $\lambda_1 \geqq \cdots \geqq \lambda_n$ and $m_1 \geqq \cdots \geqq m_k$.

The greedy algorithm described in [2] shows that the feasible solution of (1.5) given by the northwest corner rule is optimal. This solution is $x_{11} = x_{22} = \cdots = x_{kk} = 1$ and $x_{ij} = 0$ otherwise. Thus the value of the maximum in (1.5) is $\sum_{j=1}^{k} \lambda_j m_j$. Thus

$$E_{nc} \leqq \tfrac{1}{2} \sum_{j=1}^{k} \lambda_j m_j. \tag{1.6}$$

An important point to note here is that we are able to solve (1.5) with knowledge of only the k largest eigenvalues of A. This is a fortunate situation since in most applications k is very small compared to n and computing the k largest eigenvalues of A is a reasonable task.

We have found that in most cases it is possible to improve the bound (1.6) by discovering other linear conditions that can be imposed on the x's in (1.5). In using this approach, *it is important to discover conditions which are such that the resulting linear programming problem is solvable in greedy fashion* since we do not want to compute much more that the first k eigenvalues and eigenvectors of A. A set of such conditions can be derived as follows.

By Schwarz's inequality we have

$$x_{rs} = \frac{(u_r^T y_s)^2}{m_r} \leqq \sum_{t=1}^{n} u_{tr}^2 y_{ts}$$

where u_{tr} denotes the tth component of u_r. It follows that

$$\sum_{r=1}^{i}\sum_{s=1}^{j} x_{rs} \leqq \sum_{s=1}^{j}\sum_{t=1}^{n}\left(\sum_{r=1}^{i} u_{tr}^2\right) y_{ts} = \sum_{t=1}^{n}\left(\sum_{r=1}^{i} u_{tr}^2\right)\left(\sum_{s=1}^{j} y_{ts}\right). \tag{1.7}$$

Let $t_1, t_2, \cdots, t_n$ be a permutation of the numbers $1, 2, \cdots, n$ such that

$$\sum_{r=1}^{i} u_{t_1 r}^2 \geqq \sum_{r=1}^{i} u_{t_2 r}^2 \geqq \cdots \geqq \sum_{r=1}^{i} u_{t_n r}^2.$$

It then follows from (1.7), together with the conditions

$$\sum_{s=1}^{j} y_{ts} = 0 \text{ or } 1, \qquad t = 1, \cdots, n,$$

$$\sum_{t=1}^{n}\left(\sum_{s=1}^{j} y_{ts}\right) = \sum_{s=1}^{j} m_s$$

that

$$\sum_{r=1}^{i}\sum_{s=1}^{j} x_{rs} \leqq \gamma_{ij} = \sum_{l=1}^{m_1+\cdots+m_j}\sum_{r=1}^{i} u_{l r}^2. \tag{1.8}$$

These conditions, together with (1.5), give a problem of the form (1.1), (1.3). The purpose of this paper is to show that under appropriate conditions on the $(m-1)\times(n-1)$ matrix (γ_{ij}), we can solve (1.1), subject to (1.3), by a greedy algorithm similar to the one described in [2].

In § 2 we state conditions on $\{\gamma_{ij}\}$ necessary and sufficient for feasibility. In § 3, we prescribe our algorithm. It produces an $X = (x_{ij})$ which satisfies all conditions for feasibility except possibly the nonnegativity requirement. If X is feasible, then (Theorem 3.1) it is optimal. So we furnish sufficient conditions on $\{\gamma_{ij}\}$ for our algorithm to produce a feasible X. The matrix (γ_{ij}) given by (1.8) is easily shown to satisfy these conditions. Some remarks on extensions are contained in § 4.

2. Feasibility conditions.

LEMMA 2.1. *A necessary and sufficient condition for there to exist a solution of the system* (1.1b-d), (1.3) *is that*

$$\gamma_{ij} \geqq \max\left\{0, \sum_{r=1}^{i} a_r - \sum_{s=j+1}^{n} b_s\right\} \tag{2.1}$$

for $i = 1, \cdots, m-1$ *and* $j = 1, \cdots, n-1$.

Proof. In [4] M. Fréchet shows that (1.1b-d) always has a solution (x_{rs}^0) satisfying

$$\sum_{r=1}^{i}\sum_{s=1}^{j} x_{rs}^0 = \max\left\{0, \sum_{r=1}^{i} a_r - \sum_{s=j+1}^{n} b_s\right\}$$

for $i=1,\cdots,m-1, j=1,\cdots,n-1$, and that any other solution (x_{rs}) satisfies

$$\sum_{r=1}^{i}\sum_{s=1}^{j} x_{rs} \geqq \sum_{r=1}^{i}\sum_{s=1}^{j} x_{rs}^0.$$

This is clearly equivalent to our claim. But we wish to give an independent proof of this result.

First consider necessity of (2.1). If (x_{rs}) is any solution of (1.1b-d), (1.3), we have

$$\begin{aligned}\gamma_{ij} \geqq \sum_{r=1}^{i}\sum_{s=1}^{j} x_{rs} &= \sum_{r=1}^{m}\sum_{s=1}^{n} x_{rs} - \sum_{r=i+1}^{m}\sum_{s=1}^{n} x_{rs} - \sum_{r=1}^{m}\sum_{s=j+1}^{n} x_{rs} + \sum_{r=i+1}^{m}\sum_{s=j+1}^{n} x_{rs}\\ &= \sum_{r=1}^{m} a_r - \sum_{r=i+1}^{m} a_r - \sum_{s=j+1}^{n} b_s + \sum_{r=i+1}^{m}\sum_{s=j+1}^{n} x_{rs}\\ &\geqq \sum_{r=1}^{i} a_r - \sum_{s=j+1}^{n} b_s.\end{aligned}$$

This last inequality follows since each $x_{rs} \geqq 0$. This implies also that $\gamma_{ij} \geqq 0$, which proves (2.1).

Suppose now that (2.1) holds. Let (x_{rs}^0) be the solution of (1.1b-d) determined by the northeast corner rule. That is, let

$$x_{1n}^0 = \min\{a_1, b_n\}. \tag{2.2a}$$

If x_{rs}^0 has been determined for $r \leqq i$ and $s \geqq j$, $(r,s) \neq (i,j)$, Let

$$x_{ij}^0 = \min\left\{a_i - \sum_{s=j+1}^{n} x_{is}^0, b_j - \sum_{r=1}^{i-1} x_{rj}^0\right\}. \tag{2.2b}$$

This is the so-called minimal solution computed by Fréchet in [4]. It is easy to prove by induction that

$$\sum_{r=1}^{i}\sum_{s=j}^{n} x_{rs}^0 = \min\left\{\sum_{r=1}^{i} a_r, \sum_{s=j}^{n} b_s\right\}$$

for all i and j. Since $\sum_{r=1}^{i}\sum_{s=1}^{n} x_{rs}^0 = \sum_{r=1}^{i} a_r$ we have, for $j \geqq 2$,

$$\sum_{r=1}^{i}\sum_{s=1}^{j-1} x_{rs}^0 = \sum_{r=1}^{i} a_r - \min\left\{\sum_{r=1}^{i} a_r, \sum_{s=j}^{n} b_s\right\} = \max\left\{0, \sum_{r=1}^{i} a_r - \sum_{s=j}^{n} b_s\right\} \leqq \gamma_{i,j-1}.$$

This shows that (x_{ij}^0) is a feasible solution of (1.1b-d), (1.3).

3. A greedy algorithm. The algorithm we propose for solving (1.1), (1.3) is the following.

Let

$$x_{11} = \min\{a_1, b_1, \gamma_{11}\}. \tag{3.1a}$$

If x_{rs} has been defined for $r \leqq i < m$ and $s \leqq j < n$, $(r,s) \neq (i,j)$, define

$$x_{ij} = \min\left\{a_i - \sum_{s=1}^{j-1} x_{is}, b_j - \sum_{r=1}^{i-1} x_{rj}, \gamma_{ij} - \sum_{\substack{r\leqq i\, s\leqq j\\ (r,s)\neq(i,j)}}\sum x_{rs}\right\}. \tag{3.1b}$$

If $i=m$ or $j=n$, x_{ij} is defined by (3.1b) with the third term in the bracket missing. *Note that we do not require here that x_{ij} be nonnegative.*

LEMMA 3.1. *Let (y_{rs}) be any solution of* (1.1b-d), (1.3) *and let (x_{rs}) be given by* (3.1). *Then for $i=1,\cdots,m$ and $j=1,\cdots,n$,*

$$\sum_{r=1}^{i}\sum_{s=1}^{j} x_{rs} \geqq \sum_{r=1}^{i}\sum_{s=1}^{j} y_{rs}. \tag{3.2}$$

Proof. Since the y's are nonnegative and $x_{11}=\min\{a_1, b_1, \gamma_{11}\}$ the lemma clearly holds for $(i,j)=(1,1)$. Assume we know that $\sum_{s=1}^{j-1} x_{1s} \geqq \sum_{s=1}^{j-1} y_{1s}$ for some $1<j<n$. Then

$$\begin{aligned}\sum_{s=1}^{j} x_{1s} &= \sum_{s=1}^{j-1} x_{1s} + \min\left\{a_1 - \sum_{s=1}^{j-1} x_{1s},\ b_j,\ \gamma_{1j} - \sum_{s=1}^{j-1} x_{1s}\right\} = \min\left\{a_1, b_j + \sum_{s=1}^{j-1} x_{1s},\ \gamma_{1j}\right\} \\ &\geqq \min\left\{a_1, b_j + \sum_{s=1}^{j-1} y_{1s},\ \gamma_{1j}\right\} = \sum_{s=1}^{j-1} y_{1s} + \min\left\{a_1 - \sum_{s=1}^{j-1} y_{1s},\ b_j,\ \gamma_{1j} - \sum_{s=1}^{j-1} y_{1s}\right\} \\ &\geqq \sum_{s=1}^{j} y_{1s}.\end{aligned}$$

Thus, by induction, (3.2) holds for $i=1$ and all $j<n$. similarly, (3.2) holds for $j=1$ and all $i<m$.

Assume now that we have shown that

$$\sum_{r=1}^{i-1}\sum_{s=1}^{j} x_{rs} \geqq \sum_{r=1}^{i-1}\sum_{s=1}^{j} y_{rs}$$

and

$$\sum_{r=1}^{i}\sum_{s=1}^{j-1} x_{rs} \geqq \sum_{r=1}^{i}\sum_{s=1}^{j-1} y_{rs}$$

for some $i<m$ and $j<n$. Then, since the y's are nonnegative we must have

$$y_{ij} \leqq \min\left\{a_i - \sum_{s=1}^{j-1} y_{is},\ b_j - \sum_{r=1}^{i-1} y_{rj},\ \gamma_{ij} - \sum_{\substack{r\leqq i\, s\leqq j \\ (r,s)\neq(i,j)}}\sum y_{rs}\right\},$$

which implies that

$$\begin{aligned}\sum_{r=1}^{i}\sum_{s=1}^{j} y_{rs} &\leqq \min\left\{a_i + \sum_{r=1}^{i-1}\sum_{s=1}^{j} y_{rs},\ b_j + \sum_{r=1}^{i}\sum_{s=1}^{j-1} y_{rs},\ \gamma_{ij}\right\} \\ &\leqq \min\left\{a_i + \sum_{r=1}^{i-1}\sum_{s=1}^{j} x_{rs},\ b_j + \sum_{r=1}^{i}\sum_{s=1}^{j-1} x_{rs},\ \gamma_{ij}\right\} \\ &= \sum_{r=1}^{i}\sum_{s=1}^{j} x_{rs}.\end{aligned}$$

In case $j=n$ or $i=m$, the foregoing arguments are valid with the third term in brackets deleted. The conclusion of the lemma now follows by induction.

COROLLARY. *If the matrix (γ_{ij}) satisfies* (2.1) *the matrix (x_{ij}) given by* (3.1) *satisfies* (1.1b) *and* (1.1c).

Proof. Since (2.1) is satisfied (1.1), (1.3) has a feasible solution (y_{rs}). By Lemma 3.1, we have for $i=1,\cdots,m$,

$$\sum_{r=1}^{i}\sum_{s=1}^{n} x_{rs} \geqq \sum_{r=1}^{i}\sum_{s=1}^{n} y_{rs} = \sum_{r=1}^{i} a_r.$$

On the other hand, (3.1) shows that

$$\sum_{s=1}^{n} x_{rs} \leqq a_r$$

for each r so that

$$\sum_{r=1}^{i} \sum_{s=1}^{n} x_{rs} \leqq \sum_{r=1}^{i} a_r.$$

It follows that (1.1b) holds. Similarly it can be shown that (1.1c) holds.

Example 3.1. Let $m = n = 3$ and take $a_1 = a_2 = a_3 = 3$ and $b_1 = 3$, $b_2 = 2$, $b_3 = 4$. Let

$$(\gamma_{ij}) = \begin{pmatrix} 1 & 3 \\ 2 & 3 \end{pmatrix}.$$

This matrix satisfies (2.1) and so the system (1.1b-d), (1.3) has a feasible solution. In fact the algorithm (2.2) gives the feasible solution

$$(x_{ij}^0) = \begin{pmatrix} 0 & 0 & 3 \\ 0 & 2 & 1 \\ 3 & 0 & 0 \end{pmatrix}.$$

However, the algorithm (3.1) gives the matrix

$$(x_{ij}) = \begin{pmatrix} 1 & 2 & 0 \\ 1 & -1 & 3 \\ 1 & 1 & 1 \end{pmatrix},$$

which is not a feasible solution of (1.1b-d), (1.3).

We must impose further restrictions on the matrix (γ_{ij}) to ensure that (3.1) gives a feasible solution (1.1b-d), (1.3); in particular, we are interested in (1.1d).

LEMMA 3.2. *If the* $(m-1)\times(n-1)$ *matrix* (γ_{ij}) *satisfies* (2.1) *and has the property that whenever* $r \geqq i$ *and* $s \geqq j$, *the inequalities*

$$\gamma_{ij} + \gamma_{rs} \geqq \gamma_{is} + \gamma_{rj}, \tag{3.3a}$$

$$\gamma_{ij} \leqq \gamma_{is}, \tag{3.3b}$$

$$\gamma_{ij} \leqq \gamma_{rj} \tag{3.3c}$$

hold, then (3.1) *gives a feasible solution of* (1.1b-d), (1.3).

Proof. Because of the corollary to Lemma 3.1 it suffices to show that the matrix (x_{ij}) defined by (3.1) is nonnegative. And for this it suffices to show that the third term in the bracket defining x_{ij} is always nonnegative.

Since $\gamma_{11} \geqq 0$ we have $x_{11} \geqq 0$. If $x_{11}, \cdots, x_{1,j-1}$ have been shown to be nonnegative for some $1 < j < n$, we have $\sum_{s=1}^{j-1} x_{1s} \leqq \gamma_{1,j-1}$ which, together with (3.3b), implies

$$\gamma_{1j} - \sum_{s=1}^{j-1} x_{1s} \geqq \gamma_{1j} - \gamma_{1,j-1} \geqq 0.$$

It follows that $x_{1j} \geqq 0$ for $1 \leqq j < n$. Similarly $x_{i1} \geqq 0$ for $1 \leqq i < m$. For $j = n$ or $i = m$, $x_{ij} \geqq 0$ since there is no third term in the bracket defining x_{ij}.

Assume that x_{rs} has been shown to be $\geqq 0$ for $1 \leqq r \leqq i$, $1 \leqq s \leqq j$, $(r, s) \neq (i, j)$, for some $m > i > 1$ and $n > j > 1$. If $x_{is} = 0$ for $s = 1, \cdots, j-1$, then

$$\sum_{\substack{r=1 \\ (r,s)\neq(i,j)}}^{i} \sum_{s=1}^{j} x_{rs} = \sum_{r=1}^{i-1} \sum_{s=1}^{j} x_{rs} \leqq \gamma_{i-1,j}$$

which implies that

$$\gamma_{ij} - \sum_{\substack{r=1 \\ (r,s)\neq(i,j)}}^{i} \sum_{s=1}^{j} x_{rs} \geqq \gamma_{ij} - \gamma_{i-1,j} \geqq 0.$$

This in turn implies that $x_{ij} \geqq 0$.

Similarly, if $x_{rj} = 0$ for $r = 1, \cdots, i-1$,

$$\gamma_{ij} - \sum_{\substack{r=1 \\ (r,s)\neq(i,j)}}^{i} \sum_{s=1}^{j} x_{rs} \geqq \gamma_{ij} - \gamma_{i,j-1} \geqq 0$$

which implies that $x_{ij} \geqq 0$.

Assume now that there is an origin destination pair (p, q) with $p < i$ and $q < j$ such that $x_{pj} > 0$ and $x_{iq} > 0$. This means that the definition of x_{pq} did not exhaust the supply at p or the demand at q. It must therefore be the case that the capacity constraint

$$\sum_{r=1}^{p} \sum_{s=1}^{q} x_{rs} = \gamma_{pq} \tag{3.4}$$

is satisfied. Fix q, and redefine p as the largest index $< i$ for which (3.4) is satisfied. Then clearly $x_{rs} = 0$ for all $p < r < i$ and $s > q$. This follows since in defining x_{rq} by the greedy algorithm we do not exhaust the demand at q (because $x_{iq} > 0$) or satisfy a capacity constraint (by the definition of p). We therefore exhaust the supply at r, $p < r < i$. It follows that

$$\gamma_{ij} - \sum_{\substack{r=1 \\ (r,s)\neq(i,j)}}^{i} \sum_{s=1}^{j} x_{rs} = \gamma_{ij} - \left\{ \sum_{r=1}^{p} \sum_{s=1}^{j} x_{rs} + \sum_{r=1}^{i} \sum_{s=1}^{q} x_{rs} - \sum_{r=1}^{p} \sum_{s=1}^{q} x_{rs} \right\}$$

$$\geqq \gamma_{ij} - \gamma_{pj} - \gamma_{iq} + \gamma_{pq} \geqq 0$$

by (3.3a). Thus $x_{ij} \geqq 0$. By induction this is true for all i and j if $i < m$ and $j < n$. The cases where $i = m$ or $j = n$ are covered by again observing that there is no third term in the bracket.

Remark. We emphasize that Lemma 3.2 gives a sufficient condition for (3.1) to give a feasible solution of (1.1b-d) and (1.3). It is not necessary for (3.3) to hold in order to obtain a solution by our method.

Example 3.2. Let $m = n = 3$ and take $a_1 = 3$, $a_2 = 2$, $a_3 = 4$ and $b_1 = 5$, $b_2 = 3$, $b_3 = 1$. Let

$$(\gamma_{ij}) = \begin{pmatrix} 1 & 2 \\ 5 & 4 \end{pmatrix}.$$

This matrix satisfies (2.1) but not (3.3). Yet the algorithm (3.1) gives the feasible solution

$$(x_{ij}) = \begin{pmatrix} 1 & 1 & 1 \\ 2 & 0 & 0 \\ 2 & 2 & 0 \end{pmatrix}$$

of (1.1b-d), (1.3).

This brings us to the main result of our paper. It generalizes the greedy algorithm given in [2] for a cost matrix C satisfying the special Monge condition (1.2).

THEOREM 3.1. *If the matrix* (x_{rs}) *defined by* (3.1) *is a feasible solution for problem* (1.1), (1.3), *it is an optimal solution.*

Proof. It is clear from (1.1b) that the solutions of (1.1), (1.3) do not change if we add a constant to any row of C. Moreover, the property (1.2) is not destroyed by such a change in C. Thus without loss of generality, we may assume that $c_{in}=0$ for $i=1,\cdots,m$. If necessary we replace c_{ij} by $c_{ij}-c_{in}$, $j=1,\cdots,n$, in order to accomplish this. Similarly, we may assume that $c_{mj}=0, j=1,\cdots,n$.

For each $i=1,\cdots,m-1$ and each $j=1,\cdots,n-1$ let $F^{ij}=(F^{ij}_{rs})$ be the $m\times n$ matrix defined by

$$F^{ij}_{rs}=\begin{cases}1 & \text{if } r\leqq i \text{ and } s\leqq j,\\ 0 & \text{otherwise.}\end{cases}$$

Lemma 3.1 shows that

$$\max \sum_{r=1}^{m}\sum_{s=1}^{n} F^{ij}_{rs}y_{rs}=\sum_{r=1}^{m}\sum_{s=1}^{n} F^{ij}_{rs}x_{rs} \tag{3.5}$$

where the maximum is taken over matrices (y_{rs}) satisfying (1.1b-d), (1.3). This proves the theorem for the special case $C=F^{ij}$.

Let

$$f_{ij}=c_{ij}-c_{i,j+1}+c_{i+1,j+1}-c_{i+1,j}$$

for $i=1,\cdots,m-1$ and $j=1,\cdots,n-1$. Note that (1.2) implies $f_{ij}\geqq 0$. A direct calculation shows that

$$\sum_{i=r}^{u}\sum_{j=s}^{v} f_{ij}=c_{rs}-c_{r,v+1}+c_{u+1,v+1}-c_{u+1,s}$$

for any $u>r$ and $v>s$. In particular

$$\sum_{i=r}^{m-1}\sum_{j=s}^{n-1} f_{ij}=c_{rs}.$$

We can write this as

$$c_{rs}=\sum_{i=1}^{m-1}\sum_{j=1}^{n-1} f_{ij}F^{ij}_{rs},$$

or in matrix notation as

$$C=\sum_{i=1}^{m-1}\sum_{j=1}^{n-1} f_{ij}F^{ij}.$$

The fact that (x_{rs}) solves (1.1), (1.3) now follows immediately from (3.5) and the nonnegativity of the f_{ij}'s.

4. Possible extensions. The same techniques show that a greedy algorithm will solve (1.1), assuming (1.2) and some conditions more general than (1.3). For example, assume that $\{1,\cdots,m\}$ is partitioned into $\{1,\cdots,d_1\}$, $\{d_1+1,\cdots,d_2\},\cdots,\{d_k+1,\cdots,m\}$. In place of (1.3), assume (writing $d_0=0$, $d_{k+1}=m$)

$$\sum_{r=d_t+1}^{i}\sum_{s=1}^{j} x_{rs}\leqq \gamma_{ij} \quad \text{for } t=0,\cdots,k, \quad i=d_t+1,\cdots,d_{t+1},$$

where for each t, (3.3) holds for $d_t+1\leqq i\leqq d_{t+1}$. Then all the foregoing goes through. In particular, the partition $\{1\},\{2\},\cdots,\{m\}$ was very useful to us in [3]. It seems plausible that other variants of (1.3) might also be amenable to greedy algorithms.

REFERENCES

[1] G. MONGE, *Déblai et remblai*, Mémoires de l'Académie des Sciences, Paris, 1781.

[2] A. J. HOFFMAN, *On simple linear programming problems*, Convexity, Proc. Symposia in Pure Mathematics, Vol 7, American Mathematical Society, Providence, RI, 1961, pp. 317-327.

[3] E. R. BARNES AND A. J. HOFFMAN, *Partitioning spectra and linear programming*, Proc. Silver Jubilee Conference on Combinatorics, Univ. Waterloo, Waterloo, Ontario, Canada, June, 1982.

[4] M. FRÉCHET, *Sur les tableaux de correlation dont les marges sont données*, Ann. Univ. Lyon Section A, 14 (1951), pp. 53-77.

SIAM J. APPL. MATH.
Vol. 6, No. 3, July 1985

BITHRESHOLD GRAPHS*

P. L. HAMMER† AND N. V. R. MAHADEV‡

Abstract. A graph is called bithreshold if it is the edge-intersection of two threshold graphs T_1, T_2 and if every stable set of it is stable in T_1 or T_2. In this paper an easy recognition algorithm is proposed for this class of graphs and bithreshold graphs are proved to be strongly perfect.

1. Introduction, definitions and notation. All graphs considered in this paper are undirected, finite, loopless and have no multiple edges.

Let us denote by $E(G)$, the set of edges of a graph G, and by $V(G)$ the set of all vertices of G. We shall denote by (a, b), (a, b, c) and by (a, b, c, d) respectively an edge with end vertices a and b, a triangle on the vertices a, b, c and a 4-clique on a, b, c and d. The set of all vertices adjacent to a vertex x, will be denoted by $N(x)$.

A graph G is called (cf. [3]) a *threshold graph* if $N(x) \subseteq N(y) \cup \{y\}$ or $N(y) \subseteq N(x) \cup \{x\}$ for any pair of vertices x and y.

It has been shown in [3] that G is a threshold graph if and only if there are no four vertices x_1, x_2, x_3, x_4 in $V(G)$ inducing $2K_2$, P_4 or C_4. Graphs $2K_2$, P_4 and C_4 are illustrated in Fig. 1.

It has been shown in [3] that G is a threshold graph if and only if there are no four vertices x_1, x_2, x_3, x_4 in $V(G)$ inducing $2K_2$, P_4 or C_4. Graphs $2K_2$, P_4 and C_4 are illustrated in Fig. 1.

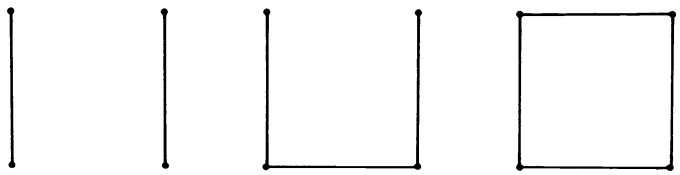


FIG. 1

The object of this paper is the study of *bithreshold graphs*, i.e., graphs G which are the edge-intersection of two threshold graphs T_1 and T_2 defined on the same vertex set, with the property that every stable set of G is also stable in T_1 or in T_2. In this case G is called *decomposable* into T_1 and T_2.

Obviously, the stability number of a bithreshold graph G is simply equal to the maximum of the stability numbers of T_1 and of T_2, and hence can easily be calculated.

The main result of this paper is a good recognition and decomposition algorithm for bithreshold graphs.

In § 2, we shall consider the general problem of recognizing monotonically increasing Boolean functions (also known as *positive Boolean functions*) which can be represented as the product (conjunction) of two "regular" functions. By a Boolean function f

* Received by the editors January 18, 1984, and in final revised form October 8, 1984. This research was supported in part by the Natural Sciences and Engineering Research Council under grant A-8552. This work was presented at the SIAM Second Conference on the Applications of Discrete Mathematics, held at Massachusetts Institute of Technology, Cambridge, Massachusetts, June 27-29, 1983.

† RUTCOR-Rutgers Center for Operations Research, Rutgers University, Hill Center, New Brunswick, New Jersey, 08903.

‡ Department of Combinatorics and Optimization, University of Waterloo, Waterloo, Ontario, N2L 3G1, Canada.

we mean a mapping of $\{0,1\}^n$ into $\{0,1\}$. A Boolean function $f(x_1, x_2, \cdots, x_n)$ is called *monotonically increasing* if $f(x_1, \cdots, x_{i-1}, 0, x_{i+1}, \cdots, x_n) \leqq f(x_1, \cdots, x_{i-1}, 1, x_{i+1}, \cdots, x_n)$ for any $i \in \{1, 2, \cdots, n\}$ and any values of $x_1, \cdots, x_{i-1}, x_{i+1}, \cdots, x_n$. If $\mathbf{x} = (x_1, x_2, \cdots, x_n)$ is any 0-1 vector, we denote by ${}^i\mathbf{x}^j$ the vector $(x_1, \cdots, x_{i-1}, 1, x_{i+1}, \cdots, x_{j-1}, 0, x_{j+1}, \cdots, x_n)$.

A monotonically increasing function is called *regular* if for any pair $i, j \in \{1, 2, \cdots, n\}$, the relation (1) $f({}^i\mathbf{x}^j) \geqq f({}^j\mathbf{x}^i)$ or the relation (2) $f({}^i\mathbf{x}^j) \leqq f({}^j\mathbf{x}^i)$ holds in any 0-1 point $\mathbf{x}$. If f is regular and (1) holds we shall say that $x_i \geqq_f x_j$, while if (2) holds we shall write $x_i \leqq_f x_j$. For a regular function, the relation "$\geqq$" being transitive defines a linear pre-order in the set of its variables. (The same relation could also be defined for arbitrary monotonic increasing functions, defining a partial preorder in the set of their variables.)

A Boolean function $f(x_1, \cdots, x_n)$ will be called *biregular* if there exist regular functions $f_1(x_1, \cdots, x_n)$ and $f_2(x_1, \cdots, x_n)$ such that (3) $f(x_1, \cdots, x_n) = f_1(x_1, \cdots, x_n) \cdot f_2(x_1, \cdots, x_n)$.

A monotonically increasing quadratic Boolean function (i.e., one which can be written in the form $f = c_1 \vee c_2 \vee \cdots \vee c_k$, where each c_i is the product of exactly two variables x_{i_1} and x_{i_2}) has an obvious graphic representation, obtained by associating a vertex to every variable and an edge to a conjunction; such functions will be called graphic. It will be shown in § 2 that a graphic function is biregular if and only if its associated graph is bithreshold.

A graph is called *signed* if every edge is given a (positive or negative) sign. A signed graph is called *balanced* if it contains no cycles involving an odd number of negative edges. It is well known that the recognition of balanced graphs can be done in $O(n^2)$ time. In § 3 we shall associate to an arbitrary graph G a signed graph H. It will be shown there that G is bithreshold if and only if the signed graph H' associated to its complement is balanced.

In the concluding § 4, we shall prove that bithreshold graphs are strongly perfect.

2. Decompositions of biregular graphic Boolean functions. A variable is Boolean if it takes values in $\{0, 1\}$. The complement $\bar{x}$ of a Boolean variable x is defined to be $1 - x$. Boolean variables and their complements are called literals.

A Boolean expression is recursively defined as follows.

(i) The constants 0, 1 and all literals are Boolean expressions.

(ii) The conjunction (product) of two Boolean expressions (the conjunction of a, b, denoted by $a \cdot b$, being defined as $\min(a, b)$) is a Boolean expression.

(iii) The disjunction (sum) of two Boolean expressions (the disjunction of a, b, denoted by $a \vee b$, being defined as $\max(a, b)$) is a Boolean expression.

(iv) The complement $\bar{a}$ of a Boolean expression a ($\bar{a}$ being defined as $1 - a$) is a Boolean expression.

(v) Every Boolean expression is obtained by applying the above three operations a finite number of times.

An expression of the form $c_1 \vee c_2 \vee \cdots \vee c_m$ where each c_i is a conjunction of one or more literals, is called a disjunctive form. Further if no c_i contains both a variable and its complement then the expression is called a *normal disjunctive form.*

LEMMA 2.1. *Every Boolean function in n variables $x_1, x_2, \cdots, x_n$ has an expression in normal disjunctive form.*

Proof. It is easy to check that

$$f(x_1, \cdots, x_n) = \vee_{(\alpha_1, \cdots, \alpha_n) \in \{0,1\}^n} f(\alpha_1, \cdots, \alpha_n) x_i^{\alpha_1} \cdots x_n^{\alpha_n}$$

is the n.d.f. for f where

$$x_i^{\alpha_i} = \begin{cases} x_i & \text{if } \alpha_i = 1, \\ \bar{x}_i & \text{if } \alpha_i = 0. \end{cases}$$ □

Further if the Boolean expression is monotonically increasing then it has an expression in normal disjunctive form in which no variable appears complemented.

DEFINITION 2.2. A Boolean expression is called *graphic* if it has a normal disjunctive form in which each conjunction contains exactly two distinct uncomplemented variables.

DEFINITION 2.3. If $f(x_1, x_2, \cdots, x_n) = c_1 \vee c_2 \vee \cdots \vee c_m$ is graphic then to f we associate the graph $G(f)$ with $V(G) = \{x_1, x_2, \cdots, x_n\}$ and $e = (x, y) \in E(G)$ iff $x \cdot y = c_i$ for some i. Conversely if G is any graph with $V(G) = \{1, 2, \cdots, n\}$ and $E(G) = \{e_1, e_2, \cdots, e_m\}$ we define a Boolean function $f(G) = c_1 \vee c_2 \vee \cdots \vee c_m$, where $c_i = x_k \cdot x_l$ iff $(k, l) = e_i$, $i \in \{1, \cdots, m\}$.

Thus a natural correspondence exists between graphs and graphic functions. It is easy to verify Lemma 2.4 given below.

LEMMA 2.4. *If f is graphic and G is the corresponding graph then $f(x) = 0$ if and only if x is the characteristic vector of a stable set in G.*

We recall that a biregular function is defined as the product of two regular functions. We shall prove that a graphic biregular function is the product of two graphic regular functions. The proof of the following lemma follows from the definition.

LEMMA 2.5. *If a positive Boolean function f is given in the form $f = Ax_ix_j \vee Bx_i \vee Cx_j \vee D$ where A, B, C, D are Boolean expressions not involving the variables x_i and x_j, then $x_i \geqq_f x_j$ if and only if $B \geqq C$.*

LEMMA 2.6. *If f is of the form described above (in Lemma 2.5) and $f' = f \vee Cx_i$ then* (i) *$x_i \geqq_f x_j$ and* (ii) *for any Boolean function g such that $x_i \geqq_g x_j$ and $f \leqq g$, we have $f \leqq f' \leqq g$.*

Proof. (i) $f' = Ax_ix_j \vee (B \vee C)x_i \vee Cx_j \vee D$. Since $B \vee C \geqq C$, $x_i \geqq_{f'} x_j$ by Lemma 2.5.

(ii) Let $F = \{x / f(x) = 1\} \cup \{{}^ix^j / f({}^ix^j) = 0 \text{ and } f({}^jx^i) = 1\}$ and let us define $f^*(x) = 1$ iff $x \in F$. Now $f(x) = 1 \Rightarrow g(x) = 1$ since $f \leqq g$ and $f({}^jx^i) = 1 \Rightarrow g({}^jx^i) = 1 \Rightarrow g({}^ix^j) = 1$ since $x_i \geqq_g x_j$. Thus $g(x) = 1$ for all $x \in F$, therefore $f^* \leqq g$. Also $f \leqq f^*$ since $f(x) = 1 \Rightarrow f^*(x) = 1$. In conclusion $f \leqq f^* \leqq g$. We now show that f^* has the form $f \vee Cx_i$. Then f^* is our required f', proving the lemma.

We apply the following three rules which are easy to verify.

If x, y and z are Boolean expressions then

$$\text{(i) } \overline{x \vee y} = \bar{x} \cdot \bar{y}, \quad \text{(ii) } (x \vee y) \cdot z = x \cdot z \vee y \cdot z, \quad \text{(iii) } x \vee \bar{x} \cdot y = x \vee y.$$

Notice that $f({}^ix^j) = B \vee D$ and $f({}^jx^i) = C \vee D$. Hence,

$$\begin{aligned} f^* &= f \vee \overline{(B \vee D)}(C \vee D)x_i\bar{x}_j \\ &= f \vee \bar{B} \cdot \bar{D}(Cx_i\bar{x}_j \vee Dx_i\bar{x}_j) \\ &= f \vee \bar{B}\bar{D}Cx_i\bar{x}_j \\ &= Ax_ix_j \vee Bx_i \vee Cx_j \vee D \vee \bar{B}Cx_i\bar{x}_j \\ &= Ax_ix_j \vee Bx_i \vee Cx_j \vee D \vee Cx_i\bar{x}_j \\ &= f \vee Cx_i. \end{aligned}$$

In conclusion $f^* = f'$, proving the lemma.

LEMMA 2.7. *If f is graphic and f^* is a regular function such that $f \leqq f^*$ then there exists a graphic regular function g such that $f \leqq g \leqq f^*$.*

Proof. Assume without loss of generality that $x_1 \geqq_{f^*} x_2 \geqq_{f^*} \cdots \geqq_{f^*} x_n$. If f is regular then $f = g$ and the lemma is proved; otherwise

(1) There exists i such that $x_i \not\geqq_f x_{i+1}$. Define f' as in Lemma 2.6 so that $x_i \geqq_f x_{i+1}$ and $f \leqq f' \leqq f^*$.

(2) If f' is regular $f' = g$ and the lemma is proved; otherwise let new $f = f'$ and go to step 1.

Since after each iteration $f' >$ old f', the process ends in a finite number of iterations. Thus the final g is regular satisfying $f \leqq g \leqq f^*$. Further after each iteration f' is graphic if f is graphic (see Lemma 2.6, applied to our case in which f is graphic and c is a single variable: thus $f \vee Cx_i$ is graphic). Thus g is graphic too, proving the lemma. □

THEOREM 2.8. *If f is graphic and biregular then there exist two graphic and regular functions f_1, f_2 such that $f = f_1 \cdot f_2$.*

Proof. By definition, there exist two regular functions f_1^*, f_2^* such that $f = f_1^* \cdot f_2^*$. Hence $f \leqq f_1^*$, $f \leqq f_2^*$.

Now, by Lemma 2.7 there exist two graphic regular functions f_1, f_2 such that $f \leqq f_1 \leqq f_1^*$ and $f \leqq f_2 \leqq f_2^*$. Hence $f \leqq f_1 \cdot f_2 \leqq f_1^* \cdot f_2^* = f$. Therefore, $f = f_1 \cdot f_2$, proving the theorem. □

Remark 2.9. If f is graphic and regular then $G(f)$ is threshold.

Remark 2.10. If f is graphic biregular and f_1, f_2 are two graphic regular functions such that $f = f_1 \cdot f_2$, then $G(f)$ is the intersection of $G(f_1)$ and $G(f_2)$; further, every stable set of G is also stable in $G(f_1)$ or $G(f_2)$. The converse is also true. Thus we have the following theorem.

THEOREM 2.11. *f is graphic biregular if and only if $G(f)$ is bithreshold.*

3. Recognition and decomposition of bithreshold graphs. It is easy to see that the complement of a threshold graph is also threshold. Thus the complement of a bithreshold graph which we shall call a *cobithreshold* graph, is the edge union of two threshold graphs T_1 and T_2, such that every clique of the graph is also a clique in T_1 or in T_2.

THEOREM 3.1. *If a graph G is the edge intersection of n graphs $G_1, G_2, \cdots, G_n$ and every stable set of size k, $3 \leqq k \leqq 2n$, in G is also stable in at least one of $G_1, G_2, \cdots, G_n$ then every stable set (of any size) of G is also stable in at least one of $G_1, G_2, \cdots, G_n$.*

Proof. Assume there exists a stable set S in G which is not stable in any of the graphs $G_1, G_2, \cdots, G_n$. Therefore the subgraph induced by S in G_i contains an edge e_i of $E(G_i)$ for $i = 1, 2, \cdots, n$. Let $S' \subseteq S$ be the set of all end vertices of the edges $e_1, e_2, \cdots, e_n$. Notice that not all of $e_1, e_2, \cdots, e_n$ define the same edge since otherwise it would be an edge in G induced by the vertices in S contradicting that S is stable in G. It follows now that $3 \leqq |S'| \leqq 2n$. But S' is not stable in any of $G_1, G_2, \cdots, G_n$; a contradiction proving the theorem. □

COROLLARY 3.2. *If G is the union of two threshold graphs T_1 and T_2 such that all 3-cliques and 4-cliques of G are also cliques of T_1 or T_2 then G is cobithreshold.*

Proof. Follows easily by applying Theorem 3.1 to $\bar{G}$. □

Let us associate to an arbitrary graph G, the *2-summability graph* G^* as follows:

The vertex set of G^* consists of edges of G. Two vertices of G^* are linked if and only if they correspond to edges of G whose end vertices induce in G a subgraph isomorphic to $2K_2$, P_4 or C_4.

Let $NV(G^*)$ denote the set of nonisolated vertices of G^*. Define now a *signed graph* $H(G)$ as follows:

The vertex set of H consists of nonisolated vertices of G^* and the set of negative edges of H consists of all the edges of G^*. Two vertices of H are joined by a positive

edge if and only if they correspond to edges of G whose end vertices induce in G a clique.

The example in Fig. 2 illustrates the graphs G^* and H associated with a graph G.

THEOREM 3.3. *If G is cobithreshold then the associated graph $H(G)$ is balanced.*

Proof. Let T_1 and T_2 be the two threshold graphs such that G is the union of T_1 and T_2 and every clique of G is also a clique of T_1 or T_2. Since the two end vertices e, f of any edge in G^* correspond to edges of G whose ends induce a subgraph isomorphic to $2K_2$, P_4 or C_4 in G, e and f belong to different threshold subgraphs of G. It follows that $NV(G^*)$ can be partitioned into two sets S_1 and S_2 where $S_1 = NV(G^*) \cap E(T_1)$ and $S_2 = NV(G^*) \cap E(T_2)$. Since no two vertices in S_i $(i = 1, 2)$ are adjacent in G^*, the edge-cut defined by the partition (S_1, S_2) contains all the negative edges of H (i.e., $E(G^*)$). We shall prove now that a cut (S_1, S_2) does not contain any positive edge of H. Assume there exists $e \in S_1$ and $f \in S_2$ such that (e, f) is a positive edge of H. Then e and f belong to a clique K of G. Then K is also a clique in T_1 or in T_2. If K is a clique in T_1 then $f \in S_1$ and if K is a clique in T_2 then $e \in S_2$. In either case it leads to a contradiction since S_1 and S_2 are disjoint. Thus the cut (S_1, S_2) contains no positive edge of H and contains all the negative edges of H. Hence by a theorem of Harary [4] H is balanced. □

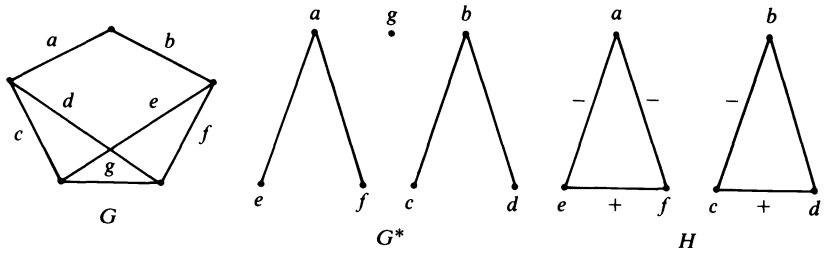


FIG. 2

We shall prove later that the converse of the above theorem is also true.

Suppose G is a graph such that its associated signed graph $H(G)$ is balanced. Construct two graphs G_1 and G_2 as described in the following four steps.

ALGORITHM A.

1) Obtain a partition (S_1, S_2), as shown in [4], of vertices of H such that the cut (S_1, S_2) consists of all the negative edges.
2) Let $\mathcal{K}$ be the set of all 3-cliques and 4-cliques in G. For $K \in \mathcal{K}$ and $S \subseteq V(H)$, we shall say that "K meets S" if i, j are vertices of K and the edge (i, j) of G corresponds to a vertex of H belonging to S.

   Let $\mathcal{K}_1 := \{K \in \mathcal{K} / K \text{ meets } S_1\}$ and $\mathcal{K}_2 := \{K \in \mathcal{K} / K \text{ meets } S_2\}$.

   Let $E(\mathcal{K}')$ be the set of vertices in G^* corresponding to edges in G that belong to a clique in $\mathcal{K}'$, where $\mathcal{K}' \subseteq \mathcal{K}$. Then $E_1 := S_1 \cup E(\mathcal{K}_1)$ and $E_2 := S_2 \cup E(\mathcal{K}_2)$.
3) Let $E(K)$ be the set of vertices in G^* corresponding to edges in the clique K.

   Let $\mathcal{K}_3 := \{K \in \mathcal{K} / E(K) \not\subseteq E_1 \text{ and } E(K) \not\subseteq E_2\}$.

   Let $E_1' := E_1 \cup E(\mathcal{K}_3)$ and $E_2' := E_2 \cup E(\mathcal{K}_3)$.
4) Let E be the set of all edges of G that do not correspond to any vertex of G^* belonging to E_1' or E_2'. Let G_1 and G_2 be the graphs defined on $V(G)$ such that $E(G_1) = E \cup \{\text{edges of } G \text{ corresponding to vertices in } E_1'\}$ and $E(G_2) = E \cup \{\text{edges of } G \text{ corresponding to vertices in } E_2'\}$.

We make the following observations about algorithm A.

1) $E_1 \cap S_2 = \emptyset = E_2 \cap S_1$. For, otherwise, if say, $E_1 \cap S_2 \neq \emptyset$ then since $S_1 \cap S_2 = \emptyset$, $E(\mathcal{K}_1) \cap S_2 \neq \emptyset$. It follows that there exists a clique $K \in \mathcal{K}_1$ that meets both S_1 and S_2 contradicting that cut (S_1, S_2) does not contain a positive edge.

2) $E_1' \cap S_2 = \emptyset = E_2' \cap S_1$. Follows using the above observation and the fact that K does not meet S_i $(i = 1, 2)$ for any K in $\mathcal{K}_3$.

LEMMA 3.4. *If G_1 and G_2 are constructed as in algorithm* A *using a graph G such that $H(G)$ is balanced then*

(A) *G is the edge union of G_1 and G_2.*

(B) *Every clique of G is also a clique in G_1 or G_2.*

(C) *The set of vertices of H corresponding to the edges of G_i $(i = 1, 2)$ do not induce a negative edge in H.*

Proof. (A) follows from the definition of E in step 4 of the algorithm.

(B) Follows from the definition of E_1' and E_2' in step 3 of the algorithm.

(C) By observation 2, $S_1 \cap E_2' = \emptyset = S_2 \cap E_1'$. It follows that $S_1 \cap E(G_2) = \emptyset = S_2 \cap E(G_1)$. Since each negative edge of H has one end in S_1 and the other end in S_2 (C) follows. □

For simplicity in proofs that follow we identify the edges of G with the corresponding vertices of G^*.

LEMMA 3.5. *If G_1 and G_2 are as in the previous lemma and if there exists a 3-clique (a, b, c) of G such that $(a, b) \in E(G_2) - E(G_1)$ and $(b, c) \in E(G_1)$ then there exists a vertex d such that $(c, d), (b, d) \in E(G_1)$ and $(a, d) \notin E(G)$.*

Figure 3 illustrates the lemma.

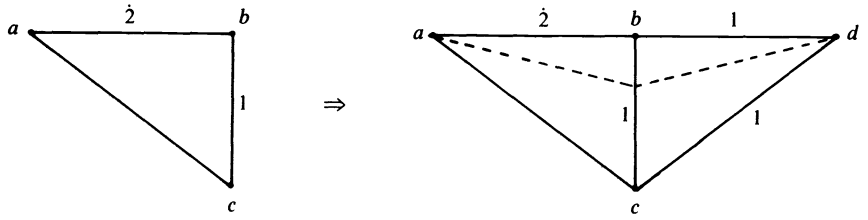


FIG. 3. *Broken lines indicate nonedges of G. Lines numbered 1 are edges of G_1 and the line numbered 2 is an edge in $E(G_2) - E(G_1)$.*

Proof. By hypothesis and by observing step (4) of the algorithm $(b, c) \in E(G_1) - E$. Hence $(b, c) \in S_1$ or $E_1 - S_1$ or $E_1' - E_1$.

Case 1. $(b, c) \in S_1$. It follows that $(a, b, c) \in \mathcal{K}_1$ and hence $(a, b) \in E_1 \subseteq E(G_1)$, a contradiction to the hypothesis. Therefore $(b, c) \notin S_1$.

Case 2. $(b, c) \in E_1 - S_1$. It follows that $(b, c) \in E(K_1)$. Let $K^* \in \mathcal{K}_1$ be such that $(b, c) \in E(K^*)$ and $(d, e) \in E(K^*) \cap S_1$.

Subcase 2.1. *Both d* and *e* are distinct from b and c. If a, b, d, e induce a 4-clique in G then such a clique is in $\mathcal{K}_1$. Hence by Step 2, $(a, b) \in E(G_1)$, a contradiction to the hypothesis. It follows that a, b, d, e do not induce a 4-clique. Hence (a, d) or (a, e) is a nonedge in G. Assume without loss of generality that (a, d) is a nonedge in G. Then the claim is proved in this case.

Subcase 2.2. *e* coincides with b or c. If a, b, c, d induce a 4-clique in G then such a clique is in $\mathcal{K}_1$. Hence by Step 2, $(a, b) \in E(G_1)$, again a contradiction to the hypothesis. It follows that a, b, c, d do not induce a 4-clique in G. Hence (a, d) is a nonedge. Then the claim is proved.

Case 3. $(b, c) \in E_1' - E_1$. There exists a $K^* \in \mathcal{K}_3$ such that $(b, c) \in E(\mathcal{K}^*)$.

Subcase 3.1. $V(K^*) = \{b, c, d\}$. If $(a, d) \in E(G)$ then a, b, c, d induces a 4-clique (a, b, c, d). But $(a, b, c, d) \not\subseteq E_1 \cup \mathcal{K}_3$ since $(a, b) \notin E_1 \cup \mathcal{K}_3$. Therefore $(a, b, c, d) \subseteq E_2$. Hence $E(K^*) \subseteq E_2$ contradicting that $K^* \in \mathcal{K}_3$. Hence, $(a, d) \notin E(G)$. Thus the claim is true in this case.

Subcase 3.2. $V(K^*) = \{b, c, d, e\}$. If $(a, d), (a, e) \in E(G)$ then a, b, c, d induce a 4-clique (a, b, d, e). But $(a, b, d, e) \not\subseteq E_1 \cup \mathcal{K}_3$ since $(a, b) \notin E_1 \cup \mathcal{K}_3$. Therefore $(a, b, d, e) \subseteq E_2$. By similar reasoning $(a, b, c, d) \subseteq E_2$ and $(a, b, c, e) \subseteq E_2$.

Thus $K^* = (b, c, d, e) \subseteq E_2$, implying that $K^* \notin \mathcal{K}_3$, a contradiction. Therefore, (a, d) or $(a, e) \notin E(G)$. Assume without loss of generality that $(a, d) \notin E(G)$. Then the lemma is proved in this case.

Since all possible cases are examined, we conclude that the lemma is true in general. □

LEMMA 3.6. *If G_1, G_2 are constructed by algorithm* A *then both G_1 and G_2 are threshold graphs.*

Proof. We prove that G_1 is a threshold graph. By a similar argument G_2 is also a threshold graph.

Assume that G_1 is not a threshold graph. Then there exists four distinct vertices a, b, c, d such that $(a, b) \notin E(G_1)$, $(c, d) \notin E(G_1)$, $(a, c) \in E(G_1)$ and $(b, d) \in E(G_1)$. Since (a, c), (b, d) do not induce a negative edge in H (by Lemma 3.4) (a, b) or $(c, d) \in E(G)$ and (a, d) or $(b, c) \in E(G)$. It follows that it is enough to consider the following 4 cases.

(i) $(a, b) \in E(G) - E(G_1), (a, d) \in E(G), (c, d) \notin E(G)$ and $(b, c) \notin E(G)$.

(ii) $(a, b) \in E(G) - E(G_1), (c, d) \notin E(G), (a, d) \in E(G)$ and $(b, c) \in E(G)$.

(iii) $(a, b) \in E(G) - E(G_1), (c, d) \in E(G) - E(G_1), (a, d) \in E(G)$ and $(b, c) \notin E(G)$.

(iv) $(a, b) \in E(G) - E(G_1), (c, d) \in E(G) - E(G_1), (a, d) \in E(G)$ and $(b, c) \in E(G)$.

All four cases are illustrated in Fig. 4. We use the same conventions as in Fig. 3.

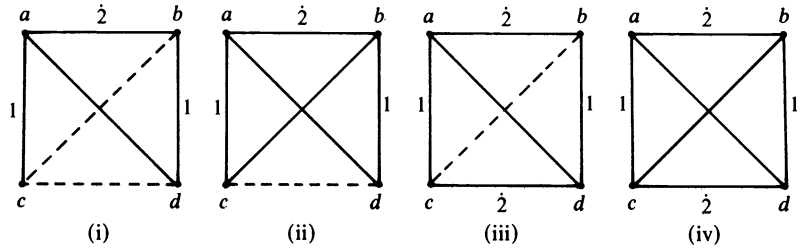


FIG. 4

In each of the four cases we shall prove that there exists a negative edge in G_i, contradicting statement C of Lemma 3.4.

(i) Apply Lemma 3.5 on (a, b, d). Then there exists e such that (b, d), (d, e) are edges in G_1 and $(a, e) \notin E(G)$. But then (b, e), (a, c) of G_1 induce a negative edge in H.

(ii) Apply Lemma 3.5 on (a, b, d). Then there exists e such that (b, e), (d, e) are edges in G_1 and $(a, e) \notin E(G)$. But then (d, e), (a, c) of G induce a negative edge in H.

(iii) Same as in case (i).

(iv) Apply Lemma 3.5 on (a, b, c) and (a, b, d). There exist e and f such that (b, e), (a, f) are in $E(G_1)$ and (a, e), (b, f) are not in $E(G)$. But then (b, e) and (a, f) of G_1 induce a negative edge in H.

Since in all four cases we get a contradiction, G_1 is a threshold graph. The lemma follows. □

THEOREM 3.7. *If $H(G)$ is balanced then G is cobithreshold.*

Proof. Since H is balanced we can obtain G_1 and G_2 as in algorithm A. By Lemmas 3.5 and 3.6, the theorem follows. □

By Theorems 3.3 and 3.7, it follows that a graph G is bithreshold if and only if the signed graph associated to its complement is balanced.

Recognizing and decomposing bithreshold graphs. The above theorem gives rise to the following method to recognize and decompose bithreshold graphs. Let G be the given graph.

(1) Construct the signed graph H associated with the complement of G.

(2) *If H* is not balanced then G is not bithreshold. If H is balanced go to step (3), noting that G is bithreshold.

(3) Construct the two threshold graphs G_1, G_2 as in algorithm A.

(4) Then G is decomposable into the complements of G_1 and G_2.

The complexity of this algorithm is at most $O(n^4)$.

4. Strong perfectness of bithreshold graphs. A graph G is *strongly perfect* if for any induced subgraph H there exists a stable set meeting all the maximal cliques of H. Strongly perfect graphs are also perfect [1].

We prove that bithreshold graphs and their complements are strongly perfect.

A graph G is *perfectly orderable* if there exists an ordering of its nodes such that no four nodes, say a, b, c, d inducing the edge set $\{(a, b), (b, c), (c, d)\}$ have the order $a < b$ and $d < c$.

Chvatal [2] proved that perfectly orderable graphs are strongly perfect. We prove that the intersection of two threshold graphs is perfectly orderable. It follows then that bithreshold graphs are strongly perfect.

THEOREM 4.1. *If G is the intersection of two threshold graphs T_1 and T_2 then G is perfectly orderable.*

Proof. Let us order the vertices of G in nonincreasing degrees in T_1. Thus x may precede y only if $\deg(x) \geqq \deg(y)$. We prove that this order is a perfect order for G. Consider any four vertices, say a, b, c, d inducing the edgeset $\{(a, b), (b, c), (c, d)\}$. It is enough to show that in T_1 $\deg(a) < \deg(b)$ or $\deg(d) < \deg(c)$. Now since T_i (for $i = 1, 2$) is a threshold graph containing all the edges of G, it follows that (a, c) or (b, d) is an edge in T_i.

Case 1. $(a, c) \in E(T_1)$. Hence $(a, c) \notin E(T_2)$. Therefore $(b, d) \in E(T_2)$, implying that $(b, d) \notin E(T_1)$. It follows that $\deg(d) < \deg(c)$ in this case.

Case 2. $(b, d) \in E(T_1)$. By a similar argument as in the above case it follows that $(a, c) \in E(T_1)$ implying that $\deg(a) < \deg(b)$ in this case.

Thus have we proved that the order is a perfect order and hence G is perfectly orderable. □

THEOREM 4.2. *If G is the union of two threshold graphs T_1 and T_2 then G is strongly perfect.*

Proof. Since every induced subgraph of G is also a union of two threshold graphs, it is enough to show that G has a stable set S meeting all the maximal cliques of G.

Let x be any vertex of largest degree in T_1. Let y be a vertex of largest degree in T_2 among all the vertices in

$$V(G) - (N(x) \cup \{x\}).$$

Let $S = V(G) - N(x) - N(y) \cdots (*)$.

Notice that both x and y are in S. We first show that S is a stable set.

Assume a and b are two vertices in S such that $(a, b) \in E(G)$. Then both a and b are different from x and y by the construction of S. Further, neither a nor b is adjacent to x or y for the same reason. It is well known that in a threshold graph T a vertex of largest degree is adjacent to all nonisolated vertices of T [3]. Hence if (a, b) is an edge in T_1 then x is adjacent to both a and b, which is not possible. Similarly if (a, b) is an edge in T_2 then y is adjacent to both a and b, which is not possible. It follows that (a, b) is not an edge in G and S is a stable set.

We shall now show that S meets all maximal cliques of G. Let K be any maximal clique of G not meeting x. Then K contains a vertex z not adjacent to x. Hence for each $l(\neq z)$ in K the edge (l, z) is in T_2. Hence K meets y unless K consists of z alone and z is not adjacent to y as well. But in this case $z \in S$. Thus S meets all maximal cliques of G. Hence G is a strongly perfect graph. □

COROLLARY. *Cobithreshold graphs are strongly perfect.*

Remark 4.1. The proof of the above theorem can be used to obtain a minimum coloring for graphs that are unions of two threshold graphs T_1 and T_2 if T_1 and T_2 are given. For instance $S_1, S_2, \cdots, S_k$ is a partition of $V(G)$ into a minimum number of stable sets where for each i, S_i is obtained from the graph induced by $(V(G) - \cup_{l=1}^{i-1} S_l)$ as given by (*) in the above theorem.

Remark 4.2. As we mentioned earlier, a maximum stable set for bithreshold graphs can be obtained in polynomial time. Further by the above remark a minimum partition of the vertex set into cliques can be obtained in polynomial time. Also given a perfect ordering of a graph G, one can obtain a minimum coloring and a maximum clique for G in polynomial time as is clear from [2]. Hence by Theorem 4.1, it follows that

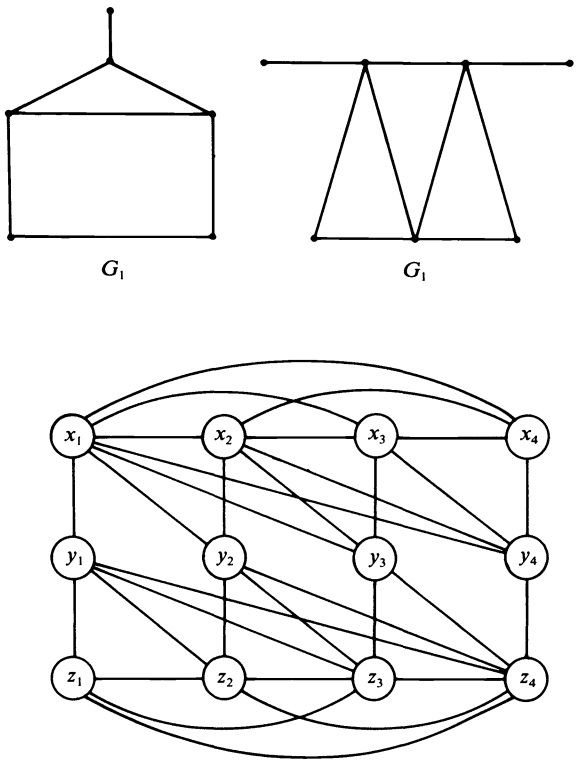


FIG. 5

minimum coloring and a maximum clique can also be obtained for bithreshold graphs in polynomial time.

Remark 4.3. A bithreshold graph need not be a comparability graph, e.g. G_1 of Fig. 5.

A cobithreshold graph need not be a comparability graph, e.g. G_2 of Fig. 5.

Neither a bithreshold graph nor its complement need be triangulated, e.g. G_1 of Fig. 5.

Remark 4.4. Given any positive integer n, there exists a bithreshold graph whose Dilworth number as well as its complement's Dilworth number is n. For example consider the graph H_n with vertices $x_1, x_2, \cdots, x_n, y_1, y_2, \cdots, y_n$ and $z_1, z_2, \cdots, z_n$ such that $\{x_1, \cdots, x_n\}$ induces a clique $\{z_1, \cdots, z_n\}$ induces a clique, $\{y_1, \cdots, y_n\}$ induces a stable set and each x_i is linked to y_j for $i \geqq j$ and each z_i is linked to y_j for $i \leqq j$. Graph H_n is cobithreshold and both H_n and $\bar{H}_n$ have Dilworth number n. See Fig. 5 for H_4.

REFERENCES

[1] C. BERGE AND P. DUCHET, *Strongly perfect graphs*, in Topics in Perfect Graphs, C. Berge and V. Chvatal, eds., to appear.

[2] V. CHVATAL, *Perfectly orderable graphs*, Report SOCS 81-82, McGill Univ., Montreal, Quebec.

[3] V. CHVATAL AND P. L. HAMMER, *Aggregation of inequalities in integer programming*, Ann. Discr. Math., 1 (1977), pp. 145-162.

[4] F. HARARY, *On the notion of balance of a signed graph*, Michigan Math. J., 2 (1953), pp. 143-146.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 3, July 1985

016

AN ASYMPTOTIC APPROACH TO THE CHANNEL ASSIGNMENT PROBLEM*

JOSHUA H. RABINOWITZ† AND VIERA KRŇANOVÁ PROULX‡

Abstract. This paper introduces a new approach to the T-coloring problem for complete graphs. The problem arises from Hale's formulation of the channel assignment problem for potentially interfering communication nets. The motivating result of this paper is that the T-span of K_n, denoted $\mathrm{sp}_T(K_n)$, is asymptotically independent of n. More precisely, each T-set has a rate, $\mathrm{rt}(T)$, and $n/\mathrm{sp}_T(K_n)$ converges to $\mathrm{rt}(T)$. We introduce a finite algorithm for computing the rate of T. This is accomplished by associating to a given set T an infinite sequence of integers with the property that the first n integers of this sequence T-color K_n in an asymptotically optimal way. Lastly, we compute $\mathrm{rt}(T)$ or bounds on its value for some interesting special cases of sets T.

AMS(MOS) subject classification. 05

1. Introduction. This paper introduces a new approach to the T-coloring problem for complete graphs. The problem arises from Hale's formulation of the channel assignment problem for potentially interfering communication nets (see [4], [2]). Let T be a nonempty finite set of positive integers and let $G=(V, E)$ be a (simple) graph. A T-coloring of G is an assignment $f: V\to Z$ of integers to the vertices of G such that $|f(v)-f(w)|\notin T\cup\{0\}$ whenever v and w are adjacent in G. The span of a T-coloring is the difference between the largest and smallest integers assigned and the T-span is the minimal possible span for a T-coloring of G.

Cozzens and Roberts [2] consider the problem of determining the T-span of the complete graph, K_n, for arbitrary T and n. A complete solution to this problem would, by a result in [2], also determine the T-span of all weakly γ-perfect graphs and would provide an upper bound for the general case. The problem has been solved in the special case where T is "r-initial"; in this case, the optimal coloring is obtained by a greedy-type algorithm (see [2]). However, a greedy-type algorithm does not work in the general case. This means that for some set T and integer n, an optimal coloring of K_{n+1} will not contain (as a subset) an optimal coloring of K_n. For general background and a discussion of applications, the reader is referred to [4].

The motivating result of this paper is that the T-span of K_n, denoted $\mathrm{sp}_T(K_n)$, is asymptotically independent of n. More precisely, let $\mathcal{T}$ be the family of finite nonempty sets of positive integers and let $I=\{q\in Q \mid 0<q\leqq\frac{1}{2}\}$, where Q denotes the rationals. Then there is a "rate" function $\mathrm{rt}: T\to I$ such that

$$\lim_{n\to\infty}\frac{n}{\mathrm{sp}_T(K_n)}=\mathrm{rt}(T)$$

for $T\in\mathcal{T}$.

Moreover, we introduce a finite algorithm for computing the rate of T. This is accomplished by associating to a given set T an infinite sequence of integers with the

* Received by the editors May 31, 1983, and in revised form April 11, 1984. This research was accomplished as part of the Mitre Corporation's Independent Research and Development Program. This work was presented at the SIAM Second Conference on the Applications of Discrete Mathematics, held at Massachusetts Institute of Technology, Cambridge, Massachusetts, June 27-29, 1983.

† The Mitre Corporation Bedford, Massachusetts 01730. Current address, Gould Defense Systems, Inc., NavCom Systems Division, El Monte, California 91731.

‡ The Mitre Corporation Bedford, Massachusetts 01730, and College of Computer Science, Northeastern University, Boston, Massachusetts 02115.

property that the first n elements of this sequence T-color K_n in an asymptotically optimal way. The algorithm is exponential in the largest element of T. It does, however, enable us to T-color K_n in an asymptotically optimal way in constant time (as a function of n). Lastly, we compute $\mathrm{rt}(T)$ or bounds on its value for some interesting special cases of sets T.

2. The rate of T-sets. The main purpose of this section is to show that

$$\lim_{n\to\infty} n/\mathrm{sp}_T(K_n)$$

exists and, thus, that $\mathrm{rt}(T)$ is well defined. We begin with a few basic definitions.

DEFINITION 2.1. Let T be a set of positive integers and let G be a simple graph, that is, a graph with no loops or multiple edges. A *T-coloring* of G is a function $f: V(G) \to Z$, Z the set of integers, such that $|f(v)-f(w)|$ is not an element of $T \cup \{0\}$ whenever v and w are adjacent in G.

DEFINITION 2.2.

(a) The *span* of a T-coloring is the difference between the largest and smallest integers used in the coloring.

(b) The *T-span* of G, denoted $\mathrm{sp}_T(G)$ is the minimum possible span for a T-coloring of G.

DEFINITION 2.3. The *complete graph* on n vertices, denoted K_n, is the graph with n vertices each pair of which is connected by an edge.

The object of this paper is to address the problem of determining the T-span of K_n for arbitrary T and n. At first glance, one is led to believe that the problem must be considered anew for each T and each n. Evidence in support of this hypothesis can be drawn from an example given in [2]. Let $T=\{1,4,5\}$; the T-span of K_2 is 2 with $(1,3)$ an optimal coloring while the T-span of K_3 is 6 with $(1,4,7)$ optimal. Thus, one cannot simply extend an optimal T-coloring of K_2 to obtain one for K_3. This is, incidentally, a rephrasing of the point made in [2] that the greedy algorithm does not, in general, provide minimal span colorings. We shall see, however, that in point of fact, one need not consider the T-span problem independently for each n.

We begin by relating our problem to the study of maximal independent sets in circulant graphs.

DEFINITION 2.4. Let T be a finite set of positive integers.

(a) $G(T)$ is the infinite graph defined as follows:
  (1) $V(G(T))=Z$ (the integers);
  (2) $\{v, w\} \in E(G(T))$ if and only if $|v-w| \in T$.

(b) $H(n, T)$ is the finite vertex subgraph of $G(T)$ defined as follows:
  (1) $V(H(n, T))=\{0, \cdots, n-1\}$;
  (2) $\{v, w\} \in E(H(n, T))$ if and only if $|v-w| \in T$.

(c) $G(n, T)$ is the finite point symmetric graph defined as follows:
  (1) $V(G(n, T))=Z_n$ (the ring of integers mod n);
  (2) $\{v, w\} \in E(G(n, T))$ if either $(v-w)_n \in T$ or $(w-v)_n \in T$, where $(x)_n = x \bmod n$.

We note that $G(n, T)$ is a *circulant graph*, that is, a graph with a circulant adjacency matrix. For fixed T and n, let S_n be the union of T and all integers of the form $n-x$, where $x \in T$, $x<n$. Then, in the notation of [5] and [1], $G(n, T)=G(n, S_n)$, that is, S_n is the *symbol* of $G(n, T)$.

DEFINITION 2.5. The *independence number* of a graph is the cardinality of a largest possible set of vertices in the graph, no two of which are adjacent.

Notation 2.6.

$\alpha(n, T) :=$ independence number of $G(n, T)$.

$\beta(n, T) :=$ independence number of $H(n, T)$.

LEMMA 2.7. (a) $\beta(m+n, T) \leqq \beta(m, T) + \beta(n, T)$.

(b) $\beta(n - \max(T), T) \leqq \alpha(n, T) \leqq \beta(n, T)$ *where* $\max(T)$ *is the largest element of* T.

(c) $\alpha(n+1, T)$ *is not necessarily greater than or equal to* $\alpha(n, T)$.

Proof. (a) This follows immediately from the definition.

(b) Since $H(n, T)$ is a subgraph of $G(n, T)$ with $V(H(n, T)) = V(G(n, T))$ and $E(H(n, T)) \subset E(G(n, T))$, we must have $\alpha(n, T) \leqq \beta(n, T)$. For the first inequality, let $\{x_1, \cdots, x_A\}$ be an independent set in $H(n - \max(T), T)$ of cardinality $\beta(n - \max(T), T)$. Then $\{x_1, \cdots, x_A\} \subset \{0, \cdots, n - \max(T) - 1\} \subset \{0, \cdots, n-1\} = V(G(n, T))$. Without loss of generality, assume $x_i < x_j$ for $i < j$. Then, $(x_j - x_i)_n = |x_j - x_i| = x_j - x_i \notin T$ for $i < j$ by assumption and $(x_i - x_j)_n \geqq \max(T) + 1$ so $(x_i - x_j)_n \notin T$ either. It follows that $\{x_1, \cdots, x_A\}$ is independent in $G(n, T)$ and hence that $\alpha(n, T) \geqq A = \beta(n - \max(T), T)$.

(c) For $T = \{1, 3\}$, we have $\alpha(6, T) = 3$ and $\alpha(7, T) = 2$. Q.E.D.

LEMMA 2.8. (a) $\mathrm{sp}_T(K_n) \leqq m$ *if and only if* $\beta(m+1, T) \geqq n$.

(b) $\beta(1 + \mathrm{sp}_T(K_n)) = h$.

(c) $\mathrm{sp}_T(K_{m+n}) \geqq \mathrm{sp}_T(K_m) + \mathrm{sp}_T(K_n)$.

Proof. (a) $\mathrm{sp}_T(K_n) \leqq m$ if and only if one can T-color K_n using integers selected from the set $\{0, \cdots, m\} = V(H(m+1, T))$. Since all n vertices of K_n are mutually adjacent, this can be done if, and only if, $H(m+1, T)$ has an independent set of size at least n.

(b) This follows from (a) and the fact that $\mathrm{sp}_T(K_{n+1}) > \mathrm{sp}_T(K_n)$.

(c) Let $r = \mathrm{sp}_T(K_{m+n})$ with $\{0 = x_1, x_2, \cdots, x_{m+n} = r\}$ an optimal T-coloring of K_{m+n}. Then $\{x_1, \cdots, x_m\}$ colors K_m and $\{x_{m+1}, \cdots, x_{m+n}\}$ colors K_n. Thus, $\mathrm{sp}_T(K_m) \leqq x_m$ and $\mathrm{sp}_T(K_n) \leqq r - x_{m+1}$ so that $\mathrm{sp}_T(K_{m+n}) = r = (r - x_{m+1}) + (x_{m+1} - x_m) + x_m \geqq x_{m+1} - x_m + \mathrm{sp}_T(K_m) + \mathrm{sp}_T(K_n) \geqq \mathrm{sp}_T(K_m) + \mathrm{sp}_T(K_n)$, with equality possible only if n or m is 0.

THEOREM 2.9. *For any fixed* T, *the limits*

$$\lim_{n\to\infty} \frac{n}{\mathrm{sp}_T(K_n)}, \quad \lim_{n\to\infty} \frac{\beta(n, T)}{n}, \quad \lim_{n\to\infty} \frac{\alpha(n, T)}{n}$$

exist and are equal.

Proof. It follows from Lemma 2.8c and a theorem of Pólya-Szegö [6, p. 17] that

$$\lim_{n\to\infty} \frac{\mathrm{sp}_T(K_n)}{n}$$

exists and that

$$\lim_{n\to\infty} \frac{\mathrm{sp}_T(K_n)}{n} = \sup \frac{\mathrm{sp}_T(K_n)}{n}.$$

Similarly, by Lemma 2.7a,

$$\lim_{n\to\infty} \frac{\beta(n, T)}{n} \text{ exists} \quad \text{and} \quad \lim_{n\to\infty} \frac{\beta(n, T)}{n} = \inf \frac{\beta(n, T)}{n}.$$

Thus, by Lemma 2.8c we have

$$\left(\lim_{m\to\infty} \frac{\beta(m, T)}{m}\right)\left(\lim_{n\to\infty} \frac{\mathrm{sp}_T(K_n)}{n}\right) = \left(\lim \frac{\beta(1 + \mathrm{sp}_T(K_n), T)}{1 + \mathrm{sp}_T(K_n)}\right)\left(\frac{\mathrm{sp}_T(K_n)}{n}\right) = 1.$$

This proves that

$$\lim_{n\to\infty} \frac{n}{\mathrm{sp}_T(K_n)} \quad \text{and} \quad \lim_{n\to\infty} \frac{\beta(n, T)}{n}$$

exist and are equal. Finally, it follows from Lemma 2.7(b) that

$$\lim_{n\to\infty} \frac{\alpha(n, T)}{n} = \lim_{n\to\infty} \frac{\beta(n, T)}{n}. \qquad \text{Q.E.D.}$$

DEFINITION 2.10. Let $\mathcal{T}$ be the family of finite nonempty sets of positive integers and let $J = \{r \in R: 0 \leqq r \leqq 1\}$, where R denotes the real numbers. The *rate* function $\mathrm{rt}: \mathcal{T} \to J$ is given by

$$\mathrm{rt}(T) = \lim_{n\to\infty} \frac{n}{\mathrm{sp}_T(K_n)}$$

for $T \in \mathcal{T}$.

We shall see later on that $0 < \mathrm{rt}(T) \leqq \frac{1}{2}$ for all T and that $\mathrm{rt}(T)$ is always a rational number.

3. An algorithm for the determination of rt (T). In this section we show that, for any T-set, the rate can be calculated in finite time. This is important, since the definition of the rate involves the determination of $\mathrm{sp}_T(K_n)$ for some infinite sequence of values of n. The technique is constructive in that it not only calculates $\mathrm{rt}(T)$ but also provides T-colorings of K_n for each n which realize this asymptotic rate. More precisely, for each T we construct an infinite sequence, the first n positive elements of which are to be used to T-color K_n (see the discussion following Theorem 3.14). Unfortunately, the algorithm is, in general, highly exponential. Nevertheless, there are interesting cases where the answer is produced quite rapidly and these are discussed at the end of this section. Subsequent sections deal with much more efficient techniques for calculating $\mathrm{rt}(T)$. These latter techniques, however, deal with restricted classes of T-sets.

LEMMA 3.1.

$$\mathrm{rt}(T) = \lim_{n\to\infty} \frac{\alpha(n, T)}{n}.$$

Proof. This follows at once from Theorem 2.9 and Definition 2.10.

For simplicity of notation, we now fix a set T and write $\alpha(n)$ insread of $\alpha(n, T)$. Also, we let $m = \max(T)$.

SUBLEMMA 3.2. *If* $n \geqq m+1$ *and* $k > 1$, *then* $\alpha(kn) \geqq k\alpha(n)$.

Proof. Let $\{x_1, \cdots, x_r\}$ be an independent set in $V(G(n, T)) = \{0, \cdots, n-1\}$ with $r = \alpha(n)$ and such that $x_i < x_j$ for $i < j$. Then, for $i < j$, $x_j - x_i = (x_j - x_i)_n \notin T$ and $(x_i - x_j)_n = n + x_i - x_j \notin T$. Fix k and consider the set $S_k \subset Z_{kn} = \{0, \cdots, kn-1\}$ defined by

$$S_k = \{y \in Z_{kn} | y = x_i + an \text{ for some } i \text{ and } a \text{ with } 1 \leqq i \leqq r \text{ and } 0 \leqq a \leqq k-1\}.$$

Since S_k has $kr = k\alpha(n)$ elements, it suffices to demonstrate that S_k is an independent set in $V(G(kn)) = Z_{kn}$. Suppose then that $y, z \in Z_{kn}$ with $y < z$ and write $y = x_i + an$ and $z = x_j + bn$. We must show that $(z-y)_{kn} = z - y \notin T$ and $(y-z)_{kn} = kn + y - z \notin T$. There are several cases to consider.

Case 1. $[b = a, x_i < x_j]$. Then $z - y = x_j - x_i \notin T$ by assumption and $kn + y - z \geqq 2n + y - z \geqq n \geqq m+1$. Since m is the largest element of T, it follows that $kn + y - z \in T$.

Case 2. $[b = a+1,\ x_i \leqq x_j]$. Then $z - y \geqq n \geqq m+1$ and $kn + y - z = (k-1)n + x_i - x_j$. This latter term is $\geqq n$ if $k \geqq 3$ and equal to $n + x_i - x_j$ if $k = 2$. In either case, it is not in T.

Case 3. $[b = a+1,\ x_i > x_j]$. Then $z - y = n + x_j - x_i \in T$ and $kn + y - z = (k-1)n + x_i - x_j > (k-1)n \geqq n \geqq m+1$.

Case 4. $[a+2 \leqq b \leqq a+k-2]$. Then $z - y \geqq n$ and $kn + y - z \geqq kn + an - (b+1)n \geqq kn + an - (a+k-1)n = n$.

Case 5. $[a = 0,\ b = k-1]$. If $k = 2$, this is Case 2; thus, assume $k \geqq 3$. Then $z - y \geqq n$ and $kn + y - z = kn + x_i - x_j - (k-1)n = n + x_i - x_j$. If $x_i < x_j$, then $n + x_i - x_j \notin T$ by assumption and if $x_i \geqq x_j$, then $n + x_i - x_j \geqq n$.

It is easy to see that these are all the possible cases. In fact, $z > y$ implies that $b \geqq a$ and moreover that if $b = a$, then $x_j > x_i$. Cases 1-3, therefore, cover all instances where $b = a$ or $b = a+1$. Since $b \leqq k-1$, the possibility that $b > a+k-2$ only occurs when $a = 0$ and $b = k-1$. Thus, Cases 4 and 5 cover all instances where $b \geqq a+2$. Q.E.D.

LEMMA 3.3. *If* $n \geqq m+1$, *then* $\mathrm{rt}(T) \geqq \alpha(n)/n$.

Proof. Since the sequence $\{\alpha(n)/n\}$ converges to $\mathrm{rt}(T)$, any subsequence also converges to $\mathrm{rt}(T)$. In particular

$$\mathrm{rt}(T) = \lim_{k \to \infty} \frac{\alpha(kn)}{kn}$$

for any fixed n. If $n \geqq m+1$, then by Sublemma 3.2 we have

$$\frac{\alpha(kn)}{kn} \geqq \frac{k\alpha(n)}{kn} = \frac{\alpha(n)}{n}.$$

It follows that

$$\mathrm{rt}(T) = \lim_{k \to \infty} \frac{\alpha(kn)}{kn} \geqq \lim_{k \to \infty} \frac{\alpha(n)}{n} = \frac{\alpha(n)}{n}.$$ Q.E.D.

LEMMA 3.4. *If* $n > 2^m$, *then there exists an integer* s *such that*

(a) $m+1 \leqq s \leqq 2^m$,

(b) $\alpha(n)/n \leqq \alpha(s)/s$.

Proof. Suppose $n > 2^m$ and let $X = \{x_1, \cdots, x_r\}$ be an independent set in $V(G(n, T)) = Z_n$ with $x_i < x_j$ for $i < j$ and $r = \alpha(n)$. Define a periodic binary sequence $(a_i)_{i \in Z}$ of period n by setting $a_i = 0$ if $(i)_n \notin X$ and $a_i = 1$ if $(i)_n \in X$. Let S_i by the m-vector

$$S_i = (a_i, a_{i+1}, \cdots, a_{i+m-1}).$$

Since $n > 2^m$ and there are only 2^m different m-vectors, there must exist an i and j with $0 \leqq i < j \leqq n-1$ and such that $S_i = S_j$. The idea of the proof is to use S_i and S_j to decompose X into two parts, one of which will yield an s such that $m+1 \leqq s < n$ and $\alpha(s)/s \geqq \alpha(n)/n$. We note that this is sufficient for the proof of the lemma since the process can be repeated, if necessary, until $s \leqq 2^m$.

It is crucial in what follows to differentiate the case where $(j-i)_n \geqq m+1$ and $(i-j)_n \geqq m+1$ from the case where one of these is $\leqq m$. Note that if $(i-j)_n \leqq m$ and $(j-i)_n \leqq m$, then $n \leqq 2m \leqq 2^m$ contradicting our assumption. We proceed to consider the two cases separately.

Case 1. $[(i-j)_n \geqq m+1$ and $(j-i)_n \geqq m+1]$. Let $X' \subset Z_{j-i}$ be defined by $x \in X'$ whenever $a_{i+x}=1$ and let $X'' \subset Z_{n+i-j}$ be defined by $x \in X''$ whenever $a_{j+x}=1$. We claim that:

(*) $|X|=|X'|+|X''|$ where $|A|$ is the cardinality of A.

(**) X' and X'' are independent sets in $G(j-i, T)$ and $G(n+i-j, T)$, respectively.

Assuming (*) and (**) and recalling that we are in the case where $j-i$ and $n+i-j$ are both $\geqq m+1$, it suffices to show that $\alpha(j-1)/(j-i)$ or $\alpha(n+i-j)/(n+i-j)$ is $\geqq \alpha(n)/n$. By (**), $\alpha(j-i)>|X'|$ and $\alpha(n+i-j)>|X''|$; by (*), it follows that $\alpha(j-i)+\alpha(n+i-j)>|X'|+|X''|=|X|=\alpha(n)$. Suppose, for the sake of contradiction, that $\alpha(j-1)/(j-i)<(\alpha(n)/n$ and $\alpha(n+i-j)/(n+i-j)<(\alpha(n)/n$. Then $\alpha(j-i)<(j-i)\alpha(n)/n$ and $\alpha(n+i-j)<(n+i-j)\alpha(n)/n$. But this implies that $\alpha(j-i)+\alpha(n+i-j)<\alpha(n)$, completing the proof.

It remains to prove (*) and (**). Let $\gamma': Z_{j-i} \to Z_n$ be defined by $\gamma'(x)=(x+i)_n$ and let $\gamma'': Z_{n+i-j} \to Z_n$ be defined by $\gamma''(x)=(x+j)_n$. If $x \in X'$, then $a_{i+x}=1$ which means that $(i+x)_n \in X$. Thus, $\gamma': X' \to X$; similarly $\gamma'': X'' \to X$. We claim that X is the disjoint union of $\gamma'(X')$ and $\gamma''(X'')$. Suppose first that $x \in X'$, $y \in X''$ and $\gamma'(x)=\gamma''(y)$. Then, $(i+x)_n=(j+y)_n$ from which it follows easily that $i+x=j+y$ or $i+x=j+y-n$. Now $x<j-i$ and $y<n+i-j$. Thus, $i+x=j+y$ implies that $j-i=x-y \leqq x<j-i$, and $i+x=j+y-n$ implies that $j-i=x-y+n \geqq n-y>n-(n+i-j)=j-i$. To complete the proof of the claim and, with it, the proof of (*), we need only show that $X \subset \gamma'(X') \cup \gamma''(X'')$.

Let $x \in X$ so that $a_x=1$. If $i \leqq x \leqq j-1$ then $x-i \in Z_{j-1}$ and $a_{i+(x-i)}=a_x=1$ so that $x-i \in X'$. It follows that $x=(x-i)+i \in \gamma'(X')$. A similar argument shows that if $0 \leqq x \leqq i-1$ or $j \leqq x \leqq n-1$, then $(x-j)_n \in X''$ and $x=\gamma''((x-j)_n)$.

For the proof of (**), let $x, y \in X'$ with $0 \leqq x<y \leqq j-i-1$. We must show that $(y-x)_{j-i}$ and $(x-y)_{j-i}$ are not in T. Clearly, $(y-x)_{j-i}=y-x=(y+i)-(x+i)=\gamma'(y)-\gamma'(x) \notin T$, since $\gamma'(y)$ and $\gamma'(x)$ are elements of X. For $(x-y)_{j-i}$, note first that if $x \geqq m$ then $(x-y)_{j-i}=x-y+j-i \geqq m-y+j-i \geqq m+(-j+i+1)+j-i=m+1$ so we need only worry about the case where $x \leqq m-1$. Since $x \in X'$, $a_{i+x}=1$ and since $S_i=S_j$ and $x \leqq m-1$, it follows that $a_{j+x}=1$ as well. Thus, $(j+x)_n \notin X$. Now $y \in X'$ implies that $(i+y)_n \in X$ and, hence, $(x-y)_{j-1}=x-y+j-i=(x-y+j-i)_n=[(j+x)_n-(i+y)_n]_n \notin T$. This proves that X' is independent in $G(j-i, T)$; the proof that X'' is independent in $G(n+i-j, T)$ is entirely analogous.

Case 2. $[(i-j)_n \leqq m$ or $(j-i)_n \leqq m]$. We consider only the case where $(j-i)_n<m$, the other case being virtually identical. Next, note that if $X=\{x_1, \cdots, x_r\}$ is independent in $G(n, T)$, then $Y=\{(x_1-i)_n, \cdots, (x_r-i)_n\}$ is independent as well. Thus, we may assume, without loss of generality, that $i=0$. We proceed to define X' and X'' exactly as in Case 1; the proofs of (*), (**), and the fact that $\alpha(j-i)/(j-i)$ or $\alpha(n+i-j)/(n+i-j)$ is $>\alpha(n)/n$ go through unscathed. The only problem is in the event that it is $\alpha(j-i)/(j-i)$ which is $\geqq \alpha(n)/n$, the conclusions of the lemma are not met since $j-i \leqq m$. Suppose then that $S_j=S_0$, $1 \leqq j \leqq m$, and $\alpha(j)/j \geqq \alpha(n)/n$. Then there is an integer $k>1$ such that $m+1 \leqq kj \leqq 2^m$. It suffices, therefore, to show that for some such k, $\alpha(kj) \geqq k\alpha(j)$.

(The situation is similar to that of Sublemma 3.2. However, the conditions of 3.2 are not met here and we must make use of the fact that $S_0=S_j$ to show that the conclusion of Sublemma 3.2 nevertheless holds.)

Since $j \leqq m$, we may write $m=\lambda j+\mu$ with $\lambda \geqq 1$ and $0 \leqq \mu \geqq j-1$. Now $(a_0, \cdots, a_{m-1})=S_0=S_j=(a_j, \cdots, a_{j+m-1})$. It follows that $a_i=a_{i+pj}$ for all i, j with $0 \leqq i \leqq j-1$ and $1 \leqq p \leqq \lambda$. Also, $(\lambda+1)_j \geqq m+1$ so that $\{a_0, a_1, \cdots, a_{j-1+\lambda j}\}$ is a

sequence of length $\geqq m+1$. Let $W \subset Z_{(\lambda+1)j}$ be defined by $x \in W$ if and only if $a_x = 1$. Then $|W| = (\lambda+1)\alpha(j)$ and it is a straightforward exercise to show that W is independent in $G((\lambda+1)j, T)$. Q.E.D.

THEOREM 3.5. *Let T be a finite set of positive integers and let m be the largest element in T. Then*

$$\operatorname{rt}(T) = \sup_{m+1 \leqq n \leqq 2^m} \frac{\alpha(n)}{n}.$$

Proof. Let M be this supremum. By Lemma 3.3, $\operatorname{rt}(T) \geqq M$. Since

$$\operatorname{rt}(T) = \lim_{n\to\infty} \frac{\alpha(n)}{n} = \lim_{\substack{n\to\infty \\ n \geqq m+1}} \frac{\alpha(n)}{n}$$

it follows at once from Lemma 3.4 that $\operatorname{rt}(T) \leqq M$. Q.E.D.

THEOREM 3.6. *For any finite T, $\operatorname{rt}(T)$ is a rational number with $0 < \operatorname{rt}(T) \leqq \frac{1}{2}$.*

Proof. The fact that $\operatorname{rt}(T)$ is rational follows at once from Theorem 3.5. If m is the largest element of T, then $\alpha(m+1) \geqq 1$ and it follows from Lemma 3.3 that $\operatorname{rt}(T) \geqq \alpha(m+1)/(m+1) \geqq 1/(m+1) > 0$. It remains to be shown that $\operatorname{rt}(T) \leqq 1/2$ for nonempty T. Since $T \supset T'$ clearly implies that $\operatorname{rt}(T') \geqq \operatorname{rt}(T)$, it suffices to show this for T a singleton, $T = \{m\}$. By Theorem 3.5, there is an s, $m+1 \leqq s \leqq 2^m$ such that $\operatorname{rt}(T) = \alpha(s)/s$. Let $X = \{x_1, \cdots, x_r\}$ be a maximal independent set in $G(s, T)$. Then $Y = \{(x_1+m)_s, \cdots, (x_r+m)_s\} \cap X = \varnothing$ and $|X| = |Y|$. Thus $2|X| \leqq |V(G(s,t))| = s$. Q.E.D.

Our next aim is to reduce the upper bound, 2^m, in Theorem 3.5.

DEFINITION 3.7. A binary k-tuple $S = (s_0, \cdots, s_{k-1})$ is *T-admissible* if $s_i = s_j = 1$ with $i < j$ implies that $j - i \notin T$. In other words, s is T-admissible if now two of its 1's are separated by a distance in T.

Notation 3.8. We denote by $N_k(T)$ the number of T-admissible k-tuples.

PROPOSITION 3.9.

$$\operatorname{rt}(T) = \sup_{m+1 \leqq n \leqq N_m(T)} \frac{\alpha(n)}{n}.$$

Proof. The proof is completely analogous to that of Theorem 3.5.

We now proceed to show that the rate of T may be determined by examining the cycles in a certain subgraph of the binary de Bruijn graph of span $m+1$. We caution the reader that there is no real connection between this work and the theory of de Bruijn sequences. The de Bruijn graph simply provides a convenient way of describing the technique.

DEFINITION 3.10. (See e.g. [3].) The *binary de Bruijn graph* of span n, denoted P_n, is given as follows:

(a) The vertices of P_n, $V(P_n)$, are the binary n-tuples.

(b) There is a directed edge originating at $V = (v_0, \cdots, v_{n-1})$ and terminating at $W = (w_0, \cdots, w_{n-1})$ if and only if $(v_0, \cdots, v_{n-2}) = (w_1, \cdots, w_{n-1})$.

DEFINITION 3.11. Let T be a set of positive integers with largest element m. $P(T)$ is the vertex subgraph of the de Bruijn graph P_{m+1} determined by the T-admissible $(m+1)$-tuples. That is, $V(P(T))$ is the set of T-admissible $(m+1)$-tuples and there is an edge from $v \in V(P(T))$ to $w \in V(P(T))$ if and only if there is an edge from v to w in P_{m+1}.

DEFINITION 3.12. Let C be a cycle in P_n. The *weight* of C, $\operatorname{wt}(C)$, is the average of the weights of the vertices in C.

PROPOSITION 3.13.

$$\mathrm{rt}(T) = \frac{1}{m+1} \max_{C \text{ a cycle in } P(T)} \mathrm{wt}(C).$$

Proof. This is essentially a graphical reformulation of Proposition 3.9. We leave details to the reader.

Our algorithm for T-coloring K_n is buried in the preceding definitions and theorems and it is appropriate at this point to describe it more explicitly:

Step 1. *The construction of* $P(T)$. Given a set T, we denote by m the largest element of T and by P_{m+1} the binary de Bruijn graph of span $m+1$. We call a vertex of P_{m+1} T-admissible if no two of its 1's are separated by a distance in T. Finally, $P(T)$ is the vertex subgraph of P_{m+1} generated by the T-admissible vertices.

Step 2. *The selection of an optimal cycle of* $P(T)$. The weight of a vertex of $P(T)$ is the number of 1's it contains. The weight of a cycle of $P(T)$ is the average of the weights of the vertices comprising it. We choose a cycle C in $P(T)$ of maximum weight. (This is the difficult part of the algorithm.)

Step 3. *The construction of the coloring sequence.* Let $C = (V^1, \cdots, V^k)$ and let v_i be the leftmost bit of V^i. Next, let $\{a_i\}$ be the sequence obtained by repeating $v_1, v_2, \cdots, v_k$ indefinitely. Finally, define a sequence of integers $B = \{b_j\}$ by letting b_j be an element of the sequence if and only if $a_{b_j} = 1$.

Step 4. *The t-coloring of* K_n. To T-color K_n use $b_1, b_2, \cdots, b_n$.

Although Step 2 is, in general, exponentially hard, there are many cases where it is, in fact, quite easy. Corollaries 3.14 through 3.16 give several such examples.

COROLLARY 3.14. *If* $\alpha(m+1, T) = \beta(m+1, T)$ *then* $\mathrm{rt}(T) = \alpha(m+1)/(m+1)$.

Proof. Since $\mathrm{rt}(T)$ equals $1/(m+1)$ times the maximal cycle weight and the weight of a cycle is the average of the weights of its vertices, it follows that $\mathrm{rt}(T) \leqq \beta(m+1, T)/(m+1)$. (Note that $\beta(m+1, T)$ is the maximum possible weight T-admissible $(m+1)$-tuple.) On the other hand, if $\alpha(m+1, T) = \beta(m+1, T)$ then there is a maximum weight vertex in $P(T)$ all of whose cyclic shifts are also in $P(T)$. Thus, there is an entire cycle in $P(T)$ consiting purely of maximum weight vertices. Q.E.D.

COROLLARY 3.15. *If* $\beta(m+1, T) = 1$ *then* $\mathrm{rt}(T) = 1/(m+1)$.

COROLLARY 3.16. *If* $\beta(m+1, T) = 2$ *and there exists a* k, $1 \leqq k \leqq m$ *such that neither* k *nor* $m+1-k$ *is an element of* T, *then* $\mathrm{rt}(T) = 2/(m+1)$.

The proofs of Corollaries 3.15 and 3.16 follow easily from Corollary 3.14; details are left to the reader.

Example 3.17. (a) Suppose $T = \{1, 4, 7\}$. Then it is an easy exercise to see that $\beta(8, T) = \alpha(8, T) = 3$. It follows from Corollary 3.15 that $\mathrm{rt}(T) = 3/8$. To T-color K_n, we use the first n elements of the sequence $(1, 3, 6, 9, 11, 14, 17, 19, 22, 25, 27 \cdots)$.

(b) Suppose $T = \{1, \cdots, r\}$. Then $\beta(r+1, T) = 1$. It follows from Corollary 3.16 that $\mathrm{rt}(T) = 1/(r+1)$. The sequence is $(1, r+2, 2r+3, 3r+4, \cdots)$. This is, in essence, the result in [2] on r-initial sets.

(c) Suppose $T = \{1, 4, 5\}$. Then $\beta(6, T) = 2$ and there is an integer, namely 3, such that 3 and $m+1-3 = 6-3 = 3$ are not in T. It follows from (3.17) that $\mathrm{rt}(T) = 2/6 = 1/3$. The sequence to be used is $(1, 4, 7, 10, 13, 16, \cdots)$.

We close this section by noting that the techniques developed here can be used to obtain precise bounds on the error involved in approximating $\mathrm{sp}_T(K_n)$ by $n/\mathrm{rt}(T)$. Moreover, Corollary 3.15 remains true if $m+1$ is replaced by any integer $n > m+1$. These results will be dealt with in a future paper.

4. Special classes of T-sets. The structure of the graphs $G(n, T)$, $H(n, T)$, and $G(T)$ reflects the structure of the set T. For example, if the elements of T have a

greatest common divisor g, then the graph $G(T)$ splits into g disjoint isomorphic components. Also, when T is a 2-set or a 3-set of the form a, b, $a+b$, then the graph $G(T)$ can be embedded on a surface of an infinite cylinder. These and similar observations lead to a number of results concerning rt (T) for special classes of T-sets.

THEOREM 4.1. *If g is the greatest common divisor of the elements of T, then the graph $G(T)$ is the disjoint union of g mutually isomorphic components $G_i(T)$, $0 \leqq i \leqq g-1$. $G_i(T)$ is defined as follows*:

$$V(G_i(T)) = \left\{ x \,\middle|\, x = i + \sum_j a_j t_j,\ t_j \in T \right\};$$

$E(G_i(T))$ is the restriction of $E(G(T))$ to $V(G_i(T))$.

Furthermore, rt (T) = rt (T/g) *where* $T/g = \{x | x = t/g,\ t \in T\}$.

Proof. Let $x_p \in V(G_p(T))$ and $x_q \in V(G_q(T))$, where $p \neq q$, $0 \leqq p, q \leqq g-1$. Then $(x_p - x_q)_g = ((p-q) + (\sum_j a_j t_j - \sum_k b_k t_k))_g = (p-q)_g \neq 0$ and so $(x_p - x_q) \notin E(G(T))$. This shows that the graphs $G_i(T)$ are disjoint. The isomorphism between the components $G_p(T)$ and $G_q(T)$ is defined by mapping $x_p = p + \sum_j a_j t_j \in G_p(T)$ to $x_q = q + \sum_j a_j t_j \in G_q(T)$. To show that rt (T) = rt (T/g), observe that the graph $H(gn, T)$ consists of g isomorphic components $H_i(gn, T) \simeq H(n, T/g)$. The isomorphism maps $x_i = i + \sum_j a_j t_j \in H_i(gn, T)$ to $x_0/g = \sum_j a_j (t_j/g)$. Thus $\beta(gn, T) = g \cdot \beta(n, T/g)$ and so rt (T) = rt (T/g). Q.E.D.

PROPOSITION 4.2. rt $(T) \geqq 1/s$, *where s is the smallest integer such that $s \not\equiv t$ for all $t \in T$.*

Proof. The set $M_n = \{ks | k = 0, 1, \cdots, [n/s]\}$ is an independent set in $H(n, T)$, and so

$$\text{rt}\,(T) = \lim_{n\to\infty} \frac{\beta(n, T)}{n} \geqq \lim_{n\to\infty} \frac{[n/s]+1}{n} \geqq \frac{1}{s}. \qquad \text{Q.E.D.}$$

COROLLARY 4.3. rt $(T) \geqq 1/(m+1)$, *where* $m = \max\{t | t \in T\}$.

PROPOSITION 4.4. *Let* $T = \{k, \cdots, r\}$. *Then* rt $(T) = k/(k+r)$.

Proof. It can be easily shown that in any set of $k+r$ consecutive vertices of $H(n)$ at most k can belong to any maximum independent set of $H(n)$ and so $\beta(nk+nr, T) \leqq nk$. But since the set $M = \{x | x = p + a(k+r);\ p = 0, 1, \cdots, k-1;\ a = 0, 1, \cdots, n-1\}$ is an independent set in $H(n)$, we conclude that $\beta(nk+nr, T) = nk$. Thus, rt $(T) = \lim_{n\to\infty} \beta(nk+nr)/(nk+nr) = k/(k+r)$. Q.E.D.

COROLLARY 4.5. *Let* $\min \leqq t \leqq \max$ *for* $t \in T$. *Then* rt $(T) \geqq \min/(\min+\max)$.

Proof. This follows immediately from the fact that the set T is contained in the set $T' = \{min, min+1, \cdots, \max\}$.

LEMMA 4.6. *Let* $T = \{a, b\}$ *with* $(a, b) = 1$. *Then the graph $H(a+b, T)$ is a simple cycle of length $a+b$.*

Proof. Since $(a, a+b) = (a, b) = 1$, we may write the vertex set of $H(a+b, T)$ as $V(H(a+b, T)) = \{v_k | v_k = ka \bmod (a+b);\ k = 0, 1, \cdots, a+b-1\}$. It can easily be shown that $E(H(a+b, T)) = \{(v_k, v_l) | l = (k+1) \bmod (a+b)\}$, which completes the proof.

LEMMA 4.7. *Let* $T = \{a, b\}$, $(a, b) = 1$. *Then the graph $H(n(a+b), T)$ has a spanning subgraph consisting of n disjoint cycles C_i of length $a+b$ defined by*

$$C_i = \{v_k^i | v_k^i = v_k + i(a+b),\ v_k \in V(H(a+b, T))\}.$$

Proof. Obviously, each of the cycles C_i is a translation of the graph $H(a+b, T)$ by $i(a+b)$. We have to show that these cycles are disjoint. But this is clearly so, since the cycle C_i consists of vertices $i(a+b)$, $i(a+b)+1, \cdots, i(a+b)+(a+b-1)$.

THEOREM 4.8. *Let* $T=\{a, b\}$, $(a, b)=1$. *Then* $\operatorname{rt}(T)=h(a+b)/(a+b)$ *where* $h(x)=$ *integer part of* $x/2=[x/2]$.

Proof. Each cycle C_i of the graph $H(n(a+b), T)$ can contain at most $h(a+b)$ independent vertices. Therefore, $\beta(n(a+b), T) \leqq n \cdot h(a+b)$. Let

$$M_0=\{v_k^0 | v_k \in C_0, k=2q, q=0, 1, \cdots, h(a+b)-1\},$$

$$M_i=\{v_k^i | v_k^i=v_k+i(a+b), v_k \in M_0\},$$

and finally,

$$M(n)=\bigcup_{i=0}^{n-1} M_i.$$

Clearly, each set M_i is a maximum independent set in the cycle C_i. If $M(n)$ is an independent set in $H(n(a+b), T)$, then $\beta(n(a+b), T)=n \cdot h(a+b)$ for all n, and so

$$\operatorname{rt}(T)=\lim_{n\to\infty} \beta(na+nb, T)/(na+nb)=h(a+b)/(a+b).$$

To show that $M(n)$ is an independent set, we first define

$$\Gamma^+(v_k^i)=\{v_k^i+a, v_k^i+b\},$$

$$\Gamma^-(v_k^i)=\{v_k^i-a, v_k^i-b\}.$$

Then $\Gamma^+(v_k^i)=\Gamma^-(v_k^{i+1})$ because

$$v_k^i+a=v_k+i(a+b)+a=v_k+(i+1)(a+b)-b=v_k^{i+1}-b,$$

$$v_k^i+b=v_k+i(a+b)+b=v_k+(i+1)(a+b)-a=v_k^{i+1}-a.$$

Moreover, the neighbors of v_k^i are precisely the four vertices in

$$\Gamma^+(v_k^i) \cup \Gamma^-(v_k^i).$$

Now two of these vertices are neighbors of v_k^i in the cycle C_i and so cannot be in $M(n)$. The other two neighbors belong to the cycles C_{i-1} and C_{i+1}. However, they too cannot belong to $M(n)$ because they are neighbors of vertices in the cycles C_{i-1} and C_{i+1}. Q.E.D.

It is worth noting that the greedy algorithm (see [2]) does not provide minimal span colorings even in this sample case. For example, if $T=\{7, 10\}$, then the greedy algorithm chooses seven out of each seventeen vertices to belong to the independent set, while Theorem 4.8 shows that eight is the maximum possible.

5. The rate of T and the clique size of $G(T)$. In this section, we consider the relationship between the maximum clique size in the graph $G(T)$ and the rate of T. In particular, we show that $\operatorname{rt}(T) \leqq 1/\operatorname{clq}(G(T))$. The proof is based on a suggestion of Mark Ramras. Finally, we give a lower bound for 3-sets T with $\operatorname{clq}(G(T))=3$ and conjecture that this lower bound is actually equal to the rate.

DEFINITION 5.1. A clique of size n in a graph G is a subgraph of G with n vertices that is isomorphic to a complete graph K_n. The size of the maximum clique of the graph G will be denoted by $\operatorname{clq}(G)$.

THEOREM 5.2. $\operatorname{rt}(T) \leqq 1/\operatorname{clq}(G(T))$.

Proof. Let $c=\operatorname{clq}(G(T))$, and let $C=\{v_i | i=1, 2, \cdots, c;\ v_1<v_2<\cdots<v_c\}$ be a clique in $G(T)$ so that $v_j-v_i \in T$ for all $i<j$. Choose $n>v_c$ and let M be a maximum independent set in $G(n, T)$. For $i=1, 2, \cdots, c$, define $M_i=\{u+v_i-v_1)_n | u \in M\}$ to be a translation of the set M by (v_i-v_1). Clearly, $M=M_1$ and $M_i \cap M_j=\varnothing$ for all pairs

$i \neq j$. But then

$$\bigcup_{i=1}^{c} M_i \subset (V(G(n, T))$$

and so $\alpha(n, T)/n \leqq \alpha(n, T)/c\alpha(n, T) = 1/c = 1/\text{clq}(G(T))$. But this implies

$$\text{rt}(T) = \lim_{n\to\infty} (\alpha(n, T)/n) \leqq 1/\text{clq}(G(T)). \qquad \text{Q.E.D.}$$

Suppose now that T is a 3-set with $\text{clq}(G(T)) = 3$, that is, $T = \{a, b, a+b\}$, $a \neq b$. If $g = \gcd(a, b)$, then by Theorem 4.1, $\text{rt}(T) = \text{rt}(T/g)$, so assume that a and b are relatively prime. By Theorem 5.2, $\text{rt}(T) \leqq \frac{1}{3}$. If 3 is relatively prime to all elements of T, then Proposition 4.2 implies that $\text{rt}(T) \geqq \frac{1}{3}$ and, hence, that $\text{rt}(T) = \frac{1}{3}$. The remainder of the paper deals with the determination of $\text{rt}(T)$ in the case where 3 divides one of the elements of T.

PROPOSITION 5.3. *Let $T = \{a, b, c\}$ be a 3-set such that*

(i) $a + b = c$;

(ii) *a, b, c are pairwise relatively prime*;

(iii) $3|a$ or $3|b$ *or* $3|c$;

(iv) $a < b$.

Let $p = a + c$ or $p = b + c$ and let $t(p) =$ "third of p" $= [p/3]$. Define $S_i = \{i(b-a) + kp | k = 0, \pm 1, \pm 2, \cdots\}$ and set

$$M_p = \bigcup_{i=0}^{t(p)-1} S_i.$$

Then M_p is an independent set in the graph $G(T)$.

Proof. We show first that each set S_i is an independent set in $G(T)$. Let $u, v \in S_i$, $u \neq v$. If $p = a + c$, then $|u - v| \geqq p = a + c = 2a + b$, and if $p = b + c$, then $|u - v| \geqq p = b + c = a + 2b$. Clearly, then S_i is independent.

To complete the proof, it suffices to show that $S_i \cap (S_j + \alpha a + \beta b) = \varnothing$ for all $\alpha, \beta \in \{0, 1\}$ and all $i \neq j$. We consider only the case where $p = a + c = 2a + b$; the other case is entirely analogous. Suppose $S_i \cap (S_j + \alpha a + \beta b) \neq \varnothing$. Then there exists i, j, k, l with $0 \leqq i, j \leqq t(p) - 1$ such that

$$i(b-a) + k(2a+b) = j(b-a) + l(2a+b) + \alpha a + \beta b.$$

This implies that

$$a(-i + 2k + j - 2l - \alpha) = b(j + l - i - k + \beta).$$

Since $(a, b) = 1$, it follows that

$$-i + 2k + j - 2l - \alpha = \lambda b, \tag{5.1}$$

$$j + l - i - k + \beta = \lambda a, \tag{5.2}$$

for some integer λ.

Equations (5.1) and (5.2) lead to

$$3(k - l) - \alpha - \beta = \lambda(b - a) \tag{5.3}$$

(5.2) implies

$$j - i = \lambda a + (k - l) - \beta. \tag{5.4}$$

We proceed to multiply (5.4) by 3 and sustitute for $3(k - l)$ from (5.3) to conclude that

$$3(j - i) = \lambda(2a + b) + \alpha - 2\beta. \tag{5.5}$$

We know that $1-t(p) \leqq j-i \leqq t(p)-1$, and so $3-3t(p) \leqq 3(j-i) \leqq 3t(p)-3$. Since $3t(p) \leqq p$, it follows that

$$3-p \leqq \lambda p+\alpha-2\beta \leqq p-3. \tag{5.6}$$

We observe that in (5.5), the left-hand side is divisible by three and is nonzero. On the right-hand side, the only values possible for $\alpha-2\beta$ are $1, 0, -1, -2$, and therefore $\lambda \neq 0$. But that with (5.6) leads to a contradiction and so S_i, S_j are disjoint and their union forms an independent set. Q.E.D.

THEOREM 5.4. *Let* $T=\{a, b, c\}$ *be such that*

(i) $a+b=c$;

(ii) a, b, c *are pairwise relatively prime*;

(iii) $3|a$ *or* $3|b$ *or* $3|c$;

(iv) $a<b$.

Then

$$\operatorname{rt}(T) \geqq \max\left(\frac{t(b+c)}{b+c}, \frac{t(a+c)}{a+c}\right).$$

Proof. The set

$$M_p^n = \bigcup_{i=0}^{t(p)-1} S_i^n \quad \text{where } S_i^n = S_i|_{H(n,T)}$$

and $n=mp$ for some m is an independent set in $H(n, T)$. Therefore, $\beta(n, T)/n \geqq m \cdot t(p)/mp = t(p)/p$ and so $\operatorname{rt}(T) \geqq t(p)/p$. Q.E.D.

CONJECTURE 5.5. *In Theorem* 5.4, *equality holds.*

Acknowledgments. The authors are indebted to the referee for several most useful suggestions for simplifying a number of theorems in this paper.

REFERENCES

[1] B. ALSPACH AND T. D. PARSONS, *Isomorphism of circulant graphs and digraphs*, Discrete Math., 25 (1979), pp. 97-108.

[2] M. B. COZZENS AND F. S. ROBERTS, *T-colorings of complete graphs and the channel assignment problem*, Congressus Numerantium, 35 (1982), pp. 191-208.

[3] S. W. GOLOMB, *Shift Register Sequences*, Aegean Park Press, Laguna Hills, CA, 1982.

[4] W. K. HALE, *Frequency assignment: theory and applications*, Proceedings of the IEEE, 68 (1980), pp. 1497-1514.

[5] T. D. PARSONS, *Circulant graph embeddings*, J. Comb. Theory, Ser. B., 29 (1980), pp. 310-320.

[6] G. PÓLYA AND G. SZEGÖ, *Aufgaben und Lehrsätze aus der Analysis* I, Springer-Verlag, Berlin, Heidelberg, New York, 1970.

[7] M. RAMRAS, private communication.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 3, July 1985

ON GRACEFUL DIRECTED GRAPHS*

G. S. BLOOM† AND D. F. HSU‡

Abstract. A digraph D with e arcs is *numbered* by assigning a distinct integer value $\theta(v)$ from $\{0, 1, \cdots, e\}$ to each node v. The node values, in turn, induce a value $\theta(u, v)$ on each arc (u, v) where $\theta(u, v) = \theta(v) - \theta(u) \bmod (e+1)$. If the arc values are all distinct and nonzero, then the numbering is *graceful*. A digraph is graceful if it has a graceful numbering.

Graceful digraphs are related in a variety of ways to other areas of mathematics. It is shown here that the graceful numberings of certain classes of digraphs are characterized by the existence of particular algebraic structures, including cyclic difference sets, sequenceable groups, $(K, 1)$ complete mappings, and $(K, 1)$ near-complete mappings. Some digraph models of cyclic groups are examined here, and cyclic neofields are shown to generate families of graceful numberings for the models.

Properties and examples of this new class of graph numberings are presented here. For instance, families of graceful digraphs include certain orientations of cycles, paths, and the unions of cycles and paths, as well as certain complete digraphs, wheels, windmills, and umbrellas. Techniques are developed by which digraphs can be embedded as subgraphs of infinite families of graceful digraphs.

A variety of fundamental questions are posed and some conjectures advanced.

AMS(MOS) subject classifications. 05B10, 05B99, 05C20, 17D99, 20F99

1. Preliminaries and elementary observations. This paper is intended to introduce some of the theoretical and applied aspects of graceful directed graphs to a variety of specialists working in areas of algebra, graph theory, combinatorics, and number theory. In consideration of this, more exposition and examples are presented than is customary for a narrower technical audience.

In recent years considerable interest has developed in studying *graph numberings*. For such numberings the nodes of an undirected graph are assigned values, which in their turn induce values upon each edge as a function of the two values on the endnodes of the edge. A wide variety of these numberings has been studied both for their intrinsic mathematical interest and for their utility to an expanding range of applied fields. Some indication of the variety of uses to which these graphs have been put can be found in [B1], [BG1] and [BG2] which cite applications to radar pulse codes, x-ray crystallography, circuit layout design, missile guidance, and numerical analysis, among others. In addition, applications have been studied for communications loop addressing [GP], radio astronomy [Be], sonar ranging [GT], coding theory [GS2], and broadcast frequency assignments [CR], [RP] (this issue, pp. 507–518).

An undirected graph with e edges is *gracefully numbered* if each node v is assigned a distinct value $\lambda(v)$ from the set $\{0, 1, \cdots, e\}$ in such a way that the set of edge numbers equals $\{1, \cdots, e\}$ when edge uv is numbered by $\lambda(uv) = |\lambda(u) - \lambda(v)|$. A graph is said to be a *graceful (undirected) graph* if it can be gracefully numbered. Examples of undirected graphs are shown in Fig. 1. The problem of characterizing graceful graphs was introduced in [Ro] and [Go] and remains unsolved. Indeed, despite considerable effort, it is still unknown whether all trees are graceful [B2]. Recently, "harmonious" and "elegant" numberings of undirected graphs have been studied. In these an edge is numbered with the modular sum of the values on its endnodes, see [GS1], [GS2], [CHR], and [H2].

---

* Received by the editors January 30, 1984, and in final revised form December 15, 1984. This research was conducted under the partial support of PSC-CUNY grant 6-63230 and Fordham University Faculty Research grant 071225. This work was presented at the SIAM Second Conference on the Applications of Discrete Mathematics, held at Massachusetts Institute of Technology, Cambridge, Massachusetts, June 27–29, 1983.

† Computer Science Department, The City College of New York, New York, New York 10031.

‡ Department of Computer and Information Science, Fordham University, Bronx, New York 10458.

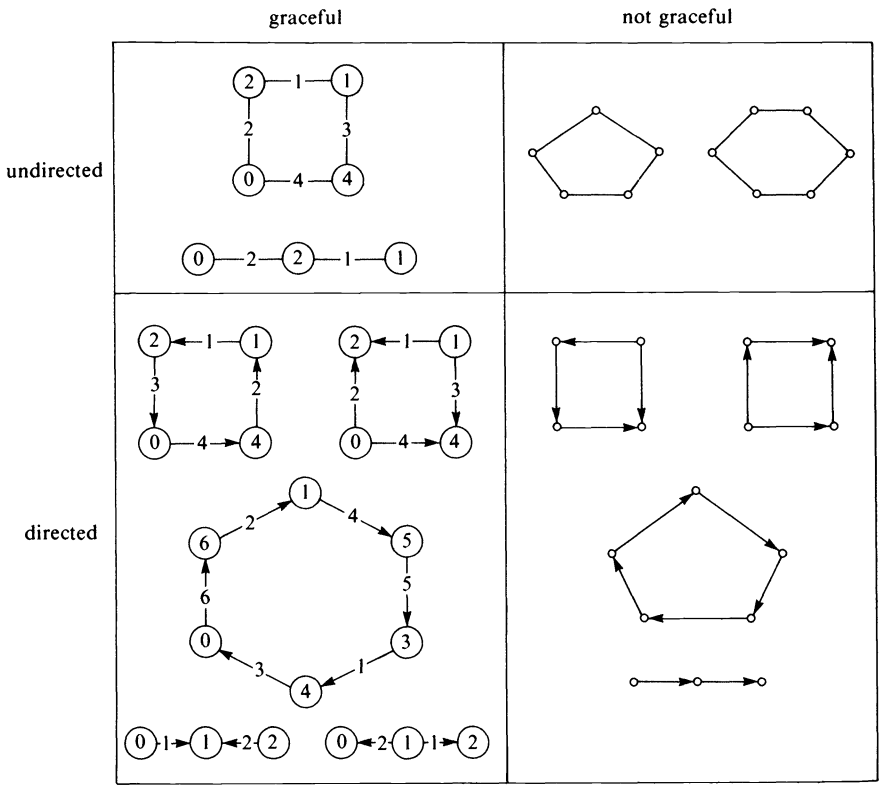


FIG. 1. *A selection of graphs and digraphs classified as graceful or not.*

One can obtain a graceful numbering for directed graphs analogous to that for undirected graphs by defining an arc value as the simple difference of the values on its endpoints, and by requiring that arc numbers be limited in value to the range of the node numbers by using modular arithmetic. Using modular arithmetic also ties graceful digraphs to a variety of algebraic problems of long standing, as will be seen later in this paper. A "real world" application of graceful digraphs to a communication network addressing problem is considered in [BH1].

A directed graph D with n nodes and e arcs, no self-loops, and no more than one arc directed from any one node x to any other node y, is numbered by assigning to each node a distinct element from the set $Z_{e+1} = \{0, 1, \cdots, e\}$. An arc (x, y) from node x to node y is numbered with $\theta(x, y) = \theta(y) - \theta(x) \bmod (e+1)$, where $\theta(y)$ and $\theta(x)$ are the values assigned to y and x. A numbering is a *graceful numbering* if all $\theta(x, y)$ are distinct. If a digraph D admits a graceful numbering, then D is a *graceful digraph.* Figure 1 shows examples of graceful numberings of several digraphs.

Often it is convenient to consider graphs associated with a particular digraph D. An undirected (simple) graph $G = |D|$ is said to be the *underlying graph* for digraph D, if G has the node set of D and includes the edge xy if either (x, y) or (y, x) or both belong to D. Because each edge of $|D|$ can be oriented in one or both of two directions in D, there are, at most, 3^e digraphs associated with each underlying graph $|D|$ on e arcs. Each of these digraphs is an *orientation* of $|D|$. If no pair of edges (x, y) and (y, x) both belong to D, then D is a *simple orientation* of $|D|$.

Given a gracefully numbered undirected graph G with node numbering $\lambda(x)$ for node x, it is trivial to assign a simple orientation that produces a graceful digraph D

with G as its underlying graph. Merely orienting each edge of D to point toward the larger node value accomplishes this.

Although a graceful graph always gives rise to a graceful digraph, an ungraceful graph may underlie a graceful directed one; moreover, not all orientations of an undirected graph are graceful, regardless of whether the underlying graph is graceful or not. Figure 1 demonstrates the possibilities.

A graceful orientation of C_4 is shown in Fig. 1, as is the gracefully numbered $\vec{C}_4$ trivially obtained from the illustrated graceful numbering of C_4. Nevertheless, the two other simple orientations of C_4 cannot be gracefully numbered. C_5 is not graceful and neither is the unidirectional $\vec{C}_5$. C_6 is not graceful, but its unidirectional orientation is. Even the gracefully numbered two-edge tree has both graceful and nongraceful simple orientations.

An undirected graph is termed *digraceful* if some orientation of its edges produces a graceful directed graph. It was previously seen that every graceful graph and some nongraceful ones are digraceful by the "trivial orientation." Nevertheless, not every graph is digraceful, i.e. has an orientation of its edges that yields a graceful digraph. A class of nondigraceful graphs as noted in [De] is specified in Proposition 1.1. This class includes C_5; so it is not only the unidirectional orientation already discussed, but every orientation of the edges, that is not graceful.

PROPOSITION 1.1. *A graph with e arcs having even degrees at each node and* $e \equiv 1$ *modulo* 4 *is not digraceful.*

Proof. The sum of the arc values is $1+2+\cdots+(4a+1)$ which is odd. On the other hand, this sum is also equal to $\sum_v [\deg_{\text{in}}(v) - \deg_{\text{out}}(v)]\theta(v)$ which is even. Although arithmetic for these cases is being done modulo $(4a+2)$, "odd" and "even" retain their meaning, and the impossibility of obtaining a numbering is established. □

In addition to the gracefully numbered simple orientations of graphs, some digraphs have arcs both from u to v and from v to u. Among these there is another set of digraphs that are immediately gracefully numbered from a graceful numbering of their underlying graphs. A *symmetric digraph* $\vec{G}$ *based on* (*underlying*) *graph* G has the same node set as G, but has arcs (x, y) and (y, x) replacing each edge xy of G. It is easy to show the following result:

PROPOSITION 1.2. *If gracefully labelled graph G has e edges, then* $\vec{G}$ *is graceful with the same node labels.*

A graceful digraph D does not have a unique graceful numbering, since adding a constant modulo $(e+1)$ to all of the node numbers of a digraph preserves the arc numbers and therefore generates a new graceful numbering of D. Graceful numberings of the nodes of D, $\theta_1(V(D))$ and $\theta_2(V(D))$ are termed *equivalent* if $\theta_1(V(D)) \equiv \theta_2(V(D)) + k \pmod{e+1}$. A set of $(e+1)$ equivalent graceful numberings of D is called *complete.* It is easily seen that a complete set of equivalent graceful numberings of C_4 results from adding constants to the sequence of labels $(0, 4, 1, 2)$ or from $(0, 4, 2, 3)$. It is useful to be able to choose a representative numbering from a complete set of equivalent graceful numberings. The *canonical representative graceful numbering* of a complete set of equivalent numberings will be chosen so that the arc labelled e is directed from the node labelled 0 to the node labelled e, i.e., $\theta(0, e) = e$.

A further obvious but useful implication of equivalent graceful numberings is the *rotatability* of node numbers. Any desired node number may be assigned to any desired node of a graceful digraph by adding an appropriate constant. It is also important to realize that not all graceful numberings of a digraph are equivalent, as is illustrated in Fig. 2.

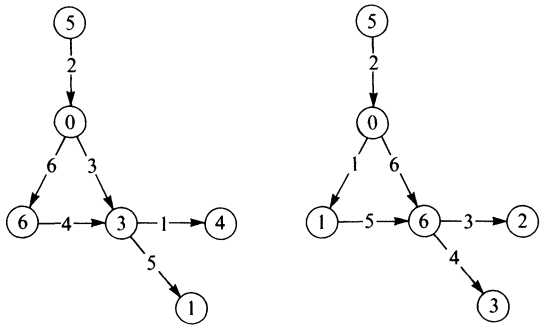


FIG. 2. *Two nonequivalent graceful numberings of a digraph.*

The next two propositions of this section demonstrate how knowing one gracefully numbered digraph is sufficient to determine other distinct graceful digraphs. For example, if D is a digraph, its *corresponding reversed digraph* $(-D)$ can be obtained from D by replacing each arc (u, v) by its reversed arc (v, u). Clearly, if $\theta(V)$ is a graceful node numbering for D, then it is also a graceful node numbering for $(-D)$. These observations are immediate corollaries to Proposition 1.3, for which the following definition is needed. Digraphs D_1 and D_2 with common underlying graph G are said to be *similar* (or *arc-pair similar*) if there is an identical node numbering which is graceful for both.

PROPOSITION 1.3. *Each digraph with e arcs having graceful node numbering $\theta(V)$ is a representative of a family of arc-pair similar digraphs containing no more than $2^{\lfloor e/2 \rfloor}$ distinct graceful digraphs, all having identically numbered nodes in the common underlying graph.*

Proof. For the given graceful digraph D with node numbering $\theta(V)$, pair two arcs (u, v) and (x, y) that are numbered respectively by values k and $-k = e+1-k \bmod (e+1)$. Replace this pair of arcs by (v, u) and (y, x) to form digraph D'. In D', $\theta(v, u) = -k$ and $\theta(y, x) = k$. Thus, the set of arc numbers is unchanged by this exchange. There are at most $\lfloor e/2 \rfloor$ pairs that can be reversed in this way to generate new graphs that preserve arc numbers. Since each arc pair can take one of two orientations, there may be formed from a given node numbering at most $2^{\lfloor e/2 \rfloor}$ distinct members of this family of graceful graphs. This upper bound is realized for digraphs containing an even number of arcs and no symmetric arc pairs (u, v) and (v, u), and having an underlying graph with identity automorphism group (which guarantees that the reversal of any set of arc pairs gives a digraph not isomorphic to any other). □

Note that nonequivalent graceful numberings of a digraph generate distinct families of graceful digraphs. For example, the two numberings of digraph D in Fig. 2 belong to families, say, A and B, of similar digraphs with eight members respectively, since three arc-pairs in each digraph can be reversed. For these two families, it is easy to see that $A \cap B = \{D, (-D)\}$. In Fig. 1 the two graceful orientations of C_4 can be obtained from one another by applying Proposition 1.3, as can the two graceful orientations of the three-point path.

Another transformation of one graceful numbering into another follows.

PROPOSITION 1.4. *If k is relatively prime to $e+1$, and if θ is a graceful numbering of the nodes of a digraph D, then $\theta' = k\theta$ is a graceful numbering of the nodes of digraph D.*

For some D, θ, and k, the graceful numberings θ and θ' are equivalent, and k is called a *multiplier* of θ.

Question 1.5. For what values of k is θ' equivalent to θ for a given digraph D?

In this section, three of four general methods for obtaining graceful digraphs have been introduced: (1) Graceful numberings of undirected graphs. (2) Ad hoc graceful numberings of particular digraphs (or families of digraphs). (3) Modification (or extension) techniques to generate new graceful digraphs from ones already found The fourth general method to obtain graceful digraphs is demonstrated in § 4, where algebraic structures produce graceful numberings for certain digraphs. Alternately, one can say that that graceful digraphs furnish models for algebraic structures.

2. Complete symmetric digraphs. The only graceful complete (undirected) graphs are those with no more than four nodes. Nevertheless, many complete symmetric digraphs are graceful. This is best understood by associating these digraph numberings with a well-studied but incompletely solved problem in combinatorial theory.

PROPOSITION 2.1. *$\vec{K}_n$ has a graceful numbering if and only if there exists a cyclic (v, k, λ)-difference set with $v = n^2 - n + 1$, $k = n$, and $\lambda = 1$.*

Proof. Suppose there exists a graceful numbering $\{a_1, a_2, \cdots, a_n\}$ of $\vec{K}_n$. Since there are $n(n-1)$ arcs in $\vec{K}_n$, the arc numberings are computed modulo $n(n-1)+1 = n^2 - n + 1$. Consider $D = \{a_1, a_2, \cdots, a_n\}$ as a subset of Z_v, $v = n^2 - n + 1$. The gracefulness of this arc numbering implies that, for each nonzero residue x modulo v, there exists a unique pair of subscripts i, j such that $a_i - a_j \equiv x \pmod{v}$. Thus, D satisfies the conditions to be a cyclic $(n^2 - n + 1, n, 1)$-difference set. The converse of the proof is similar, and follows by assigning elements of the difference set to the nodes of $\vec{K}_n$. The condition for D to be a difference set implies the gracefulness of the arc numbers. □

Giving a complete list of the values of n for which $\vec{K}_n$ is graceful is not yet possible. The well-known Singer theorem (see, e.g. [Ba]) asserts that there exists a cyclic $(v, k, 1)$-difference set when $k-1$ is any prime power, but the conjecture remains unresolved that no $(\lambda = 1)$-cyclic difference sets exist for other values of k. (See [Ba] and [St] and their references.) Consequently, despite the conjecture for the necessity of the condition the following is the strongest statement that can currently be made.

PROPOSITION 2.2. *If $n-1$ is a prime power, then $\vec{K}_n$ is graceful.*

Examples of graceful numberings of complete digraphs are listed here: $\vec{K}_4$, $\vec{K}_5$, and $\vec{K}_9$ can respectively be gracefully numbered with $\{1, 2, 4, 10\}$, $\{0, 3, 4, 9, 11\}$, and $\{0, 1, 3, 7, 15, 31, 36, 54, 63\}$. Note that 3 is a multiplier of $\vec{K}_4$, and that a partial answer to Question 1.5 is given by Hall's multiplier theorem (see, e.g. [Ba]) which implies that if θ is a graceful numbering of $\vec{K}_{p^n+1}$, then p is a multiplier for θ.

3. Trees. In Fig. 1 it is shown that one orientation of the tree with three nodes is graceful and that the other two are not. In this section what is known about graceful directed trees is summarized and a conjecture is advanced.

The most studied problem for graceful undirected graphs is to determine if all trees are graceful. A history of this problem is given in [B2]. As was explained there, it would not be difficult to number any tree gracefully, if graceful trees could be renumbered gracefully so that any specified node could be labelled by zero. This would allow an inductive growing of graceful trees, since a new arc and node can be attached to the node labelled zero without disrupting the numbering of any other nodes or arcs. Unfortunately, as was shown in [CH] such rotatable numberings cannot always be accomplished. On the other hand, the graceful numberings of directed graphs are rotatable as explained in § 2. However, adding a new node and a new arc changes the modulus and the way arithmetic is performed in the expanded digraph. Unlike the case for undirected graphs, attaching an arc to any node (even zero) in a directed

graph generally causes arc values to change even if the arc endnodes retain their original values. Thus, for quite different reasons, it is apparently as difficult to grow all graceful directed trees by adding one branch at a time as it is to grow their graceful undirected counterparts.

Beyond the facts that graceful trees trivially give graceful directed trees, and that all trees similar to these are graceful, little is known about general, arbitrarily oriented trees. Only one infinite class of graceful directed trees has been characterized. A directed path is *unidirectional* if all internal nodes have indegree = outdegree = 1.

PROPOSITION 3.1. *The unidirectional path $\vec{P}_n$ on n nodes is graceful if and only if n is even.*

Proposition 3.1 was proved in [BH1] in which the values for consecutive nodes $a_1, a_2, \cdots, a_n$ were given as $\theta(a_1) = (-1)^{i+1}\lfloor i/2 \rfloor$. A nonequivalent graceful numbering of a unidirectional path can also be generated by the process of sequencing the elements of a sequenceable cyclic group.

The procedure for using sequenceable cyclic groups to generate graceful numberings for the unidirectional path can be viewed as constructing a special class of "ruler" using the additive group of integers modulo n. First, segments of the intended ruler are created of lengths $1, \cdots, n-1$, i.e. their lengths are equal to the nonzero elements of Z_n. These segments are then put into a linear sequence to form a ruler of length $\sum_{i=0}^{n-1} i = (n^2 - n)/2$, such that the set of $n-1$ measurements made between one designated end node of the ruler and each of the other $n-1$ ruler marks are all distinct when calculated modulo n. Thus, if the sequence of "segments" is $s_0 = 0, s_1, \cdots, s_{n-1}$ and measurements $d_0, d_1, \cdots, d_{n-1}$ are made from endnode $d_0 = 0$, then Z_n is termed *sequenceable* if $\{s_1\} = \{d_i = \sum_{k=0}^{i} s_k \pmod n\} = Z_n$. This is equivalent to saying that for a sequenceable group, assigning d_i as the node number of the ith node, gives $\{s_i\}$ as the distinct arc numbers and automatically yields a graceful numbering.

The following is an alternate way of stating Proposition 3.1.

PROPOSITION 3.1′. *$\vec{P}_n$ is graceful if and only if Z_n is sequenceable.*

For an excellent current survey of sequenceable groups, see [Ke].

Example 3.2. $\{s_0, s_1, \cdots, s_7\} = \{0, 1, 6, 3, 4, 5, 2, 7\}$ is a sequencing of the cyclic group Z_8. Consequently, $\{d_0, d_1, \cdots, d_7\} = \{0, 1, 7, 2, 6, 3, 5, 4\}$ is used to label the nodes of $\vec{P}_8$. (In addition to the unidirectional $\vec{P}_8$ being graceful, so, clearly, are the seven arc-pair similar orientations of its underlying graph P_8.)

Although little specific is known about the graceful labelling of directed trees, the following conjecture seems plausible.

CONJECTURE 3.3. *All trees are digraceful.*

This is a weaker conjecture than the one that claims that all undirected trees are graceful. If the stronger conjecture holds, then Conjecture 3.3 is true by using the trivial orientation of each graceful tree to produce a graceful directed tree. Even if the graceful tree conjecture is false, it is nevertheless possible for nontrivial edge orientations of ungraceful trees to give graceful digraphs.

4. Unions of unicycles. *Unidirectional cycles* (or *unicycles*) are connected digraphs in which every node has indegree = outdegree = 1. Some unicycles are graceful and some are not, as shown in Fig. 1. Moreover, some collections of disjoint unicycle components are graceful as is illustrated in Fig. 3, and some are not as is indicated in Proposition 4.1 which was proved in [BH1].

PROPOSITION 4.1. *For a union of $n \geqq 1$ unicycles to be graceful, it is necessary that the total number of arcs in the digraph be even.*

The remainder of this section specifies the relation between graceful unicycles

and complete mappings by establishing the relation of each to a particular class of permutations.

Example 4.2. If arc numbers are ignored, Fig. 3 can be regarded as the permutation (1 8 4)(2 3 6 5 7) of $Z_9\backslash\{0\}$.

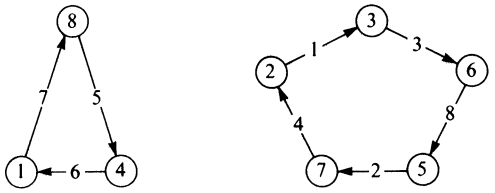


FIG. 3. *A graceful numbering of unicycle components,* $\vec{C}_3 \cup \vec{C}_5$, *using* Z_9.

DEFINITION 4.3. For a specified integer λ and sequence $K = \{k_1, k_2, \cdots, k_t\}$ in which the k_i are integers such that $\sum_{i=1}^{t} k_i = \lambda(n-1)$, a (K, λ) *complete mapping* is an arrangement of λ copies of the nonzero elements of Z_n into t cyclic sequences of lengths $k_1, k_2, \cdots, k_t$, say, $(g_{11}g_{12}\cdots g_{1k_1})(g_{21}g_{22}\cdots g_{2k_2})\cdots(g_{t1}g_{t2}\cdots g_{tk_t})$, such that the following *distinct difference property* holds. For $i = 1, 2, \cdots, t$ and $g_{i,(k_i+1)} = g_{i,1}$, the set of differences $\{g_{i,j+1} - g_{i,j}\}$ comprises λ copies of the nonzero elements of Z_n.

In other words, in the special case for $\lambda = 1$, a $(K, 1)$ complete mapping is a permutation of $Z_n\backslash\{0\}$ with t cycles, in which the set of modular differences between successive elements in the cycles equals $Z_n\backslash\{0\}$. (In Fig. 3 it is shown that Example 4.2 is a permutation which satisfies the distinct difference property.) In fact, when $\lambda = 1$, the distinct difference property is equivalent to requiring that all edge numbers be distinct in the graphical representation of the permutation cycles. Consequently, as a direct result of the definitions, the following characterization holds:

PROPOSITION 4.4. *A graceful numbering for* $\cup_{i=1}^{t} \vec{C}_{k_i}$, *where* $\sum_{1=i}^{t} k_i = e$, *exists if and only if there exists a* $(K, 1)$ *complete mapping of* Z_{e+1} *where* $K = \{k_1, \cdots, k_t\}$.

Study of complete mappings gives the following results:

PROPOSITION 4.5. *Let* $\vec{G} = \cup_{i=1}^{t} \vec{C}_i$, *the union of t disjoint identical unicycles on n nodes.* $\vec{G}$ *is graceful if* (a) $t = 1$ *and n is even*; *or if* (b) $t = 2$; *or if* (c) $n = 2$ *or* $n = 6$. *Moreover,* $\vec{G}$ *is not graceful if tn is odd.*

Proof. These have been proved in the context of generalized complete mappings (which include (K, λ) complete mappings) and these proofs are referenced rather than repeated here. Several researchers including the second author of this paper [H1] have independently proved (a). The proof for (b) is in [FGT]. For (c), the case for $n = 2$ is immediately demonstrated by letting the node numbers for the 2-cycles be $(0, e)$, $(1, e-1), \cdots, (t-1, t+1)$ where $e = 2t$ and the arc numbers are calculated in Z_{2t+1}. The case for $n = 6$ was demonstrated in [H1]. The ungraceful result for G is a special case of Proposition 4.1. □

Example 4.6. (1 6 5 7)(2 8 3 4) is a $(K, 1)$ complete mapping of Z_9 where $K = \{4, 4\}$. Hence, (1 6 5 7) and (2 8 3 4) are cyclical node sequences that give a graceful numbering of the unidirectional $\vec{C}_4 \cup \vec{C}_4$.

This section ends with two explicit constructions for gracefully numbering sets of isomorphic cycles. In the following let (x, y) denote the greatest common divisor of x and y.

PROPOSITION 4.7. *Let* $a \in Z_n\backslash\{0\}$ *such that* $a \neq 1$ *and both* $(a, n) = 1$ *and* $(a-1, n) = 1$. *Then a permutation* $\alpha(n) = an$ *on* $Z_n\backslash\{0\}$ *provides a graceful numbering of the digraph* $\vec{G} = \cup_i \vec{C}_i$ *where the unicycle length i is the least integer in* $Z_n\backslash\{0\}$ *such that* $a^i = 1$ *and where the number of unicycles is* $t = (n-1)/i$.

Proof. The nodes of the unicycles of digraph G are cyclically numbered with $(1\ a\ a^2 \cdots a^{i-1})(x_2\ ax_2\ a^2x_2 \cdots a^{i-1}x_2) \cdots (x_t\ ax_t\ a^2x_t \cdots a^{i-1}x_t)$ where x_i is any node number not listed in any of the previous $j-1$ lists of unicycle node numbers. Consequently, $ax - x = \alpha(x) - x$. Since $(a-1, n) = 1$, the arc numbers are all distinct and the numbering is graceful. □

Proposition 4.8 is an immediate corollary of Proposition 4.7.

PROPOSITION 4.8. *Let p be an odd prime. Then permutation $\alpha(n) = an$ on $Z_n\backslash\{0\}$ for $a \in \{2, 3, \cdots, p-1\}$ provides a graceful numbering of digraph $\vec{G} = \cup_t \vec{C}_i$ where i is the order of a and $t = (p-1)/i$.*

Examples 4.9. For $p = 7$, graceful numberings are generated for unidirectional $\vec{C}_6$, $2\vec{C}_3$, and $3\vec{C}_2$ as follows: For $a = 2$, (1 2 4)(3 6 5); for $a = 3$, (1 3 2 6 4 5); for $a = 4$, (1 4 2)(3 5 6); for $a = 5$, (1 5 4 6 2 3); for $a = 6$, (1 6), (2, 5), (3, 4).

In § 5 it is observed that some unions of cycles and paths are graceful; then in § 6, graceful digraphs are discussed more generally as models for algebraic structures built upon cyclic multiplicative groups.

5. Collections of unicycles and paths. Figure 4 shows a gracefully numbered three-component digraph consisting of two unidirectional 3-cycles and a single edge path. In a manner similar to that of the previous section, the graceful unions of unicycles and unidirectional paths can be characterized.

Since the components in Fig. 4 are not all cycles, Fig. 4 cannot be viewed as representing a permutation in the way that Fig. 3 was; however, it is almost a permutation. That is, the mapping is almost bijective, going from $Z_8\backslash\{4\}$ to $Z_8\backslash\{0\}$. The "almost permutation" character of Fig. 4 corresponds to the following algebraic structure for the case $\lambda = 1$.

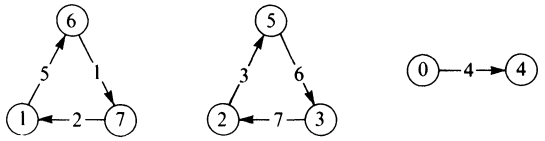


FIG. 4. *A graceful numbering of unidirectional $2\vec{C}_3 \cup \vec{P}_2$ using Z_8.*

DEFINITION 5.1. For a given integer λ and sequence $K = \{k_1, k_2, \cdots, k_r; h_1, h_2, \cdots, h_s\}$ such that the h_i and k_j are integers satisfying $\sum_{i=1}^{r} k_i + \sum_{j=i}^{s} h_j = \lambda n$, a (K, λ) *near-complete mapping* is an arrangement of λ copies of the elements of Z_n into r cyclic sequences with lengths $k_1, \cdots, k_r$ and s sequences of lengths $h_1, \cdots, h_s$, say, $(g_{11} \cdots g_{1k_1}) \cdots (g_{r1} \cdots g_{rk_r})[g'_{11} \cdots g'_{1k1}] \cdots [g'_{s1} \cdots g'_{sks}]$, such that the following distinct difference property holds for $i = 1, 2, \cdots, r$, $j = 1, 2, \cdots, s$, and $g_{i,(k_i+1)} = g_{i,1}$, the sets of differences $\{g_{i,j+1} - g_{i,j}\}$ and $\{g'_{i,j+1} - g'_{i,j}\}$ together comprise λ copies of Z_n.

The correspondence between near-complete mappings and graceful digraphs is made explicit in Proposition 5.2.

PROPOSITION 5.2. *Let n = total number of nodes and e = total number of arcs in a digraph. A graceful numbering of $(\cup_{i=1}^{r} \vec{C}_{k_i}) \cup (\cup_{j=i}^{s} \vec{P}_{h_j})$, where $\sum_{1=i}^{r} k_i + \sum_{j=i}^{s} h_j = n = e + s$, occurs if and only if there exists a $(K, 1)$ near-complete mapping of $Z_n = Z_{e+s}$ where $K = \{k_1, \cdots, k_r; h_1, \cdots, h_s\}$.*

The proof of 5.2 is similar to that of Proposition 4.3.

PROPOSITION 5.3. *A graceful digraph D comprising a collection of both unicycles and unidirectional paths must contain exactly one path and contain an odd total number of arcs.*

Proof. For a digraph with n nodes and e arcs to be graceful, it is necessary that $n \leqq e+1$. Consequently, no more than one path can be present. Proving that the cardinality of the arc set must be odd is accomplished by establishing a contradiction for the sum of the arc values when the cardinality is even. If $D = (\bigcup_{i=j}^{s} \vec{C}_{n_i}) \cup \vec{P}_k$ is gracefully numbered by θ so that $\theta(v_{ij})$ represents the node value of the jth node in cycle i and $\theta(v_j)$ represents the node value of the jth node in the path, then the sum of the arc values can be written as

$$\sum_{(u,v)} (u, v) = \sum_{(u,v)} (\theta(u) - \theta(v))$$

$$= \sum_{i=1}^{s} \sum_{j=1}^{h_i} (\theta(v_{i,j+1}) - \theta(v_{i,j})) + \sum_{j=1}^{k-1} (\theta(v_{j+1}) - \theta(v_j)).$$

For the ith cycle, the inner sum is

$$\sum_{j} (\theta(u_{i,j+1}) - \theta(u_{i,j})) = \theta(u_{i,h_i+1}) - \theta(u_{i,1}) = 0.$$

For the path on k nodes, a similar cancellation of node numbers leaves only the first and last values, $\theta(u_k) - \theta(u_1)$, in the sum. Consequently, for all of D,

$$\sum_{D} \theta(u, v) = \theta(u_k) - \theta(u_1).$$

On the other hand, the sum of arc labels in D is the sum of the nonzero elements of Z_n, i.e.

$$\sum_{D} \theta(u, v) = \sum i = n(n-1)/2 \pmod{n}.$$

If the number of arcs, e, in D is even, then n is odd, which implies that $(n-1)/2$ is an integer. Thus, $n(n-1)/2 \equiv 0 \pmod{n}$, which implies that $\theta(u_k) - \theta(u_1) \equiv 0 \pmod{n}$. Consequently, when $e = \text{even}$, $\theta(u_k) = \theta(u_1)$, which violates the requirement for distinctly numbering nodes in graceful numberings. The contradiction is established; thus for graceful D, n cannot be odd and e cannot be even. (Note that when e is odd, this contradiction does not occur.)

Example 5.4. A $(K, 1)$ near-complete mapping of Z_{14} for $K = \{3, 4, 5; 2\}$ is (1 2 4) (6 10 8 11)(3 9 5 13 2)[0 7] which provides a graceful numbering for the unidirectional components in $\vec{C}_3 \cup \vec{C}_4 \cup \vec{C}_5 \cup \vec{P}_2$.

The close relation between structures discussed in this section and in § 6 has been studied algebraically in [HK1] and [HK2]. It has been useful to have the following definition.

DEFINITION 5.5. A *generalized complete mapping* is either a $(K, 1)$ complete mapping or a $(K, 1)$ near-complete mapping.

A collection of generalized complete mappings can be found in [HK2].

6. Digraph models for cyclic groups and other structures. In §§ 3, 4, and 5 the application of certain algebraic structures for generating graceful digraphs was made evident. Moreover, as Proposition 5.3 showed, certain algebraic facts may also be gleaned from graceful digraphs. In this section some additional connections are made between graceful digraphs and Latin squares, Abelian groups, Galois fields, and neofields. More extensive expositions of the relations among the algebraic structures themselves can be found, for example, in [H1], [HK1], and [HK2].

Let H_n be the cyclic multiplicative group of order n, $H_n = \{1, a, a^2, \cdots, a^{n-1}\}$ with generator a. Augment this group with a zero element 0 $(x * 0 = 0 * x = 0)$ to form

$N_{n+1} = H_n \cup \{0\}$. A permutation π_1 is defined on N_{n+1}, in which, for reasons of convenience, $\pi_1(0) = 1$. The permutation can be represented as a digraph which is termed the unit-addition digraph A_1 for (N_{n+1}, π_1). Figure 5 illustrates the unit-addition digraphs for Examples 6.1–6.3.

Example 6.1a. *On* N_8

x	0	1	a	a^2	a^3	a^4	a^5	a^6
$\pi_1(x)$	1	0	a^2	a^4	a^6	a	a^3	a^5

Example 6.1b. *On* N_8

x	0	1	a	a^2	a^3	a^4	a^5	a^6
$\pi_1(x)$	1	0	a^2	a^3	a	a^5	a^6	a^4

Example 6.2. *On* N_9

x	0	1	a	a^2	a^3	a^4	a^5	a^6	a^7
$\pi_1(x)$	1	a^4	a^6	a^2	a^2	0	a^3	a^7	a

Example 6.3. *On* N_{10}

x	0	1	a	a^2	a^3	a^4	a^5	a^6	a^7	a^8
$\pi_1(x)$	1	0	a^8	a^3	a^6	a	a^7	a^5	a^2	a^4

A second operation, addition, is defined upon N_{n+1} in terms of the permutation for $x \in N_{n+1}$, $1 + x = \pi_1(x)$. It is required that $x(y+z) = xy + xz$ and $(y+z)x = yx + zx$ for $x, y, z \in N_{n+1}$; but no requirements of commutivity or associativity are made. These relations allow the calculation of addition tableaux for $(N_{n+1}, +)$, as shown for Examples 6.1a and 6.1b.

Example 6.1a

+	0	1	a	a^2	a^3	a^4	a^5	a^6
0	0	1	a	a^2	a^3	a^4	a^5	a^6
1	1	0	a^2	a^4	a^6	a	a^3	a^5
a	a	a^6	0	a^3	a^5	1	a^2	a^4
a^2	a^2	a^5	1	0	a^4	a^6	a	a^3
a^3	a^3	a^4	a^6	a	0	a^5	1	a^2
a^4	a^4	a^3	a^5	1	a^2	0	a^6	a
a^5	a^5	a^2	a^4	a^6	a	a^3	0	1
a^6	a^6	a	a^3	a^5	1	a^2	a^4	0

Example 6.1b

+	0	1	a	a^2	a^3	a^4	a^5	a^6
0	0	1	a	a^2	a^3	a^4	a^5	a^6
1	1	0	a^2	a^3	a	a^5	a^6	a^4
a	a	a^5	0	a^3	a^4	a^2	a^6	1
a^2	a^2	a	a^6	0	a^4	a^5	a^3	1
a^3	a^3	a	a^2	1	0	a^5	a^6	a^4
a^4	a^4	a^5	a^2	a^3	a	0	a^6	1
a^5	a^5	a	a^6	a^3	a^5	a^2	0	1
a^6	a^6	a	a^2	1	a^4	a^5	a^3	0

The next proposition highlights a graph theoretical technique for expediently calculating and representing the rows of the addition table for any $(N_{n+1}, +)$.

PROPOSITION 6.4. *Given a cyclic group* $(H_n, *)$ *of order n with generator a, the kth row of the addition table for* $(N_{n+1}, +)$ *corresponds to the labelled permutation digraph* A_k *generated by multiplying the node values of* A_k *by* a^k.

This proposition is proved by exhibiting the correspondence between the digraph mapping and the algebraic relations.

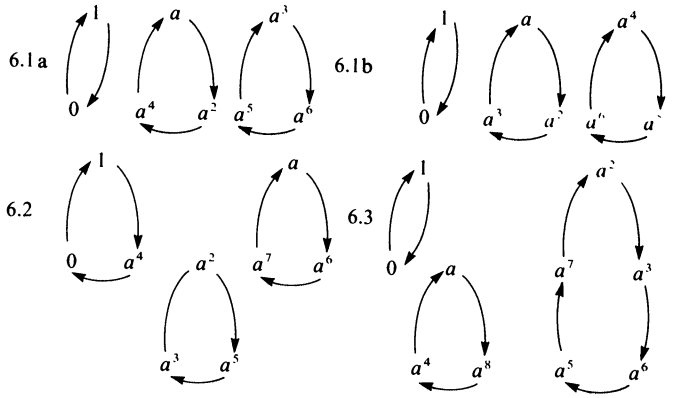


FIG. 5. *Labelled unit-addition digraphs* (A_1) *representing the permutations used in Examples* 6.1-6.3.

Proof. Let $\pi_i(x)$ indicate the image of x in the labelled digraph permutation A_i. In A_k the image of a^j under π_k is

$$\pi_k(a^j) = \pi_k(a^k a^{j-k}) = a^k \pi_1(a^{j-k}),$$

that is, in the kth permutation graph the mapping $a^j \to \pi_k(a^j)$ comes from multiplying the node labels in the mapping $a^{j-k} \to \pi_1(a^{j-k})$ given in A_1 by a^k. The correspondence to the proper result in the addition table is a direct result of (either one of) the distributive laws of multiplication over addition, e.g. $a^k + a^j = a^k(1 + a^{j-k}) = a^k \pi_1(a^{j-i})$. □

Example 6.1a *continued.* The ith entry in the kth row of the addition table, $a^k + a^j$ for $0 \leqq k \leqq 6$, can be read directly from A_k as shown in Fig. 6. Each of the node values in A_k is a^k times the unit-addition permutation A_1 shown in Fig. 5.

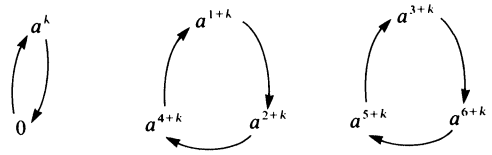


FIG. 6. *The labelled digraph* A_k *showing the result of adding* a^k *to the elements of* N_8 *from Example* 6.1a, *i.e.* $a^k + u = v$ *for arc* (u, v).

Examples 6.1a and 6.1b are also of interest because of their differences, as well as because of the similar way that their addition tables can be generated graphically. Since no element of N_8 in Example 6.1a is repeated in any row or column of the addition table, that table is a *Latin square* (written in *normalized form* with the first row equaling $0, 1, a, a^2, \cdots, a^6$). The addition table of Example 6.1b is not a Latin square. In the remainder of this section, the significance for graceful digraphs of the first of these two examples, 6.1a is explained.

The structure $(N_{n+1}, *, +)$ whose addition table is a normalized Latin square is called a cyclic neofield.

DEFINITION 6.5. A *neofield* $(S, *, +)$ consists of a set S upon which two binary operations $+$ and $*$ are defined provided that (a) the addition table for $(S, +)$ can be written as a normalized Latin square; (b) $(S\backslash\{0\}, *)$ is a group; and (c) multiplication distributes over addition on both left and right. A neofield is *cyclic* if $(S\backslash\{0\}, *)$ is a cyclic group.

A neofield is a finite field in which the associativity and commutativity of addition are not required. Cyclic neofields have been characterized in [H1].

Using $(K, 1)$ complete mappings and $(K, 1)$ near-complete mappings, it was shown in §§ 4 and 5 respectively that graceful digraphs could be generated. Proposition 6.6 in turn implies that neofields generate a class of gracefully numbered digraphs. It should also be noted that Galois fields are a class of commutative cyclic neofield.

The salient property that enables cyclic neofields, neofields, and the generalized complete mappings (that the former structures imply) to generate graceful digraphs is the "distinct difference property" (introduced in § 5) that is inherent in their permutation (or row addition) labelled digraphs. It is the lack of this property that prevents Example 6.1b from generating a Latin square and hence a cyclic neofield. A summary of the interrelationships between graceful digraphs and cyclic neofields follows.

PROPOSITION 6.6. *Let H_n be a cyclic group of order n. Let N_{n+1}, π_1, π_k, A_1, A_k be defined as before. For any fixed k, let v_0 be the node with label 0 in A_k and let A be the digraph A_k with its node labels removed. Let $D = A - v_0$. Then π_1 defines a cyclic neofield $(N_{n+1}, *, +)$ with $N_{n+1} = H_n \cup \{0\}$ if and only if the digraph D is graceful.*

Proof. Suppose π_1 defines a cyclic neofield $(N_{n+1}, *, +)$ with $N_{n+1} = H_n \cup \{0\}$. By Definition 6.5 π_1 generates a normalized Latin square addition table. By Proposition 6.4, it suffices to show that the digraph D generated by A_1 is graceful.

The labelled unit-addition digraph A_1 has node labels $0, 1, a, a^2, \cdots, a^{n-1}$. Hence D has n nodes and $n-1$ arcs. Suppose D has the same labelling as A_1. Label the arc (a^i, a^j) as $a^{j-i} = \pi_1(a^i) * a^{-i}$. If two arcs (a^i, a^j) and (a^k, a^h) have the same labelling, i.e., $\pi_1(a^i) * a^{-i} = \pi_1(a^k) * a^{-k}$, then $(1 + a^i) * a^{-i} = (1 + a^k) * a^{-k}$.

Furthermore, by the distributive law:

$$1 + a^i = a^i * (a + a^k) * a^{-k} = (a^i + a^{i+k}) * a^{-k} = a^{i-k} + a^i.$$

Since the addition table is a Latin square, $a^{i-k} = 1$. That is $a^i = a^k$. Therefore, D has distinct arc labellings and is graceful with respect to the cyclic group H_n. Hence, it is graceful.

The proof that a graceful digraph yields a cyclic neofield follows from reversing the above argument. □

Examples 6.2 *and* 6.3 (*continued*). A transformation of the latter two digraphs of Fig. 5 to the gracefully numbered digraphs in Figs. 3 and 4 can be achieved by first eliminating the nodes labelled zero from the original digraphs and then by assigning the exponents of a to be the new node numbers.

A corollary of Propositions 6.4 and 6.6 follows.

PROPOSITION 6.7. *The n labellings of the labelled digraph A generated from the n rows of a cyclic neofield addition table based on H_n correspond to the set of n equivalent graceful numberings of $A - v_0$.*

The correspondence established in §§ 4, 5 and 6 between algebraic structures and graceful digraphs generates a plethora of questions concerning how these topics can yield mutual insights. Nevertheless, the set of digraphs contributing to this correspondence is limited. In the next two sections, methods of generating additional graceful digraphs are presented.

7. A graceful supergraph construction. To extend the class of known graceful digraphs, it is natural to seek methods for building larger graceful digraphs from smaller gracefully numbered ones. Figure 7a shows such a construction in which arcs are connected from two isolated nodes to a gracefully numbered unidirectional path $\vec{P}_8$ (numbered as in Example 3.2) to form the illustrated gracefully renumbered digraph

$\overrightarrow{\vec{P}_8+\{u, v\}}$. In general, the notation $\overrightarrow{D+K_m^c}$ specifies the digraph obtained by directing arcs from each of m isolated nodes to each of the nodes of a digraph D. For any digraph obtained by this construction, the following holds:

PROPOSITION 7.1. *If D is a graceful digraph with n nodes and $n-1$ arcs, then $\overrightarrow{D+K_m^c}$ is graceful for every finite m.*

Proof. Let θ be a graceful numbering of D, a graceful digraph with n nodes $u_1, \cdots, u_n$, and $n-1$ arcs. Designate the nodes of K_m^c by $v_1, \cdots, v_m$. Number the $n+m$ nodes of the digraph $E=\overrightarrow{D+K_m^c}$ by ψ as follows:

(i) $\psi(u_i)=(m+1)\theta(u_i), \qquad i=1, \cdots, n,$

(ii) $\psi(v_i)=i, \qquad i=1, \cdots, m.$

Since the total number of arcs in E is $(n-1)+mn$ (i.e. the number of arcs in D plus the new, connecting arcs), arithmetic in E is done in Z_{n+mn}. (Inasmuch as the node numbers assigned in (i) are bounded above by $n+mn$, modular arithmetic is not needed to compute the node values.) To prove that ψ is a graceful numbering, it is necessary only to show that the $n+m$ node values are distinct and that the $(n+1)+mn$ arc values comprise $Z_{n+mn}\backslash\{0\}$.

Clearly, the node numbers in E are distinct, since (1) the nodes of D are numbered by distinct, constant multiples of their value in D, and (2) the node numbers of K_m^c are the distinct, positive integers $\{1, 2, \cdots, m\}$, each of which is less than the nonzero node values of D.

The numbers on arcs in E can be directly calculated. For convenience, they can be viewed relative to $(m+1)$ as follows: (a) For arcs in the D subgraph of E,

$$\begin{aligned}\psi(u_i, u_j) &= \psi(u_j)-\psi(u_i)\\ &=(m+1)\theta(u_j)-(m+1)\theta(u_i)\\ &=(m+1)[\theta(u_j)-\theta(u_i)]\\ &\equiv 0 \mod (m+1).\end{aligned}$$

Thus, calculating in $Z_{n(m+1)}$, the arcs of D are numbered with the $n-1$ distinct nonzero multiples of $m+1$. (b) For arcs emanating from node v_i,

$$\psi(v_i, u_j)=\psi(u_j)-\psi(v_i)=(m+1)\theta(u_j)-i\equiv -i \mod (m+1).$$

That is, in $Z_{n(m+1)}$ these n arcs bear the n distinct values $(m+1)-1$, $2(m+1)-i$, $\cdots$, $n(m+1)-i$. Since these values are distinct for each i, $1\leqq i\leqq m$, all arc numbers are distinct and E is graceful. □

Figure 7b illustrates this construction used on the disconnected gracefully numbered digraph of Fig. 4 to produce a gracefully numbered $\overrightarrow{(2\vec{C}_3\cup\vec{P}_2)+K_1}$. Because the graceful numbering of E used in the previous proof puts the arc labels of subgraph D in a congruence equivalence class distinct from the other arcs, it is possible to reverse the direction of all the arcs in E that are not in D to obtain an arc-pair similar digraph which by Proposition 1.2 is also graceful. Thus, if the notation $\overrightarrow{(D+K_m^c)}'$ specifies the digraph obtained by directing arcs to each of m isolated points from each of the nodes of a digraph D, the following proposition holds:

PROPOSITION 7.2. *If D is a graceful digraph with n nodes and $n-1$ arcs, then $\overrightarrow{(D+K_m^c)}'$ is graceful.*

This construction is applied and extended in § 8.

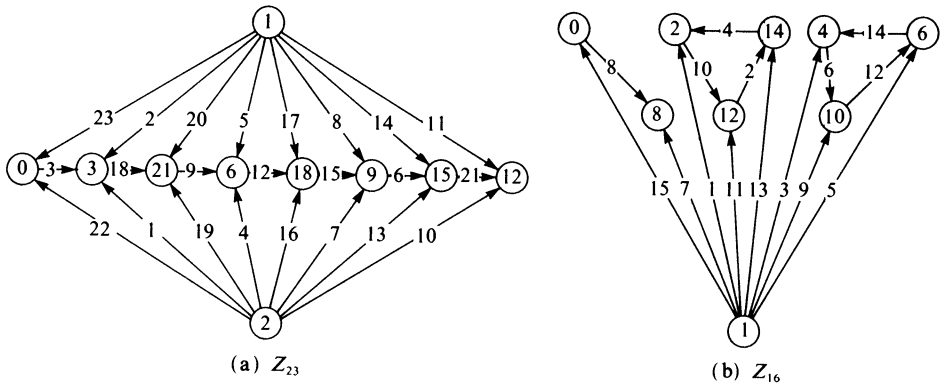


FIG. 7. *Two graceful digraphs determined by joining isolated points to each node of an already graceful digraph for which* $n = e+1$.

8. Other families of graceful digraphs.

A. Directed windmills. An undirected *windmill graph* comprises t n-cycles joined at exactly one node and is designated $(t)C_n$. A study of graceful windmills was made in [Be]. The gracefulness of one digraph family for which windmills are the underlying graphs is considered here. Recall that $\vec{C}_3$ represents a unicycle of length 3.

PROPOSITION 8.1. *The unicyclic windmill digraph* $(t)\vec{C}_3$ *is graceful if and only if* t *is even.*

Proof. If t is even, say $t=2s$, then $(t)\vec{C}_3$ contains $6s$ arcs to be numbered with $\{1, \cdots, 6s\}$. It is known (e.g. see [H1]) that $Z_{6s+1}\backslash\{0\}$ admits a partition into s sets of the form $\{x, y, y-x, -x-y, x-y\}$. Each of these sets will number the arcs of 2 "vanes" of the windmill, when the 5 nodes are numbered as shown in Fig. 8. The node assigned zero in this numbering serves as the common node for each pair of vanes. Thus, the s distinct sextuples number the $6s$ arcs of $(2s)\vec{C}_3$ when the $4s+1$ nodes are numbered as indicated. Hence, this numbering is graceful.

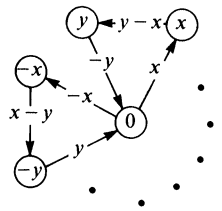


FIG. 8. *Paired vanes of a windmill graph.*

If t is odd, say $t=2s+1$, there are $3t=6s+3$ arcs. The sum of the arc numbers is

$$\sum_{i=1}^{6s+3} i = (6s+4)(6s+3)/2 = (3s+2)(6s+3).$$

The modular sum for this digraph with $6s+3$ arcs is taken in Z_{6s+4}. Thus,

$$\sum_{i=1}^{6s+3} i \equiv -(3s+2) \mod (6s+4).$$

However, each arc lies on a unicycle, around which the sum of arc numbers is zero. Since $3s+2 \not\equiv 0 \mod (6s+4)$, there can be no graceful numbering of windmills bearing an odd number of triangular unicyclic vanes. □

B. Directed wheels. An undirected graph consisting of a node (the "hub") joined to each node of a cycle (the "rim") is termed a *wheel.* It was shown in [Fr] and [HoK] that all wheels are graceful and, hence, digraceful. Here it is shown that certain nontrivial orientations of wheels are graceful.

A directed wheel is termed *outspoken* if all spokes point out from the hub to the rim and *inspoken* if all spokes point from the rim toward the hub. If the rim of a directed wheel is a unicycle, the wheel is called *unicyclic.*

PROPOSITION 8.2. *An inspoken unicyclic wheel on n nodes is graceful if and only if the outspoken unicyclic wheel on n nodes is graceful.*

Proof. This corollary to Proposition 1.3 results from noting that the outspoken unicyclic wheel $\vec{W}_n$ has the inspoken unicyclic wheel as its corresponding reversed digraph $(-\vec{W}_n)$. □

PROPOSITION 8.3. *Let p be an odd prime number and α be a primitive element of Z_p. If $\alpha^2-1\equiv\alpha^{2k+1}\,(\mathrm{mod}\,p)$ for some k, then the outspoken and inspoken unicyclic wheels $\vec{W}_q$ and $-\vec{W}_q$ are graceful for $q=(p-1)/2$.*

Proof. Since α is a primitive element of Z_p, $Z_p\backslash\{0\}=\{1,\alpha,\alpha^2,\cdots,\alpha^{p-3}\}$. $Z_p\backslash\{0\}$ can be partitioned into its quadratic residues $R=\{1,\alpha^2,\alpha^4,\cdots,\alpha^{p-3}\}$, and quadratic nonresidues $R^c=\{\alpha,\alpha^3,\alpha^5,\cdots,\alpha^{p-2}\}$. Number the unicyclic rim of $\vec{W}_q$ with the successive powers of α^2 and the hub with zero. Then the spokes take the quadratic residues as their values. Since $\alpha^2-1\equiv\alpha^{2k+1}\in R^c$, and $\alpha^{2i+2}-\alpha^{2i}=\alpha^{2i}(\alpha^2-1)$, the arcs on the rim are numbered with

$$\{\alpha^{2i+2}-\alpha^{2i}\colon i=0,1,\cdots,q-1 \text{ and } \alpha^{p-1}=1\}=R^c.$$

Since the arcs of $\vec{W}_q$ take all values in Z_p, this is a graceful numbering of $\vec{W}_q$. By Proposition 8.2 $-\vec{W}_q$ is also graceful. □

Examples 8.4. Listed in Table 1 are all rim number sequences for gracefully numbered outspoken and inspoken unicyclic wheels obtained from the quadratic residues as indicated in Proposition 8.3. The outspoken rim number sequence is read left to right and the inspoken rim number sequence is read right to left. Zero numbers the hub in all cases.

TABLE 1

n	p	$\alpha^2-1\equiv\alpha^{\mathrm{odd}}\,(\mathrm{mod}\,p)$	rim sequence (quadratic residues)
2	5	$2^2-1\equiv 3\equiv 2^3$	4 1
3	7	$5^2-1\equiv 3\equiv 5^5$	4 2 1
5	11	$6^2-1\equiv 2\equiv 6^9$	3 9 5 4 1
		$8^2-1\equiv 8\equiv 8^1$	9 4 3 5 1
8	17	$5^2-1\equiv 7\equiv 5^{15}$	8 13 2 16 9 4 15 1
		$12^2-1\equiv 7\equiv 12^7$	8 13 2 16 9 4 15 1
		$7^2-1\equiv 14\equiv 7^7$	15 4 9 16 2 13 8 1
		$10^2-1\equiv 14\equiv 10^3$	15 4 9 16 2 13 8 1
9	19	$2^2-1\equiv 3\equiv 2^{13}$	4 16 7 9 17 11 6 5 1
		$3^2-1\equiv 8\equiv 3^3$	9 5 7 6 16 11 4 17 1
		$15^2-1\equiv 15\equiv 15^1$	16 9 11 5 4 7 17 6 1
11	23	$10^2-1\equiv 7\equiv 10^{21}$	8 18 6 2 16 13 12 4 9 3 1
		$11^2-1\equiv 5\equiv 11^5$	6 13 9 8 2 12 3 18 16 4 1
		$14^2-1\equiv 11\equiv 14^{13}$	12 6 3 13 18 9 16 8 4 2 1
		$15^2-1\equiv 17\equiv 15^3$	18 2 13 4 3 8 6 16 12 9 1
		$19^2-1\equiv 15\equiv 19^7$	16 3 2 9 6 4 18 12 8 13 1

Some rim sequences arising from Proposition 8.3 meet other conditions.

PROPOSITION 8.5. *Let p be an odd prime number, let $q=(p-1)/2$, and let α be a primitive element of Z_p. If $\alpha^2-1\equiv\alpha^{2k}\pmod p$ for some k and if $p\equiv 3\pmod 4$, then the inspoken and outspoken wheels $\vec{W}_q$ are graceful.*

Proof. The nodes on the rim of the inspoken wheel are numbered with successive powers of α^2. For the arcs numbered this way, the arc numbers are themselves the quadratic residues of Z_p. Since $(-1)=\alpha^{(p-1)/2}\equiv\alpha^{2i+1}\pmod p$ for some i, if $p\equiv 3\pmod 4$, the arc numbers for inward directed spokes are the quadratic nonresidues of Z_p. Proposition 8.2 indicates that reversing all arcs in this numbered digraph generates a gracefully numbered outspoken unicyclic wheel. □

The node sequences in Examples 8.4 for $n=3, 5, 19, 23$ all can be determined from Proposition 8.5. The condition $p\equiv 3\pmod 4$ in Proposition 8.5 is essential to disallow node sequences consisting of quadratic residues that do not give graceful numberings. Ruled out, for example, is the sequence 4 3 12 9 10 1 for $n=6$ ($p=13$) where $2^2-1\equiv 3\equiv 2^4$ and $-1\equiv 12\equiv 2^6\pmod{13}$.

Unicyclic wheels for which $p=2n+1$ is not prime can also be graceful.

Example 8.6. Outspoken wheels $\vec{W}_4$ and $\vec{W}_7$ can be gracefully numbered by numbering the hubs with zero and by assigning the following sequences to the rims respectively: 3 8 6 1 and 3 8 7 11 9 6 1. The corresponding inspoken numberings use these sequences from right to left.

For $n\leqq 11$, all unicyclic wheels $\vec{W}_n$ are known to be graceful except for $n=6$ and $n=10$. Can graceful numberings for these be found? And, more generally, will the results for unicyclic wheels be as straightforward as for undirected wheels, that is, is the following conjecture true?

CONJECTURE 8.7. *All unicyclic wheels are graceful.*

C. General construction techniques. In [BH1] strict bounds were determined for the number of arcs that are required in a graceful digraph that embeds a graceful digraph augmented by a pendant arc.

Different graceful superdigraphs of an arbitrary graceful digraph D are formed by using Proposition 7.1 and the *node deficiency* of D, $d(D)=e+1-n$, where n and e respectively designate the number of nodes and arcs in D. (1) Gracefully number the nodes of D with n values from the set $\{0, 1, \cdots, e\}$; (2) augment D with d isolated nodes to which the remaining $e+1-n$ values of $\{0, 1, \cdots, e\}$ are assigned. The new graceful digraph contains D as a gracefully labelled induced subgraph; (3) now connect m new nodes to the augmented D as described in Propositions 7.1 and 7.1′.

PROPOSITION 8.8. *A graceful digraph D with n nodes and positive node deficiency d can be embedded as a subgraph in a connected graceful digraph $D'=\overrightarrow{(D\cup K_d^c)+K_m^c}$ with $n+d+1$ nodes.*

A corollary to this proposition concerns outspoken (or inspoken) *umbrella digraphs*, $\vec{U}_n=\overrightarrow{(C_n\cup K_1)+K_1}$, which are outspoken (or inspoken) wheels with one additional outwardly (inwardly) directed arc from (toward) the hub. Since a cycle has node deficiency one, the union of an isolated node with a gracefully labelled cycle allows the application of Proposition 7.1 (7.1′) as indicated.

PROPOSITION 8.9. *A unicyclic outspoken (or inspoken) umbrella digraph $\vec{U}_n$ is graceful if n is even.*

Since for odd n, $\vec{C}_n$ is not graceful, the construction in Proposition 8.9 does not apply to umbrellas with odd n.

Question 8.10. For odd values of n, is $\vec{U}_n$ graceful (or does Proposition 8.9 completely characterize graceful unicyclic umbrellas)?

9. Concluding remarks. Graceful digraphs provide a plethora of possibilities for further exploration. For example, we have shown [BH2, BH3] that graceful digraphs are characterized by a canonical form of their adjacency matrices. Moreover, a subset of these matrices give solutions to a constrained "n-queens" problem. We have also shown that graceful digraphs generated classes of combinatorial designs [BH4]. There are also possibilities to loosen constraints in investigating graceful digraphs. For instance, as the following examples demonstrate, it is not necessary to number graphs only with additive groups of integers.

Let D denote a directed graph and have G denote a group. D may be said to be *graceful with respect to* G, i.e. (D, G) is graceful, if (a) D has e arcs and G has $e+1$ elements; if (b) numbering θ assigns the distinct elements of G to the nodes of D and the arcs (u, v) are assigned labels using $\theta(v)\theta^{-1}(u)$; and if (c) the resulting arc labels are distinct and nonidentity.

For example, unidirectional paths can be labelled by assigning the appropriate sequence of elements of sequenceable groups to the nodes of a path (see, for example, [Ke] and § 5 of this paper). Thus, one can use the dihedral group of order 10 ($a^5 = b^2 = e$, $ab = ba^{-1}$) to label the nodes of a 10-node unidirectional path in the following order: $e, a, a^2, b, ba, ba^4, ba^2, ba^3, a^3, a^4$. Similarly, a unidirectional path of 21 nodes can be gracefully labelled with the non-abelian group of order 21 ($a^3 = b^7 = e$, $a^{-1}ba = b^2$) by assigning elements to the nodes in the following order: $e, b, b^2, b^3, b^5, a, b^4, ab, b^6, ab^3, a^2b, a^2b^2, ab^6, a^2, ab^2, a^2b^4, ab^5, a^2b^6, ab^4, a^2b^3, a^2b^5$.

Generalized graceful directed graphs can similarly be defined by $(D, \lambda G)$ where D has λe arcs, G has $e+1$ elements each of which appears as an arc label on D exactly λ times as a result of a graceful assignment of the $e+1$ node labels. A more extensive examination of algebraic questions associated with generalized graceful numberings is found in [BH5].

Open questions remain, of course, for the nongeneralized graceful numberings discussed in the main body of this paper. For example, the following questions are currently unanswered:

- How many distinct graceful numberings does a designated graceful digraph have?
- For which classes of undirected graphs can graceful orientations always be found?
- What is the probability that a digraph is graceful?

Of course, the following metaquestion is of central interest:

What other mathematical and "real world" applications can be determined for graceful digraphs?

Acknowledgments. Careful readings of this paper by S. A. Burr and C. Delorme and their helpful suggestions were greatly appreciated.

REFERENCES

[Ba] L. D. BAUMERT, *Cyclic Difference Sets*, Lecture Notes in Mathematics 182, Springer-Verlag, Berlin, 1971.

[Be] J. C. BERMOND, *Graceful graphs, radio antennae, and French windmills*, in Graph Theory and Combinatorics, R. Wilson, ed., Pitman, London, 1979, pp. 18–37.

[Bl] G. S. BLOOM, *Numbered undirected graphs and their uses: A survey of a unifying scientific and engineering concept and its use in developing a theory of non-redundant homometric sets relating to some ambiguities in x-ray diffraction analysis.* Ph.D. dissertation, Univ. Southern California, Los Angeles, 1975.

[B2] ———, *A chronology of the Ringel-Kotzig conjecture and the continuing quest to call all trees graceful*, in Topics in Graph Theory, Annals of the New York Academy of Sciences, 328, F. Harary, ed., New York Academy of Sciences, New York, 1979, pp. 32-51.

[BG1] G. S. BLOOM AND S. W. GOLOMB, *Applications of numbered undirected graphs*, Proc. IEEE, 65 (1977), pp. 562-570.

[BG2] ———, *Numbered complete graphs, unusual rulers, and assorted applications*, in Theory and Applications of Graphs, Lecture Notes in Mathematics, 642, Y. Alavi and D. R. Lick, eds., Springer-Verlag, Berlin, 1978, pp. 53-65.

[BH1] G. S. BLOOM AND D. F. HSU, *On graceful graphs and a problem in network addressing*, in Congressus Numerantium 35, Utilitas Mathematica, Winnipeg, 1982, pp. 91-103.

[BH2] ———, *Graceful directed graphs*, Tech. Report # CCNY-CS283, City College of New York, New York, 1983.

[BH3] ———, *Adjacency matrices of graceful directed graphs*, in preparation.

[BH4] ———, *Digraph designs*, in preparation.

[BH5] ———, *On graceful digraphs that are computational models of some algebraic systems*, in Graph Theory with Applications to Algorithms and Computer Science, Y. Alavi et al., eds., John Wiley, New York, to appear.

[CHR] G. J. CHANG, D. F. HSU AND D. G. ROGERS, *Additive variations on a graceful theme: Some results on harmonious and other related graphs*, in Congressus Numerantium 30, Utilitas Mathematica, Winnipeg, 1981, pp. 181-197.

[CH] F. R. K. CHUNG AND F. K. HWANG, *Rotatable graceful graphs*, Ars Combinatoria, 11 (1981), pp. 239-250.

[CR] M. B. COZZENS AND F. S. ROBERTS, *T-colorings of graphs and the channel assignment problem*, in Congressus Numerantium 35, Utilitas Mathematica, Winnipeg, 1982, pp. 191-208.

[De] C. DELORME, private communication.

[FGT] R. J. FRIEDLANDER, B. GORDON AND P. TANNENBAUM, *Partitions of groups and complete mappings*, Pacific J. Math., 92 (1982) pp. 283-293.

[Fr] ROBERTO FRUCHT, *The graceful numbering of wheels and related graphs*, in Proc. of 2nd International Conference on Combinatorial Mathematics, Annals of the New York Academy of Sciences, 319, A. Gewirtz and L. V. Quintas, eds., New York Academy of Sciences, New York, 1979, pp. 219-229.

[Go] S. W. GOLOMB, *How to number a graph*, in Graph Theory and Computing, R. C. Reed, ed., Academic Press, New York, 1972, pp. 23-37.

[GT] S. W. GOLOMB AND H. TAYLOR, *Two-dimensional synchronization patterns for minimum ambiguity*, IEEE Trans. Inform. Theory, 28 (1982) pp. 600-604.

[GP] R. L. GRAHAM AND H. O. POLLAK, *On the addressing problem for loop switching*, Bell Systems Tech. J., 50 (1971), pp. 2495-2519.

[GS1] R. L. GRAHAM AND N. J. A. SLOANE, *On additive bases and harmonious graphs*, this Journal, 4 (1980), pp. 382-404.

[GS2] ———, *On constant weight code and harmonious graphs*, in Proc. West Coast Conference on Combinatorics, Graph Theory and Computing, Utilitas Mathematica, Winnipeg, 1980, pp. 25-40.

[HoK] C. HOEDE AND H. KUIPER, *All wheels are graceful*, in Congressus Numerantium 14, Utilitas Mathematica, Winnipeg, 1978, p. 311.

[H1] D. F. HSU, *Cyclic Neofields and Combinatorial Designs*, Lectures Notes in Mathematics 82, Springer-Verlag, Berlin, 1980.

[H2] ———, *Harmonious labellings of windmill graphs and related graphs*, J. Graph Theory, 6 (1982), pp. 85-87.

[HK1] D. F. HSU AND A. D. KEEDWELL, *Generalized complete mappings, neofields, sequenceable groups, and block designs*, I, Pacific J. Math., 111 (1984), pp. 317-332.

[HK2] ———, *Generalized complete mappings, neofields, sequenceable groups, and block designs*, II, Pacific J. Math., to appear.

[Ke] A. D. KEEDWELL, *Sequenceable groups: A survey*, in Finite Geometries and Designs. Cambridge Univ. Press, Cambridge, 1981, pp. 205-215.

[RP] J. H. RABINOWITZ AND V. K. PROULX, *An asymptotic approach to the channel assignment problem*, this Journal, this issue, pp. 507-518.

[Ro] A. ROSA, *On certain valuations of the vertices of a graph*, in Theory of Graphs, P. Rosenstiehl, ed., Dunod, Paris, 1967, pp. 349-355.

[St] T. STORER, *Cyclotomy and Difference Sets*, Markham, Chicago, 1967.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 3, July 1985

018

PERMUTATION FACTORIZATION ON STAR-CONNECTED NETWORKS OF BINARY AUTOMATA*

MAURICE TCHUENTE†

Abstract. In this paper it is shown that, in the worst case, the delay necessary to permute boolean variables on a star-connected network of n binary automata is $n-1$.

AMS(MOS) subject classification. 94C

1. Introduction. Let $G=(V, U)$ be a graph of order n with $V=\{1, 2, \cdots, n\}$ as set of vertices, $U \subset V \times V$ as set of arcs, and let X be a finite nonempty set. A *network of automata* $N=(G, X)$ is defined by associating with any vertex $i \in V$, an automaton A_i with X as state-set and whose inputs are the states of automata A_j such $(j, i) \in U$. ([2], [3]).

Clearly, a mapping $F=(f_1, \cdots, f_n) \in A_n(X): X^n \to X^n$ is a possible *transition-function* of the network if and only if, for any $i \in V$, f_i does not depend on variables x_j such that $j \neq i$ and $(j, i) \notin U$. A function $\Phi \in A_n(X)$ is said to be *computable* on N if it can be decomposed into the form $\Phi = F_p \circ F_{p-1} \circ \cdots \circ F_1$ where any F_i is a transition function of N; $l(\Phi)$ denotes the minimum length of such a factorization and can be interpreted as the time necessary to compute Φ on N.

Example. $G=(V, U)$, $V=\{1, 2, 3\}$, $U=\{(1, 2), (2, 1), (2, 3), (3, 2)\}$, $X=\{0, 1\}$. (See Fig. 1.) The transition-functions of $N=(G, X)$ are of the form

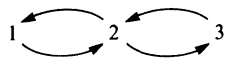


FIG. 1

$$F(x_1, x_2, x_3)=(f_1(x_1, x_2), f_2(x_1, x_2, x_3), f_3(x_2, x_3)).$$

The function $P(x_1, x_2, x_3)=(x_3, x_2, x_1)$ can be factorized as

$$P=F_1 \circ F_2 \circ F_1 \quad \text{where } F_1(x)=(x_2, x_1, x_3) \text{ and } F_2(x)=(x_1, x_3, x_2)$$

or

$$P=F_2' \circ F_1' \quad \text{where } F_1'(x)=(x_1+x_2, x_1+x_2+x_3, x_2+x_3)$$

$$\text{and} \quad F_2'(x)=(-x_1+x_2, x_1-x_2+x_3, x_2-x_3)$$

It is easily verified that $l(P)=2$.

2. Permutation factorization. Let us consider a situation where $\sigma \in S_n$ is a permutation of order n and any automaton A_i wants to send a message to $A_{\sigma(i)}$. Clearly, the transmission of the collection of messages represented by σ, can be modelized as the factorization of the mapping $P_\sigma: (x_1, \cdots, x_n) \mapsto (x_{\sigma(1)}, \cdots, x_{\sigma(n)})$.

In this paper, we study the particular case where $X=\{0, 1\}$ and G is a star (see Fig. 2).

In the sequel, any function $f: \{0, 1\}^n \to \{0, 1\}$ is identified with its unique reduced representative polynomial over the ring of integers modulo 2.

* Received by the editors July 6, 1983.
† CNRS-IMAG Laboratoire TIM3, BP 68 38402, Saint Martin d'Hères Cédex, France.

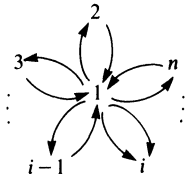


FIG. 2

LEMMA 1. *If*

$$\sigma = (1, a_2, \cdots, a_k) \in S_n$$

or

$$\sigma = (a_2, a_3, \cdots, a_k) \in S_n, \sigma(1) = 1$$

then $l(P_\sigma) \leqq k-1$.

Proof.

Case 1. $\sigma = (1, a_2, \cdots, a_k)$. Clearly,

$$P_\sigma = P_{\sigma^{(k)}} P_{\sigma^{(k-1)}} \cdots P_{\sigma^{(2)}} \quad \text{where } \sigma^{(i)} = (1, a_i), \quad i = 1, \cdots, k.$$

Hence $l(P_\sigma) \leqq k-1$.

Case 2. $\sigma = (a_2, a_3, \cdots, a_k)$, $\sigma(1) = 1$. Because of symmetry considerations, we can assume that $\sigma = (2, 3, \cdots, k)$. Let

$$F_1(x) = (x_1 + x_2 + x_k, x_1 + x_2, x_3, \cdots, x_{k-1}, x_1 + x_k, x_{k+1}, \cdots, x_n),$$

$$F_2(x) = (x_3 + x_k, x_1 - x_2, x_1 + x_3, x_4, \cdots, x_n),$$

$$F_i(x) = (x_{i+1} + x_k, x_2, \cdots, x_{i-1}, x_i - x_1, x_1 + x_{i+1}, x_{i+2}, \cdots, x_n), \qquad i \geqq 3.$$

It is easily verified that for $i \geqq 2$

$$F_i \circ F_{i-1} \circ \cdots \circ F_1(x)$$

$$= (x_1 + x_{i+1} + x_k, x_k, x_2, \cdots, x_{i-1}, x_1 + x_i + x_{i+1} + x_k,$$

$$x_{i+2}, \cdots, x_{k-1}, x_1 + x_k, x_{k+1}, \cdots, x_n) \quad \text{for } 2 \leqq i \leqq k-2,$$

hence

$$F_{k-2} \circ \cdots \circ F_1(x) = (x_1 + x_{k-1} + x_k, x_k, x_2, \cdots, x_{k-3},$$

$$x_1 + x_{k-2} + x_{k-1} + x_k, x_1 + x_k, x_{k+1}, \cdots, x_n).$$

Clearly, if

$$F_{k-1}(x) = (x_k - x_2, x_2, \cdots, x_{k-2}, x_{k-1} - x_1, x_1 - x_k, x_{k+1}, \cdots, x_n)$$

then $F_{k-1} \circ F_{k-2} \circ \cdots \circ F_1 = P_\sigma$; hence $l(P) \leqq k-1$.

PROPOSITION 2. *For any* $\sigma \in S_n$, $l(P) \leqq n-1$.

Proof (follows directly from Lemma 1).

Case 1. $\sigma = (1, a_2, \cdots, a_r)(b_1, b_2, \cdots, b_s) \cdots (z_1, z_2, \cdots, z_t)$.

$$l(P) \leqq l(P_{(1, a_2, \cdots, a_r)}) + \cdots + l(P_{(z_1, \cdots, z_t)})$$

$$\leqq r - 1 + s + \cdots + t$$

$$\leqq n - 1.$$

Case 2. $\sigma(1)=1, \sigma=(a_1,\cdots,a_r)\cdots(z_1,\cdots,z_t)$

$$l(P)\leqq l(P_{(a_1,\cdots,a_r)})+\cdots+l(P_{(z_1,\cdots,z_s)})$$
$$\leqq r+\cdots+t$$
$$\leqq n-1.$$

Let $\sigma\in S_n$ and $P_\sigma=F_q\circ F_{q-1}\circ\cdots\circ F_1$. Since P_σ is bijective it follows that any F_i is bijective. Therefore, in order to obtain a lower bound for $l(P_\sigma)$, it is necessary to study the structure of mappings $F\in A_n(X)$ which are bijective and compatible with a star.

LEMMA 3. *If $F=(f_1,\cdots,f_n)\in A_n(X)$ is bijective and compatible with a star, then any F_i, $i\geqq 2$ is affine.*

Proof. Since, for $i\geqq 2$, f_i is a function of x_1 and x_i, its reduced representative polynomial is of the form

$$P_i=a+bx_1+cx_i+dx_1x_i,\qquad a,b,c,d\in\{0,1\}.$$

If $d=1$ then it is easily verified that card $(f_i^{-1}(0))\neq 2^{n-1}$ and this contradicts the fact that F is bijective, hence $d=0$ and f_i is affine.

PROPOSITION 4. *If $\sigma=(a_2,\cdots,a_n)$, $\sigma(1)=1$ then $l(P_\sigma)\geqq n-1$.*

Proof. Let $P=F_q\circ F_{q-1}\circ\cdots\circ F_1$ where

$$F_i=(f_{i,1},\cdots,f_{i,n})\in A_n(x),\qquad i=1,\cdots,n$$

be a factorization of minimum length and let us denote

$$F_j\circ F_{j-1}\circ\cdots\circ F_1=(g_{j,1},g_{j,2},\cdots,g_{j,n}),\qquad j=1,\cdots,q.$$

Since any $f_{j,k}$, $1\leqq j\leqq q$, $2\leqq k\leqq n$ is affine (cf. Lemma 3), it is easily verified that any $g_{j,k}$, $1\leqq j\leqq q$, $2\leqq k\leqq n$ is of the form

$$g_{j,k}=\alpha_{j,k}x_1+\beta_{j,k}x_k+\sum_{i=1}^{q-1}a_{j,k}^{(i)}g_{i,1}.$$

Since $F_q\circ F_{q-1}\circ\cdots\circ F_1=(g_{q,1},\cdots,g_{q,n})=P_\sigma$ we can write

$$u_2=x_n-\alpha_{q,2}x_1-\beta_{q,2}x_2=\sum_{i=1}^{q-1}a_{q,2}^{(i)}g_{i,1},$$

$$u_3=x_2-\alpha_{q,3}x_1-\beta_{q,3}x_3=\sum_{i=1}^{q-1}a_{q,3}^{(i)}g_{i,1},$$

$$\vdots$$

$$u_n=x_{n-1}-\alpha_{q,n}x_1-\beta_{q,n}x_n=\sum_{i=1}^{q-1}a_{q,n}^{(i)}g_{i,1}.$$

In the vector-space of polynomials over X, the set of elements $\{u_2,\cdots,u_n\}$, when decomposed with respect to the independent system $\{x_1,x_2,\cdots,x_n\}$, yields the matrix

$$M=\begin{bmatrix}-\alpha_{q,2} & -\alpha_{q,3} & & -\alpha_{q,n}\\ -\beta_{q,2} & 1 & & \\ & -\beta_{q,3} & & \\ & & & \\ & & & 1\\ 1 & & & -\beta_{q,n}\end{bmatrix}$$

and it is easily verified that rank $(M) \geqq n-2$. Since any u_i, $2 \leqq i \leqq n$ is generated by $\{g_{i,1}, 1 \leqq i \leqq q\}$, it follows that $q-1 \geqq n-2$; hence $l(P_\sigma) = q \geqq n-1$. Proposition 2 and Proposition 4 can be summarized as follows.

THEOREM 5. $\max_{\sigma \in S_n} l(P_\sigma) = n-1$.

3. Conclusion. In the literature, the most widely used criteria in the study of interconnection patterns associated with finite networks of automata, is the concept of diameter, which represents the maximum number of links to be used to transmit a message [1]. Clearly, if one wants to take into account the potential parallelism of a network, then the permutation-factorization concept studied here, is more appropriate.

Let $l'(F)$ denote the minimum factorization length of F, when the transition functions of the network $N = (G, X)$, are restricted to be of the form

$$F(x_1, \cdots, x_n) = (x_{\theta(1)}, \cdots, x_{\theta(n)}), \qquad \theta \in S_n.$$

In [4] it has been proved that

- If G is a star of order n, then $\max_{\sigma \in S_n} l'(P_\sigma) = \lfloor 3(n-1)/2 \rfloor$ where $\lfloor k \rfloor$ denotes the greatest integer lower than or equal to k.
- If G is a tree of order n, then $\max_{\sigma \in S_n} l'(P) \geqq n$ and the lower bound is obtained when G is a chain (see Fig. 3). These results, together with the main theorem of this

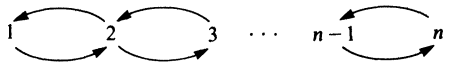


FIG. 3

paper show that, with respect to permutation-factorization criteria, decentralized (chain-connected) networks are better than centralized (star-connected) ones.

REFERENCES

[1] J. C. BERMOND, C. DELORME AND J. J. QUISQUATER, *Tables of large graphs with given degree and diameter*, Inform. Processing Lett., 15 (1982), pp. 10-13.

[2] J. R. JUMP AND J. S. KIRTANE, *On the interconnection structure of cellular networks*, Inform. and Control, 24 (1974), pp. 74-91.

[3] J. V. NEUMANN, *Theory of Self Reproducing Automata*, A. W. Burks, ed., Univ. Illinois Press, Urbana, 1966.

[4] M. TCHUENTE, *Parallel realization of permutations over tree*, Discrete Math., 39 (1982), pp. 211-214.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 4, October 1985

THE FIELD OF VALUES OF A COMPLEX MATRIX, AN EXPLICIT DESCRIPTION IN THE 2×2 CASE*

FRANK UHLIG†

Dedicated to Emilie Haynsworth

Abstract. It is shown that the field of values of a matrix $A \in C_{22}$ is an ellipse with center tr $A/2$, major axis parallel to the unit vector

$$\sqrt{\frac{-\det A_0}{|\det A_0|}}$$

with length $(\|A_0\|^2+2|\det A_0|)^{1/2}$ and minor axis of length $(\|A_0\|^2+2|\det A_0|)^{1/2}$, where $\|A\|$ denotes the Schur-norm and $A_0 := A-(\operatorname{tr} A/2)I$. Moreover, a real symmetric matrix S is given explicitly, for which the quadratic form

$$\left(x-\operatorname{Re}\frac{\operatorname{tr} A}{2}, y-\operatorname{Im}\frac{\operatorname{tr} A}{2}\right) S \begin{pmatrix} x-\operatorname{Re}\dfrac{\operatorname{tr} A}{2} \\ y-\operatorname{Im}\dfrac{\operatorname{tr} A}{2} \end{pmatrix} = \frac{1}{4}\det S$$

describes the points (x, y) on the boundary of $W(A) \subseteq C \simeq R^2$.

AMS(MOS) subject classification. 15A60

The field of values

$$W(A) := \{x^*Ax \in C \mid x \in C^n, x^*x = 1\}$$

of a matrix $A \in C_{nn}$ has been studied for over 60 years. $W(A)$ was first proved to be convex by Toeplitz [7] and Hausdorff [8]. $W(A)$ contains all eigenvalues of A and so forth. The fact that $W(A)$ is an ellipse in case $n=2$ is often used to prove convexity of $W(A)$ for arbitrary n. But for $A = \left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right) \in C_{22}$, an explicit description of this ellipse $W(A)$ in terms of the elements $a, b, c, d \in C$ is missing from the literature so far.

Our aim here is to derive an explicit description of the ellipse $W(A)$ for $A \in C_{22}$ in terms of its center, direction and lengths of its half axis as well as an algebraic equation for its boundary curve. Analogous formulas for $A \in R_{22}$ were developed by C. R. Johnson [4], but unfortunately they can not be generalized to $A \in C_{22}$ in any obvious way, as will be seen later.

The formulas for $W(A)$ can best be derived in two steps: First we form $A_0 = A-(\operatorname{tr} A/2)I$ whose field of values $W(A_0)$ is congruent to $W(A)$ and is centered at $0 \in C$ since $\operatorname{tr} A_0 = 0$. Then we rotate A_0 by an angle $-\alpha$ so that $A_1 := e^{-i\alpha}A_0$ has an ellipse as field of values whose major axis lies on the real line in C. The length of the major and minor axes of $W(A_1)$ can be determined partially in the same way as Johnson [4] has done for 2×2 real A, namely via the eigenvalues of $\operatorname{Re} A_1$ and the eccentricity equation relating the length of the two half axis and the focal distance (or eigenvalues of A_1 in our case) for ellipses. We have $A_0 = A-(\operatorname{tr} A/2)I$ and $A_1 = e^{-i\alpha}A_0$, hence $A_1 = e^{-i\alpha}(A-(\operatorname{tr} A/2)I)$ or $A = e^{i\alpha}A_1+(\operatorname{tr} A/2)I$. Thus due to the standard properties for the field of values, $W(A) = e^{i\alpha}W(A_1)+\operatorname{tr} A/2 \subseteq C$, i.e. $W(A)$ is an ellipse

* Received by the editors April 19, 1983, and in revised form April 9, 1984, This paper was presented at the SIAM Conference on Applied Linear Algebra, Raleigh, North Carolina, April 26-29, 1982.

† Department of Mathematics, Auburn University, Auburn, Alabama 36849.

in the complex plane with center tr $A/2$ of the same shape as the ellipse $W(A_1)$, but rotated back by the angle α with respect to the real line in C.

In order to determine the rotation angle α, we need to know the eigenvalues of A_0:

LEMMA 1. *A_0 has eigenvalues $\lambda_{1,2} = \pm(|\det A_0|)^{1/2} e^{i\alpha}$ for*

$$e^{i\alpha} = \sqrt{\frac{-\det A_0}{|\det A_0|}}.$$

Proof. Since $\operatorname{tr} A_0 = 0 = \lambda_1 + \lambda_2$, $\det A_0 = \lambda_1 \cdot \lambda_2 = -\lambda_1^2$. Hence

$$\lambda_{1,2} = \sqrt{-\det A_0} = \pm\sqrt{|\det A_0|}\sqrt{\frac{-\det A_0}{|\det A_0|}}. \qquad \square$$

If we rotate $W(A_0)$ by the angle $-\alpha$ around its center 0, then $A_1 := e^{-i\alpha}A_0$ has eigenvalues $\mu_i = e^{-i\alpha}\lambda_i = \pm(|\det A_0|)^{1/2} \in R$. The ellipse $W(A_1)$ can now be completely described in terms of the eigenvalues $\pm\nu$ of $\operatorname{Re} A_1$ and the eigenvalues μ_i of A_1 via the eccentricity equation $\chi^2 = \nu^2 - \mu_i^2$ (see Fig. 1). Note that $\operatorname{Re} A_1 = (A_1 + A_1^*)/2$ has real eigenvalues $\pm\nu$ which give the length of the major half axis of $W(A_1)$.

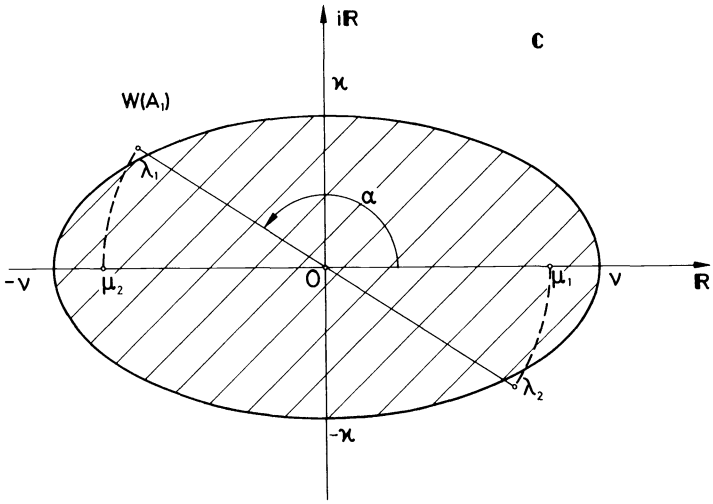


FIG. 1

In fact we have

LEMMA 2. *$\operatorname{Re} A_1$ has eigenvalues $\pm\nu = \pm\frac{1}{2}(\|A_0\|^2 + 2|\det A_0|)^{1/2}$, where*

$$\|A\| := \sqrt{|a|^2 + |b|^2 + |c|^2 + |d|^2}$$

denotes the Schur norm of

$$A = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \in C_{22}.$$

Proof. We have

$$\operatorname{Re} A_1 = \frac{A_1 + A_1^*}{2} \begin{pmatrix} \dfrac{e^{-i\alpha}(a-d) + e^{i\alpha}(\overline{a-d})}{4} & \dfrac{e^{-i\alpha}b + e^{i\alpha}\bar{c}}{2} \\ \dfrac{e^{-i\alpha}c + e^{i\alpha}\bar{b}}{2} & \dfrac{e^{-i\alpha}(d-a) + e^{i\alpha}(\overline{d-a})}{4} \end{pmatrix}.$$

Now $\det \operatorname{Re} A_1 = -\nu^2$ or

$$\begin{aligned}
\nu^2 &= -\det \operatorname{Re} A_1 \\
&= \frac{|(a-d)+e^{2i\alpha}(\overline{a-d})|^2}{16} + \frac{|b+e^{2i\alpha}\bar{c}|^2}{4} \\
&= \frac{2|a-d|^2+e^{-2i\alpha}(a-d)^2+e^{2i\alpha}(\overline{a-d})^2}{16} + \frac{|b|^2+e^{-2i\alpha}bc+e^{2i\alpha}\overline{bc}+|c|^2}{4} \\
&= \frac{1}{4}\left(2\left|\frac{a-d}{2}\right|^2+|b|^2+|c|^2+e^{-2i\alpha}\left(\left(\frac{a-d}{2}\right)^2+bc\right)+e^{2i\alpha}\left(\left(\frac{\overline{a-d}}{2}\right)^2+\overline{bc}\right)\right) \\
&= \frac{1}{4}\left(\|A_0\|^2+\frac{\overline{\det A_0}}{|\det A_0|}\det A_0+\frac{\det A_0}{|\det A_0|}\det A_0\right) \\
&= \frac{1}{4}(\|A_0\|^2+2|\det A_0|) \in R_+. \qquad \square
\end{aligned}$$

Finally for the minor axis:

LEMMA 3. *The minor axis of $W(A_1)$ has length $\chi = \frac{1}{2}(\|A_0\|^2 - 2|\det A_0|)^{1/2}$.*

Proof.

$$\begin{aligned}
\chi^2 = \nu^2 - \mu_i^2 &= \tfrac{1}{4}(\|A_0\|^2+2|\det A_0|) - |\det A_0| \\
&= \tfrac{1}{4}(\|A_0\|^2 - 2|\det A_0|). \qquad \square
\end{aligned}$$

Thus we have

THEOREM 1. *Let $A = \left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right) \in C_{22}$. Then the field of values $W(A)$ of A is an ellipse in the complex plane centered at $\operatorname{tr} A/2 \in C$. The major axis of this ellipse forms an angle α with the real line in C, where*

$$e^{i\alpha} = \sqrt{\frac{-\det A_0}{|\det A_0|}}, \qquad 0 \leqq \alpha < 2\pi, \qquad A_0 = A - \frac{\operatorname{tr} A}{2} I.$$

The major axis has length $2\nu = (\|A_0\|^2+2|\det A_0|)^{1/2}$, while the minor axis has length $2\chi = (\|A_0\|^2 - 2|\det A_0|)^{1/2}$.

Next we want to compare our formulas with those of Johnson [4] for $A \in R_{22}$. For simplicity, we assume that $\operatorname{tr} A = 0$, so that in Johnson's notation [4, Thm., p. 105], the major half-axis has length

$$y := \tfrac{1}{2}(a+d+((a-d)^2+(b+c)^2)^{1/2}).$$

With $A = \left(\begin{smallmatrix} a & b \\ c & -a \end{smallmatrix}\right) \in R_{22}$, this turns out to be:

$$y := \tfrac{1}{2}(4a^2+(b+c)^2)^{1/2},$$

while the minor half-axis has length $w := |(b-c)/2|$. Since $\operatorname{tr} A = \lambda_1 + \lambda_2 = 0$, $\det A = -a^2 - bc = \lambda_1 \cdot \lambda_2 \leqq 0$. Hence from the theorem,

$$\nu = \tfrac{1}{2}(\|A\|^2+2|\det A|)^{1/2} = \tfrac{1}{2}(2a^2+b^2+c^2+2|-a^2-bc|)^{1/2} = \tfrac{1}{2}(4a^2+(b+c)^2)^{1/2} = y,$$

$$\chi = \tfrac{1}{2}(2a^2+b^2+c^2-2|-a^2-bc|)^{1/2} = \tfrac{1}{2}((b-c)^2)^{1/2} = \frac{|b-c|}{2} = w.$$

Thus our formulas coincide for $A \in R_{22}$.

For complex matrices, Johnson's simpler formulas for y and w cannot be generalized by interpreting x^2 as $\|x\|^2$.

Example. Take $A = \left(\begin{smallmatrix} 0 & i \\ 1 & 0 \end{smallmatrix}\right) \in \in C_{22}$. Then $y = \frac{1}{2}|1+i| = \sqrt{2}/2$, while in fact $\nu = \frac{1}{2}(1+1+2)^{1/2} = 1$.

Finally we give an explicit formula for the boundary of the ellipse $W(A)$ for $A \in C_{22}$.

THEOREM 2. *In the complex plane $C \simeq R^2$, the boundary curve of the field of values of a matrix $A \in C_{22}$ is given by*

$$\left(x - \operatorname{Re}\frac{\operatorname{tr} A}{2}, y - \operatorname{Im}\frac{\operatorname{tr} A}{2}\right) S \begin{pmatrix} x - \operatorname{Re}\dfrac{\operatorname{tr} A}{2} \\ y - \operatorname{Im}\dfrac{\operatorname{tr} A}{2} \end{pmatrix} = \frac{1}{4}\det S,$$

where $\left(\begin{smallmatrix} x \\ y \end{smallmatrix}\right) \in R^2$ and

$$S = \begin{pmatrix} \|A_0\|^2 + 2\operatorname{Re}(\det A_0) & 2\operatorname{Im}(\det A_0) \\ 2\operatorname{Im}(\det A_0) & \|A_0\|^2 - 2\operatorname{Re}(\det A_0) \end{pmatrix}$$

for $A_0 := A - (\operatorname{tr} A/2)I$.

Proof. The boundary curve of $W(A_1)$ for $A_1 = e^{-i\alpha}(A - (\operatorname{tr} A/2)I)$ has the equation $x^2/\nu^2 + y^2/\chi^2 = 1$ for $\left(\begin{smallmatrix} x \\ y \end{smallmatrix}\right) \in R^2$ with ν and χ as in Theorem 1. Since $A_0 = e^{i\alpha}A_1$, $W(A_0)$ is obtained from $W(A_1)$ by a plane rotation for the angle α, and the equation of the boundary of $W(A_0)$ is

$$\chi^2(x\cos\alpha + y\sin\alpha)^2 + \nu^2(-x\sin\alpha + y\cos\alpha)^2 = \nu^2\chi^2.$$

With ν^2 and χ^2 from above and using $\chi^2 = \nu^2 - \mu^2$, this becomes

$$\begin{aligned} &\tfrac{1}{4}(\|A_0\|^2 - 2|\det A_0|(\cos^2\alpha - \sin^2\alpha))x^2 - \sin 2\alpha |\det A_0| xy \\ &\quad + \tfrac{1}{4}(\|A_0\|^2 - 2|\det A_0|(\sin^2\alpha - \cos^2\alpha))y^2 = \tfrac{1}{16}(\|A_0\|^4 - 4|\det A_0|^2). \end{aligned}$$

Now

$$\sin 2\alpha = \operatorname{Im} e^{2i\alpha} = \operatorname{Im}\frac{-\det A_0}{|\det A_0|} \quad \text{and} \quad \cos 2\alpha = \operatorname{Re}\frac{-\det A_0}{|\det A_0|}.$$

Hence the equation for the boundary of $W(A_0)$ is

$$\begin{aligned} &\|A_0\|^2 + 2\operatorname{Re}(\det A_0))x^2 + 4\operatorname{Im}(\det A_0)xy + (\|A_0\|^2 - 2\operatorname{Re}(\det A_0))y^2 \\ &\quad = \tfrac{1}{4}(\|A_0\|^4 - 4|\det A_0|^2). \end{aligned}$$

A translation by

$$\begin{pmatrix} \operatorname{Re}\dfrac{\operatorname{tr} A}{2} \\ \operatorname{Im}\dfrac{\operatorname{tr} A}{2} \end{pmatrix}$$

gives the result. □

The formulas for $W(A)$, $A \in C_{22}$, could have been derived in various other ways, none of which appear as easy computationally as ours. W. Donoghue [2, Lemma] showed that the "reduced angle" between eigenvectors of a 2×2 matrix A determines the eccentricity of $W(A)$. A direct computation of the eigenvectors of A and their reduced angle seems more complicated than our approach. R. Kippenhahn [5] used the determinant function in two variables u and v, $\det(uH_1 + uH_2 + wI) = 0$, involving the real and imaginary parts H_1 and H_2 of A for defining the generating lines of the field of values $W(A)$ for any $A \in C_{nn}$ (or $A \in H_{nn}$, the quaternion matrix algebra). This

formula was recently refined by M. Fiedler [3, Thm. 2.4, p. 87, 88] to involve the second compounds of H_i and their mixed second compound, and Fiedler gives an equation for the boundary curve of $W(A)$ in any dimension. While truly more general in scope, these two formulas unfortunately are too cumbersome in case $n = 2$ already to be used for computing $W(A)$ any faster than we did.

C. S. Ballantine [1] recently showed how to check whether a point $z \in C$ lies in $W(A)$ for any $A \in C_{nn}$. Our formulas can be used to obtain an "inner approximation" of $W(A)$ for $A \in C_{nn}$ in much the same way as Johnson [4, Cor. 2] has done for $A \in R_{nn}$. Another application of our results is given in F. Uhlig [6]. For another recent reference see J. M. Patel [9].

REFERENCES

[1] C. S. BALLANTINE, *Numerical range of a matrix, some effective criteria*, Lin. Alg. Appl., 19 (1978), pp. 117-188.

[2] W. F. DONOGHUE, JR., *On the numerical range of bounded operators*, Mich. Math. J., 4 (1957), pp. 261-267.

[3] M. FIEDLER, *Geometry of the numerical range of matrices*, Lin. Alg. Appl., 37 (1981), pp. 81-96.

[4] C. R. JOHNSON, *Computation of the field of values of a* 2×2 *matrix*, J. Res. Nat. Bur. Standards, 78B (1974), pp. 105-107.

[5] R. KIPPENHAHN, *Uber den Wertevorrat einer Matrix*, Math. Nachr., 6 (1951), pp. 193-228.

[6] F. UHLIG, *Relations between the fields of values of a matrix and of its polar factors, the* 2×2 *real and complex case*, Lin. Alg. Appl., 52 (1983), pp. 701-715.

[7] O. TOEPLITZ, *Das algebraische Analogon zu einem Satze von Fejer*, Math. Z., 2 (1918), pp. 187-197.

[8] F. HAUSDORFF, *Der Wertvorrat einer Bilinearform*, Math. Z., 3 (1919), pp. 314-316.

[9] T. M. PATEL, *On the numerical range of an operator*, Vidya B, 23 (1980), pp. 11-14.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 4, October 1985

EXPLICIT INVERSION FORMULAS FOR TOEPLITZ BAND MATRICES*

WILLIAM F. TRENCH†

Abstract. Explicit formulas are given for the elements of T_n^{-1} and the solution of $T_nX = Y$, where T_n is an $(n+1)\times(n+1)$ Toeplitz band matrix with bandwidth $k \leqq n$. The formulas involve $k\times k$ determinants whose entries are powers of the zeros of a certain kth degree polynomial $P(z)$ which is independent of n, or simple related functions of these zeros if any are repeated. It is shown that T_n is invertible if and only if a certain $k\times k$ determinant involving these zeros is nonvanishing.

AMS(MOS) classification numbers. 15, 65F

1. Introduction.

We consider Toeplitz band matrices, i.e., matrices of the form

$$T_n = (\phi_{r-s})_{r,s=0}^{n},$$

where there are nonnegative integers p and q such that

$$\phi_\nu = 0 \quad \text{if } \nu > p \text{ or } \nu < -q. \tag{1}$$

We use the notation of [13] and [15]. Notice that T_n is of order $n+1$, with rows and columns numbered from 0 to n. We write

$$T_n^{-1} = B_n = (b_{rsn})_{r,s=0}^{n}.$$

It is assumed throughout that

$$\phi_p\phi_{-q} \neq 0 \quad \text{and} \quad p+q = k \leqq n. \tag{2}$$

Our main results are explicit formulas for the elements of T_n^{-1} and for the solution of the system

$$T_nX = Y, \tag{3}$$

in terms of the zeros of the polynomial

$$P(z) = \sum_{\mu=-q}^{p} \phi_\mu z^{\mu+q}. \tag{4}$$

These formulas involve determinants of order k, the bandwidth of T_n.

Many authors (e.g., [1], [3], [6], [7], [9], [10], [12], [15]) have given formulas and algorithms for inverting Toeplitz band matrices. Efficient methods have also been developed for solving (3) (e.g., [2], [4], [11], [14]). Since a survey of results along these lines is given in the introduction to the recent paper [9], there is no need to review earlier work here. We believe that the results presented here are new, and more general and explicit than others heretofore published. We treat the general Toeplitz band matrix, without assuming that T_n is symmetric or hermitian. Our formulas are explicit (i.e., not recursive with respect to n), and we do not have to assume that any matrix other than T_n is nonsingular; however, we do give a method for computing T_n^{-1} efficiently in the case where T_{n-1} is also nonsingular.

The idea motivating our approach is that if n is large compared to k, then T_n is "nearly triangular" in an obvious visual sense, which need not be defined precisely.

* Received by the editors December 20, 1983, and in revised form April 19, 1984.

† Department of Mathematics and Computer Science, Drexel University, Philadelphia, Pennsylvania 19104.

Therefore, it is not surprising that the elements of T_n^{-1} are closely related to those of $(T_n^L)^{-1}$, where T_n^L is the lower triangular Toeplitz matrix

$$T_n^L = (\phi_{r-s-q})_{r,s=0}^n.$$

The inverse of this matrix is the lower triangular Toeplitz matrix

$$(T_n^L)^{-1} = (\alpha_{r-s})_{r,s=0}^n \qquad (\alpha_\nu = 0 \text{ if } \nu < 0), \tag{5}$$

with elements independent of n, defined by

$$(P(z))^{-1} = \sum_{\nu=0}^{\infty} \alpha_\nu z^\nu. \tag{6}$$

It is easy to find an explicit formula for α_ν in terms of the zeros of $P(z)$. Moreover, we will show that the differences

$$b_{rsn} - \alpha_{r-s-q}, \qquad 0 \leqq r, s \leqq n$$

can be found easily and explicitly in terms of the zeros of $P(z)$. This leads to explicit formulas for T_n^{-1} and the solution of (3).

We also give analogous formulas based on the inverse of the upper triangular Toeplitz matrix

$$T_n^U = (\phi_{r-s+p})_{r,s=0}^n.$$

The inverse of this matrix is

$$(T_n^U)^{-1} = (\beta_{s-r})_{r,s=0}^n \qquad (\beta_\nu = 0 \text{ if } \nu < 0), \tag{7}$$

with elements independent of n, defined by

$$\left(z^k P\left(\frac{1}{z}\right)\right)^{-1} = \sum_{\nu=0}^{\infty} \beta_\nu z^\nu. \tag{8}$$

2. Preliminary results. The following assumption applies throughout. (Recall (2) here.)

Assumption A. The distinct zeros of (4) are $z_1, \cdots, z_m$, with multiplicities $\mu_1, \cdots, \mu_m$; thus, $m \leqq k$, $\mu_i \geqq 1$, and

$$\mu_1 + \cdots + \mu_m = k.$$

DEFINITION 1. If $j_1, \cdots, j_k$ are integers, let

$$C(z; j_1, \cdots, j_k) = \text{col}\,[z^{j_1}, \cdots, z^{j_k}],$$

and let $C^{(l)}(z; j_1, \cdots, j_k)$ denote the lth derivative of this column vector. Now define the $k \times k$ determinant $D(j_1, \cdots, j_k)$ as follows: Its first μ_1 columns are $C^{(l)}(z_1; j_1, \cdots, j_k)(0 \leqq l \leqq \mu_1 - 1)$; it next μ_2 columns are $C^{(l)}(z_2; j_1, \cdots, j_k)(0 \leqq l \leqq \mu_2 - 1)$; and so forth.

For example, if (4) has k distinct roots, then

$$D(j_1, \cdots, j_k) = \det\,(z_s^{j_r})_{r,s=1}^k.$$

There is an ambiguity in Definition 1, since the m zeros of $P(z)$ may be numbered in any order; however, our results involve ratios of the form

$$D(j_1, \cdots, j_k)/D(j_1', \cdots, j_k'),$$

which are left invariant if $z_1, \cdots, z_m$ are permuted. Because of (2), $z_j \neq 0$ $(1 \leqq j \leqq m)$,

so $D(j_1, \cdots, j_k)$ exists for all $j_1, \cdots, j_k$. It can be shown that

$$D(0, 1, \cdots, k-1) = K \prod_{1 \leqq i < j \leqq m} (z_j - z_i)^{r_{ij}},$$

where $K > 0$ and the r_{ij}'s are positive integers. If $m = k$, then $D(0, 1, \cdots, k-1)$ is the Vandermonde determinant, so $K = r_{ij} = 1$.

LEMMA 1. *Suppose* $1 \leqq l \leqq k$ *and* $j_1, \cdots, j_{l-1}, j_{l+1}, \cdots, j_k$ *are fixed integers. Then the sequence*

$$e_r = D(j_1, \cdots, j_{l-1}, r, j_{l+1}, \cdots, j_k), \qquad -\infty < r < \infty \tag{9}$$

satisfies the difference equation

$$\sum_{\nu=-q}^{p} \phi_\nu e_{\nu+r} = 0. \tag{10}$$

Proof. Because of Definition 1, expanding the determinant in (9) in terms of cofactors of its lth row yields

$$e_r = \sum_{j=1}^{m} \sum_{i=0}^{\mu_j - 1} a_{ij} (r)^{(i)} z_j^{r-i}, \tag{11}$$

where

$$(r)^{(0)} = 1, \qquad (r)^{(i)} = r(r-1) \cdots (r-i+1), \quad i \geqq 1,$$

and the a_{ij}'s are constants. But if $1 \leqq j \leqq m$ and $0 \leqq i \leqq \mu_j - 1$, then

$$\sum_{\nu=-q}^{p} \phi_\nu (\nu + r)^{(i)} z_j^{\nu+r-i} = 0, \qquad -\infty < r < \infty,$$

since z_j is a zero of $z^{r-q}P(z)$, with multiplicity μ_j, for every r. This and (11) imply (10).

LEMMA 2 a) *The sequence* $\{\alpha_r\}$ *defined by*

$$\begin{aligned} &\text{(a)}\ \ \alpha_r = 0, \qquad r < 0, \\ &\text{(b)}\ \ \alpha_r = \frac{1}{\phi_{-q}} \frac{D(-r, 1, \cdots, k-1)}{D(0, 1, \cdots, k-1)}, \qquad r \geqq -k+1, \end{aligned} \tag{12}$$

satisfies

$$\sum_{\nu=-q}^{p} \phi_\nu \alpha_{j-q-\nu} = \delta_{j0}, \qquad -\infty < j < \infty. \tag{13}$$

b) *The sequence* $\{\beta_r\}$ *defined by*

$$\begin{aligned} &\text{(a)}\ \ \beta_r = 0, \qquad r < 0, \\ &\text{(b)}\ \ \beta_r = \frac{1}{\phi_p} \frac{D(0, 1, \cdots, k-2, r+k-1)}{D(0, 1, \cdots, k-1)}, \qquad r \geqq -k+1, \end{aligned} \tag{14}$$

satisfies

$$\sum_{\nu=-q}^{p} \phi_\nu \beta_{j-p+\nu} = \delta_{j0}, \qquad -\infty < j < \infty. \tag{15}$$

(*Note.* The definitions (12) and (14) are redundant, but consistent, for $-k+1 \leqq j \leqq -1$. They are stated this way for convenience.)

Proof. (a) If $j<0$, then (12a) implies (13). If $j=0$, then (13) reduces to

$$\phi_{-q}\alpha_0=1,$$

again because of (12a). This is consistent with (12b) with $r=0$. If $j\geqq 1$, then (12b) and Lemma 1 imply (13).

(b) Similar proof.

Notice that the sequences $\{\alpha_r\}$ and $\{\beta_r\}$ can be computed recursively from (13) and (15), or explicitly from (12) and (14).

Lemma 2 implies (5), (6), (7), and (8). Therefore, (12) provides an explicit formula for T_n^{-1} if $q=0$ (T_n is lower triangular), while (14) serves the same purpose if $p=0$ (T_n is upper triangular). Of course, the inversion of triangular Toeplitz matrices—banded or not—is very simple, as was observed in [15]. We assume henceforth that

$$p\geqq 1 \quad \text{and} \quad q\geqq 1. \tag{16}$$

3. The main results. The next theorem follows from a result in [16] concerning the eigenvalues of Toeplitz band matrices; however, since the proof in [16] utilizes a more involved argument than is needed here, it is convenient to prove Theorem 1 directly.

THEOREM 1. *If* (2) *and* (16) *hold, then* T_n *is invertible if and only if*

$$D(0,1,\cdots,q-1,n+q+1,\cdots,n+k)\neq 0. \tag{17}$$

Proof. We prove the equivalent assertion that the system

$$T_n'X=0 \qquad (' = \text{transpose}) \tag{18}$$

has a nontrivial solution $X=\operatorname{col}[x_0,\cdots,x_n]$ if and only if

$$D(0,1,\cdots,q-1,n+q+1,\cdots,n+k)=0. \tag{19}$$

Easy manipulations show that (18) holds if and only if the finite sequence

$$x_{-q},\cdots,x_{-1},x_0,\cdots,x_n,x_{n+1},\cdots,x_{n+p}$$

satisfies the boundary value problem

$$\begin{aligned} &\text{(a)} \quad \sum_{\nu=-q}^{p}\phi_\nu x_{\nu+r}=0, \qquad 0\leqq r\leqq n,\\ &\text{(b)} \quad x_r=0 \quad \text{if } -q\leqq r\leqq -1 \text{ or } n+1\leqq r\leqq n+p. \end{aligned} \tag{20}$$

However, because of Assumption A and the fact that $z_j\neq 0$ $(1\leqq j\leqq m)$, the elementary theory of constant coefficient difference equations implies that a solution of (20a) must be of the form

$$x_r=\sum_{j=1}^{m}\sum_{i=0}^{\mu_j-1}a_{ij}(q+r)^{(i)}z_j^{q+r-i}, \qquad -q\leqq r\leqq n+p, \tag{21}$$

where

$$A=\operatorname{col}[a_{01},\cdots,a_{\mu_1-1,1},\cdots,a_{0m},\cdots,a_{\mu_m-1,m}]$$

is a constant vector. On recalling Definition 1, it can be seen that (21) is consistent with (20b) if and only if A satisfies the $k\times k$ system

$$HA=0, \tag{22}$$

where

$$\det H = D(0, 1, \cdots, q-1, n+q+1, \cdots, n+k).$$

Therefore (22) has a nontrivial solution, and the same is true of (18), if and only if (19) holds. This completes the proof.

Henceforth, we assume that (17) holds.

DEFINITION 2. Let

$$U_n = \{0, 1, \cdots, q-1, n+q+1, \cdots, n+k\}.$$

If $\mu \in U_n$ and l is an arbitrary integer, define

$$a_n(\mu | l) = \frac{D(j_0, \cdots, j_{q-1}, j_{n+q+1}, \cdots, j_{n+k})}{D(0, 1, \cdots, q-1, n+q+1, \cdots, n+k)},$$

where

$$j_i = \begin{cases} i & \text{if } i \in U_n - \{\mu\}, \\ l & \text{if } i = \mu. \end{cases}$$

For example,

$$a_n(0 | l) = \frac{D(l, 1, \cdots, q-1, n+q+1, \cdots, n+k)}{D(0, 1, \cdots, q-1, n+q+1, \cdots, n+k)}$$

and

$$a_n(n+q+1 | l) = \frac{D(0, 1, \cdots, q-1, l, n+q+2, \cdots, n+k)}{D(0, 1, \cdots, q-1, n+q+1, \cdots, n+k)}.$$

Lemma 1 and Definition 2 imply the following lemma.

LEMMA 3. *If μ is a fixed integer in U_n, then*

$$\sum_{\nu=-q}^{p} \phi_\nu a_n(\mu | \nu + r) = 0, \qquad -\infty < r < \infty,$$

and

$$a_n(\mu | r) = \delta_{\mu r}, \qquad r \in U_n;$$

i.e., $e_r = a_n(\mu | r)$ is the unique solution of (10) *which satisfies the boundary conditions*

$$e_r = \delta_{\mu r}, \qquad 0 \leqq r \leqq q-1, \quad n+q+1 \leqq r \leqq n+k.$$

The uniqueness assertion of Lemma 3 follows from (17), as can be seen from the proof of Theorem 1, since the difference of two solutions would satisfy (20).

The next two theorems give explicit formulas for b_{rsn}, the general element of T_n^{-1}. The formula in Theorem 2 is more convenient if $q < p$, while the formula in Theorem 3 is more convenient if $p > q$.

THEOREM 2. *The general element b_{rsn} of $B_n = T_n^{-1}$ is given by*

$$b_{rsn} = \alpha_{r-s-q} - \sum_{l=0}^{q-1} \alpha_{r-l} a_n(l | q+s), \tag{23}$$

where $\{\alpha_r\}$ is as in (12).

Proof. The condition $B_n T_n = I_n$ is equivalent to

$$\sum_{j=0}^{n} b_{rjn} \phi_{j-s} = \delta_{rs}, \qquad 0 \leqq r, s \leqq n.$$

which can be rewritten as

$$\sum_{j=-q}^{n+p} b_{rjn}\phi_{j-s} = \delta_{rs}, \qquad 0 \leqq r, s \leqq n, \tag{24}$$

if we define

$$b_{rsn} = 0 \quad \text{when } -q \leqq s \leqq -1 \text{ or } n+1 \leqq s \leqq n+p. \tag{25}$$

Shifting the index of summation in (24) and recalling (1) yields

$$\sum_{\nu=-q}^{p} \phi_\nu b_{r,\nu+s,n} = \delta_{rs}, \qquad 0 \leqq r, s \leqq n. \tag{26}$$

Now let

$$b_{rsn} = \alpha_{r-s-q} + u_{rsn}, \qquad 0 \leqq r \leqq n, -q \leqq s \leqq n+p, \tag{27}$$

where u_{rsn} is to be determined. From (13) with $j = r-s$,

$$\sum_{\nu=-q}^{p} \phi_\nu \alpha_{r-s-q-\nu} = \delta_{0,r-s} = \delta_{rs}, \qquad 0 \leqq r, s \leqq n;$$

hence, substituting (27) into (26) and recalling (25) shows that for each r in $\{0, \cdots, n\}$, the sequence $\{u_{rsn}\}_{s=-q}^{n+p}$ satisfies the difference equation

$$\sum_{\nu=-q}^{p} \phi_\nu u_{r,\nu+s,n} = 0, \qquad 0 \leqq s \leqq n,$$

and the boundary conditions

$$u_{rsn} = -\alpha_{r-s-q} = 0, \qquad n+1 \leqq s \leqq n+p \quad \text{(cf. (12a))},$$

$$u_{rsn} = -\alpha_{r-s-q}, \qquad -q \leqq s \leqq -1.$$

This and Lemma 3 imply that

$$u_{rsn} = -\sum_{l=0}^{q-1} \alpha_{r-l} a_n(l \,|\, q+s),$$

which, with (27), implies (23).

THEOREM 3. *The general element b_{rsn} of $B_n = T_n^{-1}$ is given by*

$$b_{rsn} = \beta_{s-r-p} - \sum_{l=0}^{p-1} \beta_{s-p+l+1} a_n(n+q+l+1 \,|\, n+q-r), \tag{28}$$

with $\{\beta_r\}$ as defined by (14).

Proof. The condition $T_n B_n = I_n$ is equivalent to

$$\sum_{j=0}^{n} \phi_{r-j} b_{jsn} = \delta_{rs}, \qquad 0 \leqq r, s \leqq n,$$

which can be rewritten as

$$\sum_{j=-p}^{n+q} \phi_{r-j} b_{jsn} = \delta_{rs}, \qquad 0 \leqq r, s \leqq n, \tag{29}$$

if we define

$$b_{rsn} = 0 \quad \text{when } -p \leqq r \leqq -1 \text{ or } n+1 \leqq r \leqq n+q. \tag{30}$$

Changing the index of summation in (29) and recalling (1) yields

$$\sum_{\nu=-q}^{p} \phi_\nu b_{r-\nu,sn} = \delta_{rs}, \qquad 0 \leqq r, s \leqq n. \tag{31}$$

Now let

$$b_{rsn} = \beta_{s-r-p} + v_{rsn}, \tag{32}$$

where v_{rsn} is to be determined. From (15) with $j = s - r$,

$$\sum_{\nu=-q}^{p} \phi_\nu \beta_{s-r-p+\nu} = \delta_{rs}, \qquad 0 \leqq r, s \leqq n;$$

hence, substituting (32) into (31) and recalling (30) shows that for each s in $\{0, \cdots, n\}$, the sequence $\{v_{rsn}\}_{r=-p}^{n+q}$ satisfies the difference equation

$$\sum_{\nu=-q}^{p} \phi_\nu v_{r-\nu,sn} = 0, \qquad 0 \leqq r \leqq n,$$

and the boundary conditions

$$v_{rsn} = -\beta_{s-r-p} = 0, \qquad n+1 \leqq r \leqq n+q,$$

$$v_{rsn} = -\beta_{s-r-p}, \qquad -p \leqq r \leqq -1.$$

This and Lemma 3 imply that

$$v_{rsn} = -\sum_{l=0}^{p-1} \beta_{s-p+l+1} a_n(n+q+l+1 \,|\, n+q-r);$$

which, with (32), implies (28).

The next theorem provides explicit formulas for the solution of (3) when T_n is invertible. Here we write

$$X = \operatorname{col}[x_0, \cdots, x_n] \quad \text{and} \quad Y = \operatorname{col}[y_0, \cdots, y_n]$$

and adopt the convention that

$$\sum_{\mu}^{\nu} = 0 \quad \text{if } \nu < \mu.$$

THEOREM 4. *If T_n is invertible, then the solution of* (3) *is given by*

$$x_r = \sum_{s=0}^{r-q} \alpha_{r-s-q} y_s - \sum_{l=0}^{q-1} \alpha_{r-l} \sum_{s=0}^{n} y_s a_n(l \,|\, q+s), \qquad 0 \leqq r \leqq n, \tag{33}$$

and by

$$x_r = \sum_{s=r+p}^{n} \beta_{s-r-p} y_s - \sum_{l=0}^{p-1} \left(\sum_{s=0}^{n} \beta_{s-p+l+1} y_s \right) a_n(n+q+l+1 \,|\, n+q-r), \qquad 0 \leqq r \leqq n. \tag{34}$$

Proof. Since

$$x_r = \sum_{s=0}^{n} b_{rsn} y_s, \qquad 0 \leqq r \leqq n,$$

(23) implies (33) and (28) implies (34).

Since convolutions can be implemented efficiently by means of fast Fourier transforms, (33) provides an efficient computational method for solving (3). The

quantities

$$M_l = \sum_{s=0}^{n} y_s a_n(l|q+s), \qquad 0 \leqq l \leqq q-1$$

(each of which can be expressed as the ratio of two $k \times k$ determinants) would be computed first. Then, from (33),

$$x_r = -\sum_{l=0}^{r} M_l \alpha_{r-l}, \qquad 0 \leqq r \leqq q-1,$$

and

$$x_r = \sum_{s=0}^{r-q} \alpha_{r-s-q} y_s - \sum_{l=0}^{q-1} M_l \alpha_{r-l}, \quad q \leqq r \leqq n.$$

It is easily verified that (33) and (34) remain valid if $p=0$ or $q=0$.

The next lemma follows trivially from the last four equations of [13].

LEMMA 4. *Suppose T_n is an arbitrary (not necessarily banded) Toeplitz matrix, with inverse $T_n^{-1} = (b_{rsn})_{r,s=0}^{n}$, where*

$$b_{00n} \neq 0. \tag{35}$$

Then the elements $b_{rsn}(1 \leqq r, s \leqq n)$ are determined in terms of $b_{r0n}(0 \leqq r \leqq n)$ and $b_{0sn}(0 \leqq s \leqq n)$ by the recursion formula

$$b_{rsn} = b_{r-1,s-1,n} + (b_{00n})^{-1}(b_{r0n}b_{0sn} - b_{n-s+1,0,n}b_{0,n-r+1,n}), \qquad 1 \leqq r, s \leqq n. \tag{36}$$

Since $b_{00n} = \det T_{n-1}/\det T_n$, (35) implies that T_{n-1} is also invertible. Since T_n^{-1} is persymmetric (i.e., symmetric about its secondary diagonal), it is only necessary to use (36) for $r+s \leqq n$, and then take

$$b_{rsn} = b_{n-s,n-r,n}, \qquad 1 \leqq r, s \leqq n, \quad r+s > n.$$

Lemma 4 was rediscovered and presented in a useful matrix form by Gohberg and Semencul [5]. In most applications (e.g., [3], [8], [13], [15], [16]), it has been coupled with recursive procedures for obtaining the elements of the zeroth row and column of T_n^{-1}; however, these methods usually require that other matrices in the sequence $\{T_0, T_1, \cdots\}$ be nonsingular. Since Theorems 2 and 3 provide convenient explicit formulas for the zeroth row and column of T_n^{-1}, we can dispense with this additional assumption here. Thus, from (12) and (23),

$$b_{0sn} = -(\phi_{-q})^{-1} a_n(0|q+s), \qquad 0 \leqq s \leqq n, \tag{37}$$

while from (14) and (28),

$$b_{r0n} = -(\phi_p)^{-1} a_n(n+k|n+q-r), \qquad 0 \leqq r \leqq n. \tag{38}$$

If evaluating the $k \times k$ determinants in (37) is inconvenient, then it is only necessary to use (37) for $0 \leqq s \leqq p$; define

$$b_{0sn} = 0, \qquad -q+1 \leqq s \leqq -1,$$

and compute recursively:

$$b_{0sn} = -(\phi_p)^{-1} \sum_{\nu=-q}^{p-1} \phi_\nu b_{0,\nu+s-p,n}, \qquad p+1 \leqq s \leqq n.$$

(See Lemma 3.) Similarly, we can use (38) for $0 \leqq r \leqq q$; define

$$b_{r0n} = 0, \qquad -p+1 \leqq r \leqq -1,$$

and compute recursively:

$$b_{r0n} = -(\phi_{-q})^{-1} \sum_{\nu=-p}^{q-1} \phi_{-\nu} b_{r+\nu-q,0,n}, \qquad q+1 \leqq r \leqq n.$$

REFERENCES

[1] E. L. ALLGOWER, *Exact inverses of certain band matrices*, Numer. Math., 21 (1973), pp. 279-284.

[2] G. BECK, *Fast algorithms for the solution of banded Toeplitz sets of linear equations*, Alkal. Mat. Lapok, 8 (1982), pp. 157-176. (In Hungarian, with English summary.)

[3] R. P. BRENT, F. G. GUSTAVSON AND D. Y. Y. YUN, *Fast solution of Toeplitz systems of equations and computation of Padé approximants*, J. Algorithms, 1 (1980), pp. 259-295.

[4] B. W. DICKINSON, *Efficient solution of linear equations with banded Toeplitz matrices*, IEEE Trans. Acoust., Speech, Sig. Proc., ASSP-27 (1979), pp. 421-423.

[5] I. C. GOHBERG AND A. A. SEMENCUL, *On the inversion of finite Toeplitz matrices and their continuous analogs*, Mat. Issled, 2 (1972), pp. 201-233. (In Russian.)

[6] W. D. HOSKINS AND P. J. PONZO, *Some properties of a class of band matrices*, Math. Comp., 26 (1970), pp. 393-400.

[7] A. K. JAIN, *Fast inversion of banded Toeplitz matrices by circular decompositions*, IEEE Trans. Acoust., Speech, Sig. Proc., ASSP-26 (1978), pp. 121-126.

[8] T. KAILATH, A. VIERA AND M. MORF, *Inverses of Toeplitz operators, innovations, and orthogonal polynomials*, SIAM Rev., 20 (1978), pp. 106-119.

[9] D. S. MEEK, *The inverses of Toeplitz band matrices*, Lin. Alg. Appl., 49 (1983), pp. 117-129.

[10] R. P. MENTZ, *On the inverse of some covariance matrices of Toeplitz type*, SIAM J. Appl. Math., 31 (1976), pp. 426-437.

[11] M. MORF AND T. KAILATH, *Recent results in least-square estimation theory*, Ann. Econ. Social Meas., 6 (1977), pp. 261-274.

[12] L. REHNQVIST, *Inversion of certain symmetric band matrices*, Nordisk. Tidskr. Informationsbehandlung (BIT), 12 (1972), pp. 90-98.

[13] W. F. TRENCH, *An algorithm for the inversion of finite Toeplitz matrices*, J. Soc. Indust. Appl. Math., 12 (1964), pp. 515-522.

[14] ———, *Weighting coefficients for the prediction of stationary time series from the finite past*, SIAM J. Appl. Math., 15 (1967), pp. 1502-1510.

[15] ———, *Inversion of Toeplitz band matrices*, Math. Comp., 28 (1974), pp. 1089-1095.

[16] ———, *On the eigenvalue problem for Toeplitz band matrices*, Lin. Alg. Appl., to appear.

[17] S. ZOHAR, *Toeplitz matrix inversion: the algorithm of W. F. Trench*, J. Assoc. Comp. Mach., 16 (1967), pp. 592-601.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 4, October 1985

003

ON THE OPTIMIZATION OF THE CLASSICAL ITERATIVE SCHEMES FOR THE SOLUTION OF COMPLEX SINGULAR LINEAR SYSTEMS*

A. HADJIDIMOS†

Abstract. For the numerical solution of a class of Complex Singular Linear Systems $Ax = b$, with $\det(A) = 0$ and b in the range of A, the generalized iterative methods of Extrapolated Jacobi (JOR) and of Successive Overrelaxation (SOR), first introduced by Buoni and Varga, are considered. Under some basic assumptions the various parameters of the optimal Generalized JOR and SOR schemes are determined through formulas given by means of specific algorithms which are proposed. A number of numerical examples are also presented to show how one can apply the algorithms and determine, subsequently, the optimal parameters which make the corresponding schemes to semiconverge as fast as possible.

AMS(MOS) subject classification. 65F10

1. Introduction.

Assume that

$$Ax = b \tag{1.1}$$

is a complex, in general, linear system to be solved iteratively with $A \in \mathbb{C}^{n,n}$, $\det(A) = 0$, $x, b \in \mathbb{C}^n$ and b in the range of A. We write A as follows

$$A = D - L - U, \tag{1.2}$$

where $\det(D) \neq 0$ and D, L and U are not necessarily diagonal, strictly lower and strictly upper triangular matrices respectively. As in Buoni and Varga ([3] and [4]) we form the Generalized Jacobi (GJ) the Extrapolated Generalized Jacobi (EGJ or GJOR) and the Generalized Successive Overrelaxation (GSOR) iterative schemes associated with splitting (1.2) for the solution of (1.1). These are the following

$$x^{(m+1)} = Tx^{(m)} + c, \qquad m = 0, 1, 2, \cdots, \tag{1.3}$$

$$x^{(m+1)} = T_\omega x^{(m)} + \omega c, \qquad m = 0, 1, 2, \cdots \tag{1.4}$$

and

$$x^{(m+1)} = \mathcal{L}_\omega x^{(m)} + \omega(D - \omega L)^{-1} b, \qquad m = 0, 1, 2, \cdots \tag{1.5}$$

respectively, where

$$\begin{aligned} &T \equiv D^{-1}(L + U), \qquad c = D^{-1}b, \\ &T_\omega \equiv (1 - \omega)I + \omega T, \\ &\mathcal{L}_\omega \equiv (D - \omega L)^{-1}[(1 - \omega)D + \omega U]. \end{aligned} \tag{1.6}$$

In (1.4) $\omega \in \mathbb{C} - \{0\}$ is the extrapolation parameter and in (1.5) $\omega \in \mathbb{C} - \{0\}$, with $\det(D - \omega L) \neq 0$, is the overrelaxation parameter. (*Note.* In view of splitting (1.2), T, in (1.6), may well have nonzero diagonal entries, which is not the case with the classical Jacobi matrix, where D is chosen either as the diagonal or as a block diagonal part of A. Thus T, and its components $D^{-1}L$ and $D^{-1}U$, will be considered, in the sequel, in this generalized form unless otherwise stated.)

Assume further that the spectrum $\sigma(T)$ of T, that is the discrete set of the eigenvalues λ_j, $j = 1(1)n$, of T, is known. Since A, in (1.1), is singular it is implied that

* Received by the editors October 18, 1983, and in revised form April 5, 1984.
† Department of Mathematics, University of Ioannina, Ioannina, Greece.

the real number one (1) is an eigenvalue of T with the same multiplicity with which the number zero (0) is an eigenvalue of A. We denote by H the hull of $\sigma(T)$, that is the smallest convex polygon containing $\sigma(T)$ in the closure of its interior. We also denote by $\gamma(T)$ the number defined by

$$\gamma(T) \neq \max_j \{|\lambda_j|: \lambda_j \in \sigma(T), \lambda_j \neq 1\}, \tag{1.7}$$

which is connected with the semiconvergence of scheme (1.3) (Note: Since at least one of the eigenvalues of T is known to be the number 1, $\rho(T) \geqq 1$ and (1.3) does not converge). Here it is simply reminded that $\rho(T)=1$, $\gamma(T)<1$ together with the fulfillment of the requirement index $(I-T)=1$ (that is, all the elementary divisors of T associated with the eigenvalue 1 are linear) imply that (1.3) semiconverges (see Berman and Plemmons [1, p. 152] as well as [2]). In such a case $\gamma(T)$ is the asymptotic semiconvergence factor and $-\ln \gamma(T)$ gives the asymptotic rate of semiconvergence of (1.3) (see e.g. [1, p. 198]).

In what follows we shall accept that the following two assumptions are valid.

Assumption I. The point $A(1, 0)$ of the complex plane is one of the vertices of H and

Assumption II. index $(I-T)=1$.

The purpose of this paper is twofold: i) To prove, under Assumptions I and II, the existence and uniqueness of an optimum $\omega(\omega_{\text{opt}}) \in \mathbb{C}-\{0\}$ such that the GJOR scheme (1.4) semiconverges as fast as possible and also to give expressions (by means of a known algorithm) for both ω_{opt} and $\gamma(T_{\omega_{\text{opt}}})$; ii) Under some further assumptions concerning the components $D^{-1}L$ and $D^{-1}U$ of T as well as the matrix T itself to prove again the existence and uniqueness of an optimum $\omega(\omega_{\text{opt}}) \in \mathbb{C}-\{0\}$, with $\det(D-\omega L) \neq 0$, such that the GSOR scheme (1.6) semiconverges as fast as possible and also to give, as before, a means of determining ω_{opt} and $\gamma(\mathcal{L}_{\omega_{\text{opt}}})$.

2. Optimum GJOR iterative scheme. Consider the GJOR scheme (1.4) together with the appropriate relationships from (1.6). Since

$$I - T_\omega = \omega(I-T) \tag{2.1}$$

and $\omega \neq 0$, it is concluded that in view of Assumption II for T

$$\text{index}\,(I-T_\omega) = \text{index}\,(I-T) = 1. \tag{2.2}$$

Because of the expression of T_ω as a function of T (see (1.6)), the eigenvalues μ_j of T_ω will be given in terms of those of T through similar relationships. Namely

$$\mu_j = 1-\omega+\omega\lambda_j, \qquad j=1(1)n. \tag{2.3}$$

It is obvious that the point $A(1, 0)$ of the complex plane is invariant under the transformation corresponding to (2.3). A consequence of this is that $\rho(T_\omega) \geqq 1$. So our objective is to make $\rho(T_\omega)=1$ on the one hand and on the other hand the GJOR scheme (1.4) to semiconvergence as fast as possible. For this we state and prove the following theorem which will be very useful in our analysis.

THEOREM 2.1. *Let the matrix T in* (1.6) *satisfy Assumptions* I *and* II. *Then the problem of determining an $\omega(\omega_{\text{opt}})$ such that the* GJOR *scheme* (1.4) *semiconverges in an optimum sense is equivalent to the problem of minimizing the spectral radius $\rho(\tilde{T}_\omega)$ of an extrapolation matrix $\tilde{T}_\omega$ of $\tilde{T}$ with spectrum of the latter $\sigma(\tilde{T}) \equiv \sigma(T)-\{1\}$.*

Proof. Since T satisfies Assumption II there will exist a nonsingular matrix Q such that

$$T = Q\begin{bmatrix} I_s & 0 \\ 0 & \tilde{T} \end{bmatrix} Q^{-1}, \tag{2.4}$$

where I_s is the $s \times s$ unit matrix $(1 \leqq s \leqq n-1)$ and $\tilde{T}$ is a matrix of order $n-s$ with eigenvalue spectrum $\sigma(\tilde{T}) \equiv \sigma(T) - \{1\}$. In view of (2.4) and Assumption I for T, the hull $\tilde{H}$ of $\tilde{T}$ will leave the point $A(1, 0)$ strictly in its exterior. From (1.6) and (2.4) it is concluded that T_ω will have the form

$$T_\omega = Q\begin{bmatrix} I_s & 0 \\ 0 & \tilde{T}_\omega \end{bmatrix} Q^{-1}, \tag{2.5}$$

where

$$\tilde{T}_\omega = (1-\omega) I_{n-s} + \omega \tilde{T} \tag{2.6}$$

so that $\tilde{T}_\omega$ is an extrapolation matrix of $\tilde{T}$ with extrapolation parameter ω. We notice that $\tilde{T}_\omega$ can not have the number 1 as an eigenvalue, for otherwise $\tilde{T}$ will have the same eigenvalue as well which is not possible. From (2.6) it is seen that the eigenvalues μ_j of $\tilde{T}_\omega$ will be given again by (2.3) with, however, $\lambda_j \in \sigma(\tilde{T})$. Because of the form (2.5) it is $\rho(T_\omega) = \max\{1, \rho(\tilde{T}_\omega)\}$ and, by virtue of (1.7), $\gamma(T_\omega) = \rho(\tilde{T}_\omega)$. Since, from (2.2), T_ω satisfies Assumption II scheme (1.4) will be a semiconvergent one iff $\rho(\tilde{T}_\omega) < 1$. On the other hand (1.4) will semiconverge in an optimum sense for that $\omega(\omega_{\text{opt}})$ for which $\rho(\tilde{T}_\omega)(<1)$ is a minimum and the theorem is proved. □

Having proved among others in Theorem 2.1 that $A(1, 0) \notin \tilde{H}$ we state the main result of [7] which guarantees the existence and the uniqueness of the optimum extrapolation parameter of the previous theorem.

THEOREM 2.2. *If the hull $\tilde{H}$ of $\tilde{T}$ of Theorem* 2.1 *is known and $A(1, 0) \notin \tilde{H}$ then there always exists a unique $\omega(\omega_{\text{opt}})$ for which the spectral radius $\rho(\tilde{T}_\omega)$ of the extrapolation matrix $\tilde{T}_\omega$ of $\tilde{T}$ becomes a minimum (less than one).*

The determination of ω_{opt} of Theorem 2.2 above is achieved by means of an algorithm (see [7]) which is based on the concept of the optimum capturing circle and is outlined very briefly below.

ALGORITHM FOR THE DETERMINATION OF ω_{opt}. Assume that λ_j, $j = 1(1)l \in [1, n-s]$ is the reordered set of the eigenvalues of T different from 1. Let λ_j, $j = 1(1)k \in [1, l]$, be the new reordered set of those eigenvalues of $\sigma(\tilde{T})$ which are vertices of $\tilde{H}$, and let P_j, $j = 1(1)k$, be their images (points) in the complex plane. Omitting the trivial case $k = 1$, when

$$\omega_{\text{opt}} = 1/(1-\lambda_1), \qquad \gamma(T_{\omega_{\text{opt}}}) = \rho(\tilde{T}_{\omega_{\text{opt}}}) = 0,$$

the algorithm to determine ω_{opt} and $\gamma(T_{\omega_{\text{opt}}})$ runs as follows: 1) Take the points P_j, $j = 1(1)k$ two at a time (P_{j_1}, P_{j_2}) and consider the intersection $K_{j_1 j_2}(c_1, c_2)$ of the perpendicular to $P_{j_1}P_{j_2}$ at its midpoint with the circle circumscribed to the triangle $AP_{j_1}P_{j_2}$. If the circle with center point $K_{j_1 j_2}$ and radius $R = (K_{j_1 j_2} P_{j_1})$ captures $\tilde{H}$, the circle at hand is the optimum capturing circle, so go to step 3. If no such circle exists go to the next step. 2) Take the points P_j, $j = 1(1)k$ three at a time $(P_{j_1}, P_{j_2}, P_{j_3})$ and consider the circle circumscribed to the triangle $P_{j_1}P_{j_2}P_{j_3}$. Let $K_{j_1 j_2 j_3}(c_1, c_2)$ be its center and $R = (K_{j_1 j_2 j_3} P_{j_1})$ be its radius. If this circle captures $\tilde{H}$ and leaves $A(1, 0)$ in its exterior then it is a possible candidate for the optimum capturing one. Find all possible candidates and proceed to the next step. 3) In case we have come from step 1 or from

step 2 with only one possible candidate the optimum values ω_{opt} and $\gamma(T_{\omega_{\text{opt}}})$ are determined through the relationships

$$\omega_{\text{opt}}=\frac{(1-c_1)+ic_2}{(1-c_1)^2+c_2^2},\qquad \gamma(T_{\omega_{\text{opt}}})=\frac{R}{((1-c_1)^2+c_2^2)^{1/2}}. \tag{2.7}$$

In case we have come from step 2 with more than one possible candidate, the optimum capturing circle is the one which corresponds to the smallest $\gamma(T_{\omega_{\text{opt}}})$ of (2.7). As is proved in [7] the circle in question always exists and is unique. □

Remarks. i) In case $\tilde{H}$ is, in addition, symmetric with respect to (wrt) the real axis (then H will also be symmetric wrt the real axis, as for example in the case where T is real), ω_{opt} is a real number and the algorithm given previously may be simplified (see [7], [5] or [6]).

ii) In case $\tilde{H}$ is not symmetric wrt the real axis but lies strictly to the left or strictly to the right of the line $z=1$ of the complex plane and we restrict ourselves to finding a real ω such that scheme (1.4) semiconverges as fast as possible, a new auxiliary hull $\tilde{H}'$ of $\sigma(\tilde{T})$ symmetric wrt the real axis is considered. Then what is mentioned in i) above is now applied with $\tilde{H}'$ taking the place of $\tilde{H}$.

3. Optimum GSOR iterative scheme. In this section together with the assumptions I and II we shall consider two more basic assumptions. More specifically we consider:

Assumption III. The components $D^{-1}L$ and $D^{-1}U$ of T, in (1.6), are strictly lower and strictly upper triangular matrices respectively.

Assumption IV. T is a weakly p-cyclic consistently ordered matrix.

Assumption III insures on the one hand that $\det(D-\omega L)=\det(D)\det(I-\omega D^{-1}L)=\det(D)\neq 0$ for any $\omega\in\mathbb{C}$ and on the other hand that an obvious extension of Kahan's theorem (see [9, p. 75]) holds. In other words a necessary condition for semiconvergence of the GSOR iterative scheme (1.5) is $|\omega-1|\leqq 1$, with $\omega\in\mathbb{C}-\{0\}$. Assumption IV, together with Assumption III, provides us with a functional relationship connecting the eigenvalues $\lambda_j, j=1(1)n$ of T and the eigenvalues $\mu_j, j=1(1)n$ of $\mathcal{L}_\omega$. This is the well-known one

$$(\mu_j+\omega-1)^p=\mu_j^{p-1}\omega^p\lambda_j^p \tag{3.1}$$

(see [9] or [10]).

Before we go on with the determination of the optimum semiconvergent GSOR scheme we make two observations: i) As is known and because of Assumptions III and IV and by virtue of Romanovsky's theorem (see [9, p. 40, Thm. 2.4]) apart from the zero eigenvalues of T, all others appear in p-tuples with the same multiplicity for the p elements of each p-tuple. ii) Because of Assumption I or II and observation i) the numbers $e^{2\pi iq/p}$, $q=0(1)p-1$ are eigenvalues of T of the same multiplicity.

Now we can prove the following theorem.

THEOREM 3.1. *If T in* (1.6) *satisfies Assumptions* I-IV *and* $\omega\in\mathbb{C}-\{0, p/(p-1)\}$ *with* $|\omega-1|\leqq 1$, $\mathcal{L}_\omega$ *given in* (1.6) *will satisfy Assumption* II.

Proof. First we observe that no eigenvalue λ_j of T with modulus different from 1 can have as an image, through (3.1), the eigenvalue $\mu_j=1$ of $\mathcal{L}_\omega$. Indeed, if that were the case then on substitution in (3.1) we would obtain $(1+\omega-1)^p=1\omega^p\lambda_j^p$. Since $\omega\neq 0$, $|\lambda_j|=1$ which contradicts our assumption that $|\lambda_j|\neq 1$. By virtue of observation ii) made previously for each set of p eigenvalues $\lambda_j=e^{2\pi iq/p}$, $q=0(1)p-1$, of T, (3.1) will give

$$f(\mu_j)\equiv(\mu_j+\omega-1)^p-\omega^p\mu_j^{p-1}=0. \tag{3.2}$$

Since it is readily verified that $f(1)=0$, while $f'(1)=p(1+\omega-1)^{p-1}-(p-1)\omega^p=\omega^{p-1}(p-(p-1)\omega)\neq 0$, $f(\mu_j)$ will have the number 1 as a simple root. This result together with observation ii) imply that the eigenvalue 1 of $\mathcal{L}_\omega$ has the same multiplicity which the number 1 has as an eigenvalue of T. Since it can be obtained that

$$I-\mathcal{L}_\omega=\omega(I-\omega D^{-1}L)^{-1}(I-T)$$

and the number 1 is an eigenvalue of T and $\mathcal{L}_\omega$ with the same multiplicity, then the relationship above and Assumption II for T imply that

$$\text{index}\,(I-\mathcal{L}_\omega)=\text{index}\,(I-T)=1, \tag{3.3}$$

which proves the present theorem. □

To be able to prove one of our main results of this section we present Varga's theorem [9, Thm. 4.4, pp. 111-112].

THEOREM 3.2. *Let T in* (1.6), *which corresponds to a nonsingular matrix A in* (1.1)-(1.2), *satisfy Assumptions* III *and* IV. *If all the eigenvalues of the pth power of T are real and nonnegative, and $0\leqq\rho(T)<1$, and with ω_b the unique positive real root (less than $p/(p-1)$) of the equation*

$$(\rho(T)\omega_b)^p=[p^p(p-1)^{1-p}](\omega_b-1), \tag{3.4a}$$

then

$$\min_{\omega\in\mathbb{R}}\rho(\mathcal{L}_\omega)=\rho(\mathcal{L}_{\omega_b})=(\omega_b-1)(p-1). \tag{3.4b}$$

Now we are in a position to state and prove an analogue to the previous theorem for the determination of the optimum $\omega(\tilde{\omega})$ in the singular case.

THEOREM 3.3. *Let T in* (1.6) *satisfy Assumptions* I-IV *and let the eigenvalues of its pth power be real and nonnegative with $\rho(T)=1$. Let also $\tilde{\sigma}(T)\equiv\sigma(T)$, $\{\lambda_j\colon \lambda_j\in\sigma(T)$ and $|\lambda_j|=1\}$ and $\tilde{\rho}(T)\equiv\max\{|\lambda_j|, \lambda_j\in\tilde{\sigma}(T)\}$. Then the optimum $\omega(\tilde{\omega})$ for which the* GSOR *scheme* (1.5) *semiconverges as fast as possible is the $\tilde{\omega}=\omega_b$ given by* (3.4a) *with $\rho(T)$ being substituted by $\tilde{\rho}(T)$, namely*

$$(\tilde{\rho}(T)\tilde{\omega})^p=[p^p(p-1)^{1-p}](\tilde{\omega}-1) \tag{3.5a}$$

and

$$0<\tilde{\omega}<p/(p-1). \tag{3.5b}$$

Then

$$\min_{\omega\in\mathbb{R}}\gamma(\mathcal{L}_\omega)=\gamma(\mathcal{L}_{\tilde{\omega}})=(\tilde{\omega}-1)(p-1). \tag{3.5c}$$

Proof. From the relationships (3.4b) of Theorem 3.2 and the assumptions of our present theorem it is obvious that the images μ_j, through (3.1), of all eigenvalues $\lambda_j\in\tilde{\sigma}(T)$ of T will satisfy (3.5a)-(3.5b) and the corresponding to (3.4b) relationships which will be

$$\min_{\omega\in\mathbb{R}}\tilde{\rho}(\mathcal{L}_\omega)=\tilde{\rho}(\mathcal{L}_{\tilde{\omega}})=(\tilde{\omega}-1)(p-1), \tag{3.6a}$$

where

$$|\mu_j|\leqq\tilde{\rho}(\mathcal{L}_{\tilde{\omega}})=(\tilde{\omega}-1)(p-1). \tag{3.6b}$$

Therefore it is concluded that

$$\gamma(\mathcal{L}_{\tilde{\omega}})\geqq(\tilde{\omega}-1)(p-1). \tag{3.7}$$

So it remains to prove that the images μ_j of the eigenvalues $\lambda_j = e^{2\pi iq/p}$, $q = 0(1)p-1$, of T (of modulus 1 except the eigenvalue 1 itself) satisfy (3.6b). (In fact the relationship which they satisfy is a strict one). Since for the eigenvalues in question $\lambda_j^p = 1$, equation (3.2) will be valid. From Theorem 3.1 we know that $f(\mu_j)$ possesses 1 as a simple root. We divide out $f(\mu_j)$ by $\mu_j - 1$, consider $\tilde{\omega}$ in the place of ω, drop indices for simplicity and put $y = \tilde{\omega} - 1$ to obtain

$$
\begin{aligned}
g(\mu) \equiv \frac{f(\mu)}{\mu - 1} = \mu^{p-1} &+ \left[1 + \binom{p}{1} y - \tilde{\omega}^p\right] \mu^{p-2} \\
&+ \left[1 + \binom{p}{1} y + \binom{p}{2} y^2 - \tilde{\omega}^p\right] \mu^{p-3} + \cdots \\
&+ \left[1 + \binom{p}{1} y + \binom{p}{2} y^2 + \cdots + \binom{p}{p-1} y^{p-1} - \tilde{\omega}^p\right].
\end{aligned} \tag{3.8}
$$

To prove that the roots of $g(\mu)$ are strictly less than $(\tilde{\omega} - 1)(p-1) = yz$ in modulus (where we put $z = p - 1$) it is sufficient and necessary to prove that the roots of $h(\nu) \equiv h(\mu/(yz)) \equiv g(\mu)/(yz)^{1-p}$ have moduli strictly less than 1. From (3.8) we have

$$
\begin{aligned}
h(\nu) \equiv \nu^{p-1} &+ \frac{1}{yz}\left[1 + \binom{p}{1} y - (1+y)^p\right] \nu^{p-2} \\
&+ \frac{1}{y^2 z^2}\left[1 + \binom{p}{1} y + \binom{p}{2} y^2 - (1+y)^p\right] \nu^{p-3} + \cdots \\
&+ \frac{1}{y^{p-1} z^{p-1}}\left[1 + \binom{p}{1} y + \binom{p}{2} y^2 + \cdots + \binom{p}{p-1} y^{p-1} - (1+y)^p\right]
\end{aligned}
$$

or

$$
\begin{aligned}
h(\nu) \equiv \nu^{p-1} &- \frac{1}{yz}\left[\binom{p}{2} y^2 + \binom{p}{3} y^3 + \cdots + \binom{p}{p} y^p\right] \nu^{p-2} \\
&- \frac{1}{y^2 z^2}\left[\binom{p}{3} y^3 + \cdots + \binom{p}{p} y^p\right] \nu^{p-3} - \cdots - \frac{1}{y^{p-1} z^{p-1}} \binom{p}{p} y^p.
\end{aligned} \tag{3.9}
$$

From (3.9), by putting $h(\nu) = 0$ and since $\nu \neq 0$, we can obtain

$$
\begin{aligned}
\nu = &\frac{1}{yz}\left[\binom{p}{2} y^2 + \binom{p}{3} y^3 + \cdots + \binom{p}{p} y^p\right] \frac{1}{\nu} \\
&+ \frac{1}{y^2 z^2}\left[\binom{p}{3} y^3 + \cdots + \binom{p}{p} y^p\right] \frac{1}{\nu^2} + \cdots + \frac{1}{y^{p-1} z^{p-1}} \binom{p}{p} y^p \frac{1}{\nu^{p-2}}.
\end{aligned} \tag{3.10}
$$

Suppose now that at least one root of $h(\nu)$ or equivalently of (3.10) is in modulus greater than or equal to 1. By taking absolute values in (3.10) we successively obtain

$$
\begin{aligned}
1 \leqq |\nu| &\leqq \frac{1}{yz}\left[\binom{p}{2} y^2 + \binom{p}{3} y^3 + \cdots + \binom{p}{p} y^p\right] \\
&\quad + \frac{1}{y^2 z^2}\left[\binom{p}{3} y^3 + \cdots + \binom{p}{p} y^p\right] + \cdots + \frac{1}{y^{p-1} z^{p-1}} \binom{p}{p} y^p \\
&= \frac{y}{z}\binom{p}{2} + \left(\frac{y^2}{z} + \frac{y}{z^2}\right)\binom{p}{3} + \left(\frac{y^3}{z} + \frac{y^2}{z^2} + \frac{y}{z^3}\right)\binom{p}{4}
\end{aligned}
$$

$$+\cdots+\left(\frac{y^{p-1}}{z}+\frac{y^{p-2}}{z^2}+\cdots+\frac{y}{z^{p-1}}\right)\binom{p}{p}$$

$$=\frac{yz}{1-yz}\left[\left(\frac{1}{z^2}-\frac{y}{z}\right)\binom{p}{2}+\left(\frac{1}{z^3}-\frac{y^2}{z}\right)\binom{p}{3}\right.$$

$$\left.+\left(\frac{1}{z^4}-\frac{y^3}{z}\right)\binom{p}{4}+\cdots+\left(\frac{1}{z^p}-\frac{y^{p-1}}{z}\right)\binom{p}{p}\right]$$

$$=\frac{yz}{1-yz}\left\{\left[\frac{1}{z^2}\binom{p}{2}+\frac{1}{z^3}\binom{p}{3}+\cdots+\left(\frac{1}{z^p}\right)\binom{p}{p}\right]\right.$$

$$\left.-\frac{1}{yz}\left[y^2\binom{p}{2}+y^3\binom{p}{3}+\cdots+y^p\binom{p}{p}\right]\right\}$$

$$=\frac{yz}{1-yz}\left\{\left(\frac{1}{z}+1\right)^p-1-\frac{p}{z}-\frac{1}{yz}[(1+y)^p-1-py]\right\}.$$

From the first and the last members of the above series of relationships we take

$$1\leqq\frac{1}{1-yz}\left[\frac{y}{z^{p-1}}(1+z)^p-yz-py-(1+y)^p+1+py\right],$$

or equivalently, after a simple manipulation and returning again back to the original variables $\tilde{\omega}=1+y$ and $p=1+z$ we have

$$\tilde{\omega}^p\leqq[p^p(p-1)^{1-p}](\tilde{\omega}-1). \tag{3.11}$$

However, from (3.5a) and since $\tilde{\rho}(T)<1$ it is

$$\tilde{\omega}^p>[p^p(p-1)^{1-p}](\tilde{\omega}-1),$$

which contradicts (3.11). Therefore the images μ_j of $\lambda_j=e^{2\pi iq/p}$, $q=0(1)p-1$ of T (except the image $\mu_j=1$) are in a modulus strictly less than $(\tilde{\omega}-1)(p-1)$. Consequently (3.7) turns out to be an equality and (3.5c) has been proved. □

A much stronger result is obtained in case $p=2$. More specifically we have the following theorem.

THEOREM 3.4. *Let T in* (1.6) *satisfy Assumptions* I-IV *with $p=2$. If there exists an eigenvalue-pair $\lambda_{j_1}=-\lambda_{j_2}(\neq 1)$ of T with corresponding images, through* (3.1) *(for $p=2$), such that* $\max(|\mu_{j_1}|,|\mu_{j_2}|)$ *is larger than the moduli of all other images μ_j of $\mathcal{L}_\omega$ (except 1 and $(\omega-1)^2$ which come from the pair ±1 of T) for all ω's for which $|\omega-1|<1$, then the optimum parameters for the* GSOR *scheme* (1.5) *are given by*

$$\omega_{\text{opt}}=\tilde{\omega}=\frac{2}{1+(1-\lambda_j^2)^{1/2}},\qquad \gamma(\mathcal{L}_{\omega_{\text{opt}}})=\left|\frac{1-(1-\lambda_{j_1}^2)^{1/2}}{1+(1-\lambda_{j_1}^2)^{1/2}}\right|. \tag{3.12}$$

Proof. Let now that $\tilde{\sigma}(T)=\sigma(T)-\{-1,1\}$ and $\tilde{H}$ is the hull of $\tilde{\sigma}(T)$. In view of the previous observations (i) and (ii) and Assumption I it is concluded that H will be symmetric wrt the origin $0(0,0)$ and that the point $A'(-1,0)$ will be also a vertex of H. Similarly $\tilde{H}$ will be symmetric wrt the origin. For the time being we shall restrict ourselves to considering $\tilde{\sigma}(T)$ and its hull $\tilde{H}$ ignoring the eigenvalues ±1 of T. As is known (see Kredell [8]) any ellipse $\mathscr{E}$ centered at the origin and capturing $\tilde{H}$ in the closure of its interior will be mapped, through (3.1), onto a circle $\mathscr{C}_\omega$, centered at the origin, which, in turn, will capture all the eigenvalues of $\mathcal{L}_\omega$, except 1 and $(\omega-1)^2$, which are the images of the pair ±1, in the closure of its interior for any $\omega\neq0$. Since

according to our hypotheses there exists a critical eigenvalue-pair $\lambda_{j_1} = -\lambda_{j_2}$ in the sense of [8] with corresponding images μ_{j_1}, μ_{j_2}, through (3.1), are such that $\max(|\mu_{j_1}|, |\mu_{j_2}|)$ is larger than the moduli of all other μ_j's of $\mathcal{L}_\omega$, except 1 and $(\omega-1)^2$, for all possible ω's for which $|\omega-1|<1$. We then have that there exists a complex, in general, $\tilde{\omega}$ determined by

$$\tilde{\omega} = \frac{2}{1+(1-\lambda_{j_1}^2)^{1/2}},$$

where the square root of the complex number with the positive real part is taken, which minimizes the quantity $|\omega-1|(<1)$, namely the radius of $\mathcal{C}_\omega$. Since such is the situation we turn our attention to the eigenvalues ± 1 of T which we have ignored. These have images 1 and $(\tilde{\omega}-1)^2$. The following are deduced. The number 1 will be an eigenvalue of $\mathcal{L}_{\tilde{\omega}}$ such that (3.3) will be satisfied and the number $(\tilde{\omega}-1)^2$ will be an eigenvalue of $\mathcal{L}_{\tilde{\omega}}$ satisfying $|(\tilde{\omega}-1)^2|<|\tilde{\omega}-1|$. Thus $(\tilde{\omega}-1)^2$ will lie strictly in the interior of the circle $\mathcal{C}_{\tilde{\omega}}$ whose radius equals $|\tilde{\omega}-1|$. Therefore by virtue of the analysis so far it is implied that the optimum parameters are given by (3.12). □

In Theorem 3.4 the optimum $\tilde{\omega}(\omega)$ is in general complex. However, one may find in some cases an optimum real ω. In this direction the following analysis is of great value, under the hypotheses that Assumptions I-IV for T are valid and $p=2$.

In case $|\mathrm{Re}\,\lambda_j| \leqq 1$ holds for all the eigenvalues of T (with equality holding for those eigenvalues equal to ± 1 only) and no critical eigenvalue-pair exists or if it exists it is very difficult or even impossible to determine, then the following strategy to obtain an optimum real ω and the corresponding optimum semiconvergent scheme is adopted. Let $\tilde{H}'$ be the hull of $\tilde{\sigma}(T)$ which is now symmetric wrt both axes. By ignoring again, for the time being, the eigenvalues ± 1 of T and by following the algorithm by Young and Eidson [11] (see also [10, pp. 194-200]) an optimum capturing ellipse $\mathcal{E}$ which contains all the eigenvalues of T except those equal to ± 1 in the closure of its interior is uniquely determined. As is obvious this ellipse will be an optimum capturing one for $\tilde{H}$ as well as $\tilde{H}'$. Let $\hat{M}_r < 1$ and $\hat{M}_i$ be its real and imaginary semi-axes respectively. The optimum value for $\omega(\hat{\omega})$ as well as the value for the radius $\hat{\rho}^2$ of the optimum capturing circle $\mathcal{C}_{\hat{\omega}}$, which is the image of $\mathcal{E}$, through (3.1), will be given by the expressions

$$\hat{\omega} = \frac{2}{1+(1-\hat{M}_r^2+\hat{M}_i^2)^{1/2}}, \qquad \hat{\rho}^2 = \left(\frac{M_r+M_i}{1+(1-\hat{M}_r^2+\hat{M}_i^2)^{1/2}}\right)^2 \tag{3.13}$$

(see [10, p. 194, equations (4.14)-(4.15)]). The optimum above is in the sense that the radius $\hat{\rho}^2$ of the capturing circle $\mathcal{C}_{\hat{\omega}}$ is a minimum. Coming now to the eigenvalues 1 and $(\hat{\omega}-1)^2$ of $\mathcal{L}_{\hat{\omega}}$ which are the images of the eigenvalues ± 1 of T we readily notice, as before in Theorem 3.4, the following. First the number 1 is an eigenvalue of $\mathcal{L}_{\hat{\omega}}$ with (3.3) holding and then the number $(\hat{\omega}-1)^2$ is also an eigenvalue of $\mathcal{L}_{\hat{\omega}}$ which, because of the relationship $|(\hat{\omega}-1)^2| \leqq \hat{\rho}^2$, lies strictly in the interior of the optimum capturing circle $\mathcal{C}_{\hat{\omega}}$. Therefore

$$\omega_{\mathrm{opt}} = \hat{\omega}, \qquad \gamma(\mathcal{L}_{\omega_{\mathrm{opt}}}) = \hat{\rho}^2, \tag{3.14}$$

with $\hat{\omega}$ and $\hat{\rho}$ being given by (3.13). It is remarked that if, besides the eigenvalues ± 1, T possesses no other eigenvalues (or the eigenvalue zero only) then $\hat{M}_r = \hat{M}_i = 0$ and from (3.6) $\hat{\omega}=1$, $(\hat{\omega}-1)^2=0$ and $\hat{\rho}^2=0$. Hence from (3.14), $\omega_{\mathrm{opt}}=1$ and $\gamma(\mathcal{L}_{\omega_{\mathrm{opt}}})=0$. In other words the optimum semiconvergent scheme in this case is the Generalized Gauss-Seidel (GGS) one with $\gamma(\mathcal{L}_1)=0$!

Remark. If the matrix $I-T$ is an M-matrix then because of Assumption I or II, $I-T$ is singular. By virtue of the definition of a singular M-matrix (see [1]) it is implied that T is a (real) nonnegative matrix and $\rho(T)\leqq 1$, with the only eigenvalues satisfying $|\mathrm{Re}\,\lambda_j|=1$ those equal to ± 1. As is noticed Assumption I follows directly from Assumption II and the fact that $\rho(T)\leqq 1$. Therefore it can be dropped from the basic assumptions. Since T is real and Assumption IV is valid, $\sigma(T)$ and $\tilde{\sigma}(T)$ of Theorem 3.4 will be symmetric wrt both axes; so will be their hulls H and $\tilde{H}$. Consequently the analysis of the case examined previously can be applied straightforwardly to give a real value ω_{opt} and then $\gamma(\mathcal{L}_{\omega_{\mathrm{opt}}})$ by means of the formulas (3.13)-(3.14). (*Notes*: i) If instead of assuming that $I-T$ is simply an M-matrix, one assumes that $I-T$ is an irreducible singular M-matrix then Assumption II follows, so that it is not needed to be included in the basic assumptions (see [1, Thm. 4.16, p. 156]) and the previous theory to determine the optimum values for ω and $\gamma(\mathcal{L}_\omega)$ can be applied again as before. ii) The present remark and the previous note i) can be also applied in a straightforward way for the case of the optimum GJOR method. This is an immediate consequence of the remark i) of the previous section.) □

4. Numerical examples.

A. *Optimum* GJOR *method.*

i) Let $T\in\mathbb{C}^{4,4}$ with $\sigma(T)\equiv\{-1+3i, 3+3i, 3i, 1\}$. As is seen the eigenvalue 1 is simple so that $\mathrm{index}(I-T)=1$ and all others lie on the straight line $z=3i$ of the complex plane. Thus $A(1,0)$ is a vertex of the hull H of $\sigma(T)$, which is the triangle with vertices $A\equiv P_3(1,0)$, $P_1(-1,3)$ and $P_2(3,3)$. We also have $\tilde{\sigma}(T)\equiv\{-1+3i, 3+3i, 3i\}$. Its hull $\tilde{H}$ is the line segment with end points $P_1(-1,3)$ and $P_2(3,3)$ (see Fig. 1). By following the algorithm of § 2 we find the point of intersection $K_{12}(c_1, c_2)$

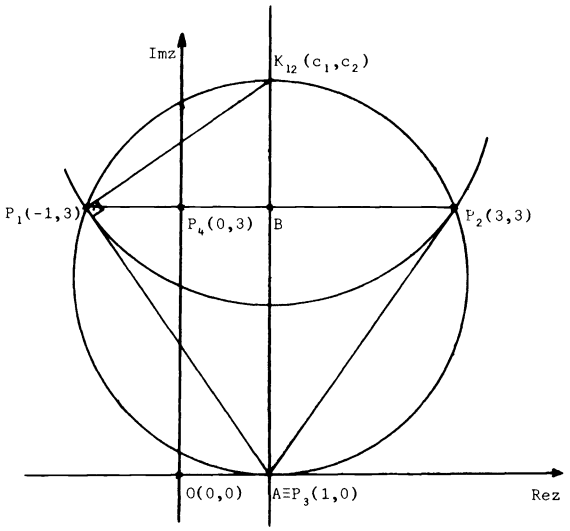


FIG. 1

of the perpendicular to P_1P_2 at its midpoint (it is the line $z=1$) with the circle circumscribed to the triangle AP_1P_2. From Fig. 1, $(AP_1)^2=(AK_{12})(AB)$ or $(-1-1)^2+(3-0)^2=(AK_{12})((1-1)^2+(3-0)^2)^{1/2}$, which gives $(AK_{12})=13/3$. Therefore $K_{12}(c_1, c_2)\equiv K_{12}(1, 13/3)$, and $R=(K_{12}P_1)=((-1-1)^2+(3-13/3)^2)^{1/2}=2\sqrt{13}/3$.

Using (2.7) we obtain

$$\omega_{\text{opt}}=\frac{(1-1)+i13/3}{(1-1)^2+(13/3)^2}=\frac{3i}{13}$$

and

$$\gamma(T_{\omega_{\text{opt}}})=\frac{2\sqrt{13}/3}{((1-1)^2+(13/3)^2)^{1/2}}=\frac{2\sqrt{13}}{13}\approx 0.5547.$$

If we restrict ourselves to real values of ω in order to find an optimum semiconvergent scheme we simply note that no semiconvergent scheme can be obtained by extrapolation. This is because the hull $\tilde{H}$ does not lie strictly to the left or strictly to the right of the line $z=1$.

ii) Let $T\in\mathbb{C}^{5,5}$ and $\sigma(T)\equiv\{-1,-1+i,-1+3i/4,1\}$, where 1 is a double eigenvalue of T with index$(I-T)=1$. In this case the hull H of $\sigma(T)$ is the triangle with vertices $A\equiv P_3(1,0)$, $P_1(-1,0)$ and $P_2(-1,1)$. It is $\tilde{\sigma}(T)\equiv\{-1,-1+i,-1+3i/4\}$ with hull $\tilde{H}$ the line segment with end points $P_1(-1,0)$ and $P_2(-1,1)$ (see Fig. 2). By following the same algorithm as in the previous example we find that $K_{12}(c_1,c_2)$ is the intersection of the perpendicular to the midpoint of the line segment P_1P_2 (with equation $z=i/2$) with the circle circumscribed to the triangle AP_1P_2. (Its diameter is the line segment AP_2.) From Fig. 2 we have $(BK_{12})^2=(BP_1)^2=(-1-0)^2+(0-1/2)^2=5/4$,

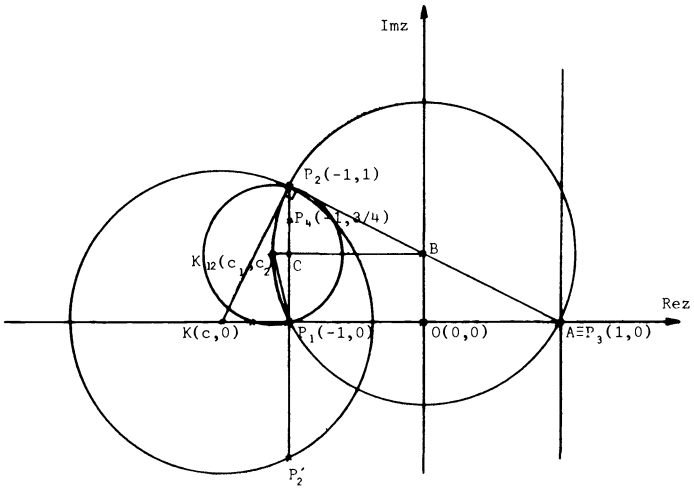


FIG. 2

so that $(BK_{12})=\sqrt{5}/2$. Thus $c_1=-\sqrt{5}/2$, $c_2=1/2$. Also $R=(K_{12}P_1)=((P_1C)^2+(CK_{12})^2)^{1/2}=((-1-(-1))^2+(1/2-0)^2+(-\sqrt{5}/2-(-1))^2+(1/2-1/2)^2)^{1/2}=(10-4\sqrt{5})^{1/2}/2$. Therefore

$$\omega_{\text{opt}}=\frac{(1-(-\sqrt{5}/2))+i1/2}{(1-(-\sqrt{5}/2))^2+(1/2)^2}=\frac{\sqrt{5}}{5}+i\frac{5-2\sqrt{5}}{5}$$

and

$$\gamma(T_{\omega_{\text{opt}}})=\frac{(10-4\sqrt{5})^{1/2}/2}{((1-(-\sqrt{5}/2))^2+(1/2)^2)^{1/2}}=\sqrt{5}-2\approx 0.2361.$$

If we are now interested in real values of ω only, the corresponding optimum semiconvergent scheme is found by the same algorithm as before applied to $\tilde{H}'$ which is the hull of $\tilde{\sigma}(T)$ symmetric wrt the real axis. $\tilde{H}'$ is the segment with end points $(-1,-1)$ and $(-1,1)$. Thus the center of the optimum capturing circle is at the point $K(c,0)$ (see Fig. 2), where $K\hat{P}_2A = 90°$. It is readily found that $c=-3/2$ and $R=(KP_2)=\sqrt{5}/2$. Thus

$$\omega'_{\text{opt}}=\frac{(1-(-3/2))+i\,0}{(1-(-3/2))^2+0^2}=\frac{2}{5}$$

and

$$\gamma(T_{\omega'_{\text{opt}}})=\frac{\sqrt{5}/2}{((1-(-3/2))^2+0^2)^{1/2}}=\frac{\sqrt{5}}{5}\approx 0.4472.$$

As is seen $\gamma(T_{\omega_{\text{opt}}})$ is much better than $\gamma(T_{\omega'_{\text{opt}}})$, something which was expected.

B. *Optimum* GSOR *method.* For all the examples we shall give below it will be accepted that Assumptions III and IV of § 3 hold with $p=2$.

i) Let $T\in\mathbb{C}^{6,6}$ with $\sigma(T)\equiv\{\pm 1, \pm(1+i), \pm\sqrt[4]{2}(1+i)\}$. Since 1 is a simple eigenvalue of T, index $(I-T)=1$. It is obvious that the pair $\pm\sqrt[4]{2}(1+i)$ constitutes a critical eigenvalue-pair because the other two eigenvalues are interior points of the same line segment with end points the images of the numbers $\pm\sqrt[4]{2}(1+i)$. We try therefore to apply the results (3.12) of Theorem 3.4. Thus we have

$$\omega_{\text{opt}}=\frac{2}{1+(1-(\sqrt[4]{2}(1+i))^2)^{1/2}}=\frac{2}{1+\sqrt{2}-i}=\frac{\sqrt{2}}{2}+\frac{(2-\sqrt{2})}{2}\,i$$

and

$$\gamma(\mathscr{L}_{\omega_{\text{opt}}})=|\omega_{\text{opt}}-1|=\sqrt{2}-1\approx 0.4142.$$

Since the eigenvalues of T of modulus different from 1 do not all satisfy the restriction $|\text{Re}\,\lambda_j|<1$, an optimum real ω by using the algorithm by Young and Eidson cannot be obtained.

ii) Let $T\in C^{5,5}$ with $\sigma(T)\equiv\{\pm 1, 0, \pm(0.6+0.8i)\}$. Since index$(I-T)=1$ and the eigenvalues $\pm(0.6+0.8i)$ constitute obviously a critical eigenvalue-pair we work as before in i). Consequently

$$\omega_{\text{opt}}=\frac{2}{1+(1-(0.6+0.8i)^2)^{1/2}}=\frac{2}{1+1.2-0.4i}=\frac{22}{25}+\frac{4}{25}\,i$$

and

$$\gamma(\mathscr{L}_{\omega_{\text{opt}}})=|\omega_{\text{opt}}-1|=\tfrac{1}{5}=0.2.$$

Now because all other eigenvalues of T, except ± 1, satisfy the requirement $|\text{Re}\,\lambda_j|<1$ the algorithm by Young and Eidson (in fact a simplified version of it) can be applied. Some of the corresponding optimum results are given in [10, Table 4.1, pp. 198-199]. Thus

$$\omega'_{\text{opt}}=0.74913,\quad \gamma(\mathscr{L}_{\omega'_{\text{opt}}})=0.69118\quad\text{and}\quad \hat{M}_r=0.70737.$$

Comparing the values $\gamma(\mathscr{L}_{\omega_{\text{opt}}})=0.2$ with the value $\gamma(\mathscr{L}_{\omega'_{\text{opt}}})=0.69118$ we see that the former is much better than the latter.

iii) Let $T\in\mathbb{C}^{8,8}$ with $\sigma(T)\equiv\{\pm 1, \pm 1/2, \pm i\sqrt{6}/2\}$, where the eigenvalues ± 1 are double ones, and index $(I-T)=1$. Because the other eigenvalues are two opposite

pairs of real and purely imaginary numbers satisfying $|\mathrm{Re}\,\lambda_j|<1$ and it is obvious no critical eigenvalue-pair exists for all possible ω's, Theorem 3.4 cannot be applied. Thus we turn our attention to real ω's. It is readily obtained (see [10, pp. 194-195]) that $\hat{M}_r=1/2$ and $\hat{M}_i=\sqrt{6}/2$. Therefore applying formulas (3.13) and (3.14) we have

$$\omega_{\mathrm{opt}}=\frac{2}{1+(1-(1/2)^2+(\sqrt{6}/2)^2)^{1/2}}=\frac{4}{5}$$

and

$$\gamma(\mathcal{L}_{\omega_{\mathrm{opt}}})=\left(\frac{1/2+\sqrt{6}/2}{1+(1-(1/2)^2+(\sqrt{6}/2)^2)^{1/2}}\right)^2=\frac{7+2\sqrt{6}}{25}\approx 0.4760.$$

iv) Let $T\in\mathbb{C}^{6,6}$ with $\sigma(T)\equiv\{\pm1,0\}$, where all eigenvalues are double ones, and index $(I-T)=1$. Again either Theorem 3.4 with critical eigenvalue-pair with the eigenvalue zero or the simplified version of the algorithm by Young and Eidson of § 3 can be applied with $\hat{M}_r=\hat{M}_i=0$. In either case the optimum results are the same and lead to the GGS iterative scheme. More specifically

$$\omega_{\mathrm{opt}}=\frac{2}{1+(1-0)^{1/2}}=1 \quad \text{and} \quad \gamma(\mathcal{L}_{\omega_{\mathrm{opt}}})=|\omega_{\mathrm{opt}}-1|=0!$$

v) Let $T\in\mathbb{C}^{10,10}$ with $\sigma(T)\equiv\{\pm1,\pm0.2\pm0.3i,\pm0.6\pm0.4i\}$ and $I-T$ is an irreducible M-matrix. In such a situation we are in the case of the Remark of § 3 so that the algorithm by Young and Eidson applies. Here $\tilde{H}$ is the rectangle with vertices $(\pm0.6,\pm0.4)$ and since there is only one point in the first quadrant we obtain some of the optimum results directly from [10, Table 4.1, pp. 198-199]. These are the following

$$\omega_{\mathrm{opt}}=0.97150, \quad \gamma(\mathcal{L}_{\omega_{\mathrm{opt}}})=0.51626 \quad \text{and} \quad \hat{M}_r=0.69876.$$

Acknowledgment. The author is most indebted to the referee for valuable comments and suggestions.

REFERENCES

[1] A. BERMAN AND R. J. PLEMMONS, *Nonnegative Matrices in the Mathematical Sciences*, Academic Press, New York, 1979.

[2] J. J. BUONI, M. NEUMANN AND R. S. VARGA, *Theorems of Stein-Rosenberg Type* III, *The singular case*, Linear Algebra Appl. 42 (1982), pp. 183-198.

[3] J. J. BUONI AND R. S. VARGA, *Theorems of Stein-Rosenberg type*, in Numerical Mathematics, R. Ansorge, K. Glanshoff and B. Werner, eds., ISNM 49, Birkhäuser Verlag, Basel, 1979, pp. 65-75.

[4] ———, *Theorems of Stein-Rosenberg Type* II, *Optimum paths of relaxation in the complex domain*, in Elliptic Problem Solvers, M. H. Schultz, ed., Academic Press, New York, 1981, pp. 231-240.

[5] A. HADJIDIMOS, *The extrapolation technique as a preconditioning strategy*, in Preconditioning Methods, Theory and Applications, D. J. Evans, ed., Gordon and Breach, London, 1983, pp. 47-67.

[6] ———, *The optimal solution of the extrapolation problem of a first order scheme*, Internat. J. Comput. Math., 13 (1983), pp. 153-168.

[7] ———, *The optimal solution to the problem of complex extrapolation of a first order scheme*, Linear Algebra Appl., (in press).

[8] B. KREDELL, *On complex successive overrelaxation*, BIT, 2 (1962), pp. 143-152.

[9] R. S. VARGA, *Matrix Iterative Analysis*, Prentice-Hall, Englewood Cliffs, NJ, 1962.

[10] D. M. YOUNG, *Iterative Solution of Large Linear Systems*, Academic Press, New York, 1971.

[11] D. M. YOUNG AND H. EIDSON, *On the determination of the optimum relaxation factor for the* SOR *method when the eigenvalues of the Jacobi method are complex*, Report CNA-1, Center for Numerical Analysis, Univ. Texas, Austin, 1970.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 4, October 1985

004

IMPLEMENTATION OF A DOUBLE-BASIS SIMPLEX METHOD FOR THE GENERAL LINEAR PROGRAMMING PROBLEM*

P. E. PROCTOR†

Abstract. The basis handling procedures of the simplex method are formulated in terms of a "double basis." The basis is expressed as a matrix product, one of the factors being the basis matrix of the last refactorization. Forward and backward transformations and update are presented for each of two implementations of the double-basis method. The double-basis update is restricted to a matrix of dimension limited by the refactorization frequency and two permutation matrices. This can lead to a saving in storage space and updating time. The cost is that the time for the forward and backward transformations is about double.

Computational comparisons of storage and speed are made with the standard simplex method on problems of up to 1,480 constraints. Generally, the double-basis method performs best on larger, denser problems. Density seems to be the more important factor, and the problems with large nonzero growth between refactorizations are the better ones for the double-basis method. Storage saving in the basis inverse representation versus the standard method is as high as 36%, whereas the double-basis run times are 1.2 or more times greater.

AMS(MOS) subject classifications. 65K05, 90C05, 90C06

1. Introduction. Suppose that in solving a linear program of m constraints, the basis inverse is refactored every r iterations. Then at any given iteration, the basis matrix differs from the basis matrix of the last refactorization in no more than r/m columns. This redundancy can be exploited to avoid updating the full basis inverse at every iteration.

This paper presents computational experience with a general algorithm for linear programming that uses the update alluded to in the first paragraph. For historical reasons this algorithm is called a "double-basis" method.

Double-basis methods have appeared in various contexts since 1955 when Dantzig [8] suggested separating the block-triangular part of an LP coefficient matrix. Another, related paper [6] is by Beale. His pseudo-basic variable method for problems with coupling variables is equivalent to a double-basis factorization. Aonuma [1]–[5] and Marsten and Shepardson [13] use double-basis methods to decompose problems along the lines of "hard" and "easy" bases.

Bisschop and Meeraus [7] and Kallio [11] discuss the double-basis factorization in the general context. Bisschop and Meeraus give estimates for storage requirements. Kallio presents some details of a proposed implementation involving a product-form representation of one of the basis factors.

Section 2 presents the double-basis method of this paper. Two implementations of it were tested. These are introduced in § 3. Section 4 gives the details of one version, § 5 the other. Experimental results form the subject matter of § 6. Finally, conclusions are drawn in § 7.

2. The double-basis method. Let the basis matrix at the last refactorization be $\bar{B}$. Let B be the current basis matrix. Since $\bar{B}$ and B have many columns in common, we obtain mostly standard unit vectors for the columns of the product $\bar{B}^{-1}B$. More exactly,

$$\bar{B}^{-1}B = P\begin{bmatrix} G & 0 \\ H & I \end{bmatrix}Q,$$

* Received by the editors July 12, 1982, and in final revised form April 2, 1984. This paper was presented at the SIAM Conference on Applied Linear Algebra, Raleigh, North Carolina, April 26-29, 1982.

† Bell Communications Research, Red Bank, New Jersey 07701.

where I is the identity, P and Q are permutation matrices, and the dimension of G is at most the number of iterations since the last refactorization.

It turns out that the simplex method can be performed using only $\bar{B}$, G, P, and Q. Specifically, this reduces the update to that of the small matrix G, and the revision of P and Q. However, one pays the price of four matrix multiplies with $\bar{B}^{-1}$ in each iteration, compared with two B^{-1} multiplies in the standard simplex method.

The double-basis method distinguishes four types of columns. The type of a column can change from iteration to iteration. At any given iteration we have the following partition:

B_p—those columns in $\bar{B}$, but not in B.

B_d—those columns in both $\bar{B}$ and B.

B_q—those columns in B, but not in $\bar{B}$.

B_r—all other columns.

Due to this partition we have four possible cases for the basis exchange (e and l are the entering and leaving columns, respectively):

I. $e \in B_p$, $l \in B_q$. Since the basis is regaining a column lost since the last refactorization, G decreases in dimension by one.

I. $e \in B_r$, $l \in B_q$. G stays the same size in this case.

III. $e \in B_p$, $l \in B_d$. G stays the same size.

IV. $e \in B_r$, $l \in B_d$. G increases in dimension by one.

The forward and backward multiplies with the basis inverse, B, are computed as follows:

$y = B^{-1}e$(FTRAN)

1) $\tilde{z} = \bar{B}^{-1}e$.

2) $\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = P^T\tilde{z}$.

3) $\tilde{y}_1 = G^{-1}z_1$.

4) $\begin{bmatrix} 0 \\ \tilde{y}_2 \end{bmatrix} = P^T\bar{B}^{-1}(e - B_q\tilde{y}_1)$.

5) $y = Q^T\tilde{y}$.

Our research did not examine the possiblity of using the computed zeros in 4) to monitor numerical error.

$u = c_B B^{-1}$ (BTRAN)

1) $(\tilde{c}_1 | \tilde{c}_2) = c_B Q^T$, where $\tilde{c}_1$ has dimension equal to that of G.

2) $m = \tilde{c}_1 - (0 | \tilde{c}_2)P^T\bar{B}^{-1}B_q$.

3) $u = (mG^{-1} | \tilde{c}_2)P^T\bar{B}^{-1}$.

3. Implementation. The double-basis method was implemented in two versions. The first keeps G^{-1} as an explicit matrix, and the second uses Reid's LA05 software (17) to handle an LU factorization of G. Both implementations use Reid's LU software for the matrix $\bar{B}$. Also, both implementations employ Marsten's XMP package [12] to perform the higher-level functions of the simplex method.

Note that the matrix H is nowhere used in the double-basis method computing. However, its counterpart in the problem matrix, B_q, is. B_q is retrieved from the raw problem data via the knowledge of which columns are basic, and information provided by the permutation Q.

4. Update of G^{-1}—explicit version.

Case I. The departing row and column of G are permuted to the border of the matrix, and a bordered-matrix technique is used.

Case II. G^{-1} is multiplied by an elementary column matrix corresponding to the column exchange.

Case III. G^{-1} is multiplied by an elementary row matrix, since rows in G and H are exchanged (corresponding to an exchange of identity columns between B_p and B_d).

Case IV. A row and column are appended to G, and G^{-1} is updated by a bordered-matrix technique.

5. Update of G—LU version.

Reduction to two cases. No easy way to implement the case I and III updates could be found for the LU representation of G. Therefore, case I was absorbed into case II, and case III into case IV. As a result, an identity column is introduced into G each time cases I or III occur. In our experiments these cases did not occur too frequently. (See Table 1.)

TABLE 1
Basis-exchange cases—explicit version. Entries are percentages.

Problem	Entering variable same as leaving variable	Case I	Case II	Case III	Case IV
SCTAP3	0	0	16	0	83
SCTAP2	0	1	17	0	82
SCFXM3	0	2	19	3	75
BP1	0	1	53	0	45
PILOT.WELFARE	4	2	47	1	47
SCFXM2	0	4	23	4	70
SCRS8	0	2	13	5	80
SCAFR25	0	5	23	5	66
SCSD8	0	0	23	0	76
STAIR	0	6	30	7	56
STAIR (CYBER)	0	5	37	4	54
SCFXM1	0	9	27	7	57
SCTAP1	0	3	18	5	73
SC205	0	1	9	0	88
SCSD6	0	1	24	1	74
SCAGR7	0	4	19	7	70
SCSD1	0	2	25	2	71

TRAN *simplification.* The reduction to two cases results in simplification of the TRAN computations. In the two-case situation, $P = Q^T$. Consequently, successive applications of P^T and Q^T can be eliminated since these two matrices are now inverses of each other. With this elimination G^{-1} becomes the only matrix to which permutations are applied. We have achieved the two following simplifications:

1) Elimination of four permutation applications per simplex iteration.

2) Reduction in storage requirements for the permutations from two arrays of length equal to the problem dimension to one array of length equal to the refactorization frequency.

Update of the G factorization. In Reid's software the L factor is stored as a sequence of pivots. We may visualize the U factor as a triangular matrix. We have $L^{-1}G = U$.

Case II *update.* In Case II G receives a replacement column. This column, z_1, has already been computed in the FTRAN. It is composed of the components of $P^T\bar{B}^{-1}e$ that correspond to the rows of G.

From the LU factorization of G, we have $L^{-1}G_{\text{new}} = U_{\text{spike}}$, where U_{spike} is identical to the old U except for the column corresponding to z_1. In general this new column of U destroys U's triangularity—therefore, it is called a "spike." Reid's software removes this spike from U, possibly adding pivots to L^{-1} in the process.

Case IV *update.* The situation here is very similar to case II, except that we are gaining a column and a row. They are visualized as being prefixed to G, thus:

$$G_{\text{new}} = \begin{bmatrix} z_R & t \\ z_1 & G \end{bmatrix}.$$

Premultiplying by L^{-1}, we obtain

$$\begin{bmatrix} 1 & 0 \\ 0 & L^{-1} \end{bmatrix}\begin{bmatrix} z_R & t \\ z_1 & G \end{bmatrix} = \begin{bmatrix} z_R & t \\ L^{-1}z_1 & U \end{bmatrix}.$$

This multiplication is implicit since we already have all the quantities on the right.

It is seen that the right-hand matrix is triangular with a spike in the first column. It is passed to Reid's software which returns the updated factorization.

6. Experimental results. Both versions of the double-basis simplex method were compared to the standard simplex method on VAX 11/780 and CYBER 175 computers. XMP was used for the testing because its modular structure enabled simple substitution of FORTRAN subroutines. Special double-basis subroutines were written for the forward and backward transformations and the update.

Seventeen problems were employed for testing (Table 2). All of them except HELSI2 are staircase problems [10], [18]. More results can be found in [16].

TABLE 2
Test problems.

Problem	Rows	Columns*	Nonzeros*	Density (%)
SCTAP3	1,480	3,960	10,354	0.18
SCTAP2	1,0906	2,970	7,804	0.24
SCFXM3	990	2,361	8,767	0.38
BP1	821	2,392	11,221	0.57
PILOT.WELFARE	722	3,591	10,384	0.40
SCFXM2	660	1,574	5,843	0.56
SCRS8	490	1,659	3,672	0.45
SCAGR25	471	971	2,025	0.44
SCSD8	397	3,147	8,981	0.72
STAIR	356	829	4,230	1.43
SCFXM1	330	787	2,919	1.12
SCTAP1	300	780	1,992	0.85
SC205	205	408	756	0.90
SCSD6	147	1,497	4,463	2.03
SCAGR7	129	269	549	1.58
SCSD1	77	837	2,465	3.82
HELSI2	57	608	3,664	10.57

* Included is one entry per row for an identity matrix.

All problems were run with refactorization frequency equal to 51. Zero tolerance was 10^{-10}, and pivot tolerance was 10^{-6}, except for:PILOT.WELFARE (10^{-12}, 10^{-7}) and STAIR (CYBER) and HELSI2 (CYBER) (LU double-basis pivot 10^{-4}). On the whole, more numerical problems were encountered with the double-basis method, but in certain instances it behaved better than the standard method.

BP1 and PILOT.WELFARE were started from feasible bases, all others from slack bases.

The number of iterations per second was logged for each test. Table 3 gives the ratio of that rate for the standard method to that for the LU version of the double-basis method, which was always faster than the explicit version.

TABLE 3
Ratios of standard rates to LU rates.

Problem	Ratio
STAIR (CYBER)	1.19
STAIR	1.24
PILOT.WELFARE	1.41
SCAGR25	1.41
BP1	1.48
SCFXM3	1.49
SCFXM2	1.51
SCTAP2	1.54
SCTAP3	1.58
SCRS8	1.58
SCFXM1	1.60
SC205	1.61
SCSD8	1.63
SCTAP1	1.73
HELSI2	1.8
SCSD6	1.82
SCAGR7	1.85

7. Conclusions. A comparison of storage requirements for the basis inverse representation between the standard and LU double-basis methods is shown in Table 4. Neither method is the winner. However, storage requirements could be more predictable for the double-basis method because nonzero growth after refactorization is limited to the small G matrix.

The results of Table 3 are repeated in Table 5 which also presents measures of problem size and density. Average column length is shown instead of density because it is independent of the number of rows. Also, average column length in the optimal basis might serve as a better indicator of the densities truly encountered during the simplex iterations. From the table it appears that density is more imortant than problem size in determining the performance of the double-basis method.

With sparse matrix techniques execution times for the forward and backward transformations (TRAN's) on particular problems are roughly proportional to the number of nonzeros involved in their computation. Some information along this line was collected during testing, and it is analyzed here. First, a calculation is performed.

The number of nonzeros handled between refactorizations in both the standard simplex and double-basis simplex methods is estimated, and a ratio of the two numbers is formed. Let a be the number of nonzeros in the basis inverse representation immediately following refactorization. We will assume that both methods follow the

TABLE 4
Differences in storage requirements for the standard method and the LU version of the double-basis method (in full words).

Problem	Maximum required by standard	Standard − *LU*	(%)
SCTAP3	25,411	−1,777	−7.0
SCTAP2	19,120	−1,420	−7.4
SCFXM3	23,807	−2,144	−9.0
BP1	28,818	1,824	6.3
PILOT.WELFARE	29,433	4,020	13.7
SCFXM2	16,346	−290	−1.8
SCRS8	11,719	1,703	14.5
SCAGR25	17,369	6,280	36.2
SCSD8	12,130	−616	−5.1
STAIR	26,604	3,837	14.4
SCFXM1	8,297	−1,142	−13.8
SCTAP1	5,939	−653	−11.0
SC205	5,380	−46	−0.9
SCSD6	4,493	−173	−3.9
SCAGR7	4,451	1	0.0

TABLE 5
Results comparison. The last two columns give average column lengths.

Problem	*LU*/Std	Rows	Entire problem	Optimal basis
STAIR (CYBER)	1.19	356	5.1	10.1
STAIR	1.24	356	5.1	10.1
PILOT.WELFARE	1.405	722	2.9	4.3
SCAGR25	1.411	471	2.1	2.7
BP1	1.48	821	4.7	5.3
SCFXM3	1.49	990	3.7	4.1
SCFXM2	1.51	660	3.7	4.1
SCTAP2	1.54	1,090	2.6	2.3
SCTAP3	1.576	1,480	2.6	2.4
SCRS8	1.580	490	2.2	2.6
SCFXM1	1.60	330	3.7	4.0
SC205	1.61	205	1.9	2.6
SCSD8	1.63	397	2.9	2.8
SCTAP1	1.73	300	2.6	2.6
HELSI2 (CYBER)	1.8	57	6.0	6.4
SCSD6	1.82	147	3.0	2.6
SCAGR7	1.85	129	2.0	3.0

same path to optimality, so that a is invariant between the two methods. Let b be the number of nonzeros gained by the basis inverse representation in the standard method at each iteration. We assume that b is a constant. Let c be the number of iterations between refactorizations.

We will also make two assumptions regarding the G matrix: First, G^{-1} is represented explicitly, and each element is handled as a nonzero. Second, G increases in dimension by one at each iteration (not true in general—see below).

We obtain that the standard simplex method handles

$$\sum_{i=0}^{c-1} (a+bi) = ac + \frac{(c-1)cb}{2}$$

nonzeros in matrix multiplies between refactorizations. Given that each double-basis method TRAN requires two applications of $\bar{B}^{-1}$ and one of G^{-1} (see below), we handle

$$a + \sum_{i=1}^{c-1} (a+i^2+a) = (2c-1)a + \frac{(c-1)c(2c-1)}{6}$$

nonzeros between refactorizations.

Forming the ratio of the number of double-basis nonzeros to that of the standard method, we have

$$R = \frac{(2c-1)[12a + c(c-1)]}{c[12a + 3b(c-1)]}.$$

Much study could be devoted to the factors influencing this ratio , and also to the plausibility of our previous assumptions. However, our experiments, and others in the literature [9], [14], [15] documenting values for b, indicate that it will probably never be close to 1.0 for sparse problems. This is one reason that the double-basis method is more promising for denser problems.

Average times for single simplex method operations were recorded. *LU*-to-standard ratios for these figures are given in Table 6. As was pointed out previously, the theoretical TRAN ratio is over 2.0 when nonzero growth in the standard method is small. As standard method nonzero growth increases, the ratios become more favorable to the double-basis method.

Table 7 gives the average number of nonzeros gained per iteration between basis refactorizations in the standard simplex method. These can serve as estimates for the numbers b. Some sample calculations with b values and sample values of a yielded results similar to the experimentally achieved values of R (Table 6).

TABLE 6
LU/standard ratios of average single operation times.

Problem	BTRAN	FTRAN	Update
STAIR (CYBER)	2.3	2.1	0.39
STAIR	1.9	1.7	0.50
PILOT.WELFARE	2.0	1.9	0.75
SCAGR25	2.1	1.9	0.78
BP1	2.0	1.9	0.77
SCFXM3	2.0	2.1	1.0
SCFXM2	2.1	2.1	1.1
SCTAP2	2.2	2.3	1.2
SCTAP3	2.2	2.2	1.1
SCRS8	2.3	2.1	1.1
SCFXM1	2.4	2.1	1.2
SC205	2.7	2.0	1.6
SCSD8	2.1	2.2	1.4
SCTAP1	2.4	2.5	1.6
SCSD6	2.4	2.6	2.0
SCAGR7	2.7	2.7	1.7

TABLE 7
Average nonzero growth. Average nonzero growth per iteration between refactorizations in the LU representation of the basis matrix in the standard simplex method.

Problem	Average growth
STAIR	27.4
PILOT.WELFARE	29.0
SCAGR25	11.2
BP1	23.4
SCFXM3	10.6
SCFXM2	10.8
SCTAP2	5.3
SCTAP3	5.3
SCRS8	9.2
SCFXM1	8.7
SC205	7.0
SCSD8	12.7
SCTAP1	4.3
SCSD6	6.7
SCAGR7	6.6
SCSD1	6.3

The theoretical TRAN ratio, R, is plotted in Fig. 1 as a function of a and b. Also plotted are the test problems versus their a values at the optimal basis, and their average b values (given in Table 7). The R values thus determined can be compared to the experimental values (Table 6).

The line $R = 2 - 1/c$ is of particular interest in the figure. Below this line R is decreasing in both a and b, while above, it is increasing in a for fixed b. To determine optimal problem locations on the plot as a function of problem size, density, and

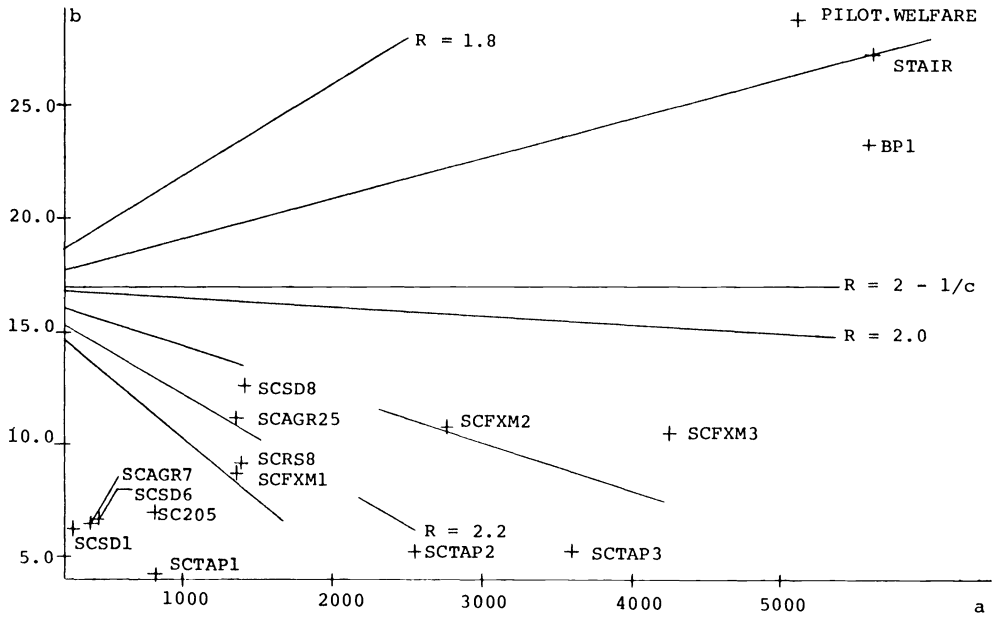


FIG. 1. TRAN *ratio* (R) *vs. refactor nonzeros* (a) *and nonzero growth per iteration* (b).

structure, and to further relate these variables to the performance of the updates, would be necessary for a full comparison of the standard and double-basis methods. This is not to mention the necessity of validating the use of average, or typical, values for a and b.

The double-basis method seems to hold promise for larger, denser problems. More experiments and closer analysis are now needed.

REFERENCES

[1] T. AONUMA, *Two-level approach to dynamically decomposing a linear planning model*, Working paper 26, Kobe Univ. Commerce, Tarumi, Kobe 655, Japan, 1975.

[2] ———, *Hierarchical decomposition in dynamic linear models*, Working paper 31, Kobe Univ. Commerce, Tarumi, Kobe 655, Japan, 1976.

[3] ———, *A nested multi-level planning approach to a multi-period linear planning model*, Working paper 35, Kobe Univ. Commerce, Tarumi, Kobe 655, Japan, 1977.

[4] ———, *A two-level algorithm for two-stage linear programs*, J. Oper. Res. Soc. Japan, 21 (1978), pp. 171-187.

[5] ———, *An extension of the two-level algorithm to optimizing weakly coupled dynamic linear systems*, Working paper 41, Kobe Univ. Commerce, Tarumi, Kobe 655, Japan, 1978.

[6] E. M. L. BEALE, *The simplex method using pseudo-basic variables for structured linear programming problems*, in Recent Advances in Mathematical Programming, R. L. Graves and P. Wolfe, eds., McGraw-Hill, New York, 1963.

[7] J. BISSCHOP AND A. MEERAUS, *Matrix augmentation and partitioning in the updating of the basis inverse*, Math. Programming, 13 (1977), pp. 241-254.

[8] G. B. DANTZIG, *Upper bounds, secondary constraints and block triangularity in linear programming*, Econometrica, 23 (1955), pp. 174-183.

[9] D. GAY, *On comibning the schemes of Reid and Saunders for sparse LP bases*, in Sparse Matrix Proceedings, I. Duff and G. Stewart, eds., Society for Industrial and Applied Mathematics, Philadelphia, 1978.

[10] J. K. HO AND E. LOUTE, *A set of staircase linear programming test problems*, Math. Programming, 20 (1981), pp. 245-250.

[11] M. KALLIO, *On the simplex method using an artificial basis*, Working paper, Helsinki School of Economics, Helsinki, 1979.

[12] R. E. MARSTEN, *The design of the* XMP *linear programming library*, ACM Trans. Math. Software, 7 (1981), pp. 481-497.

[13] R. E. MARSTEN AND F. SHEPARDSON, *A double basis simplex method for linear programs with complicating variables*, Technical report No. 531, Management Information Systems Dept., Univ. Arizona, Tucson, 1978.

[14] R. D. MCBRIDE, *A spike collective dynamic factorization algorithm for the simplex method*, Management Sci., 24 (1978), pp. 1031-1042.

[15] ———, *A bump triangular dynamic factorization algorithm for the simplex method*, Math. Programming, 18 (1980), pp. 49-61.

[16] P. E. PROCTOR, *Double-basis simplex method for large scale linear programming*, Ph.D. dissertation , Univ. Arizona, Tucson, 1982.

[17] J K. REID, FORTRAN *subroutines for handling sparse linear programming bases*, Report AERE-R 8269, AERE Harwell, Oxfordshire, 1976.

[18] F. SHEPARDSON AND R. E. MARSTEN, *A Lagrangean relaxation algorithm for the two duty period scheduling problem*, Management Sci., 26 (1980), pp. 274-281.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 4, October 1985

005

SOME GRAPH-COLOURING THEOREMS WITH APPLICATIONS TO GENERALIZED CONNECTION NETWORKS*

DAVID G. KIRKPATRICK‡, MARIA KLAWE† AND NICHOLAS PIPPENGER†

Abstract. With the aid of a new graph-colouring theorem, we give a simple explicit construction for generalized n-connectors with $2k-1$ stages and $O(n^{1+1/k}(\log n)^{(k-1)/2})$ edges. This is asymptotically the best explicit construction known for generalized connectors.

1. Introduction. Our goals in this paper are twofold. Our first goal is to give a new construction for generalized connectors. Under certain circumstances this new construction is superior to all other known constructions. Our second goal is to use graph-colouring theorems to systematically derive old and new results on networks.

For the purposes of this introduction, we shall give some informal definitions. We shall give more precise and more general definitions in the next section. An *n-network* is an acyclic directed graph with n distinguished vertices called *inputs* and n other distinguished vertices called *outputs*. We shall be concerned with the minimum possible *size* (number of edges) and *depth* (number of edges in the longest path from an input to an output) that n-networks with various connectivity properties can possess. An *n-connector* is an n-network such that, for any one-to-one correspondence between certain inputs and distinct outputs, there exist vertex-disjoint paths joining each chosen input to the corresponding output. A *generalized n-connecter* is an n-network such that, for any one-to-many correspondence between certain inputs and disjoint sets of outputs, there exist vertex-disjoint trees joining each chosen input to the corresponding set of outputs. An *n-crossbar* is an n-network with depth 1 and size n^2, with an edge joining each input to each output. For both of the problems considered here, a crossbar provides a solution with small depth and large size. Our goal is to find alternate solutions with larger but limited depth and smaller size.

Let $f(n)$ denote the minimum possible size of an n-connector. It has long been known (see Beneš [2]) that $f(n)=O(n\log n)$ and $f(n)=\Omega(n\log n)$. (For the sharpest known estimates, see Pippenger [12] for the upper bound and Pippenger [13] for the lower bound.)

Let $g(n)$ denote the minimum possible size of a generalized n-connector. It was shown by Ofman [10] that $g(n)=O(n\log n)$ and by Pippenger [11] that $g(n)=f(n)+O(n)$. The first of these results is proved by an extension of the explicit construction used to show that $f(n)=O(n\log n)$. (For the best explicit construction known, see Dolev et al. [4].) The second result was established by a probabilistic construction. It is now possible to replace this by an explicit construction (see Gabber and Galil [5]), but in any case the constants involved are so large as to render the result completely impractical. (For the best probabilistic construction known, see Bassalygo [1].)

Let $f_k(n)$ denote the minimum possible size of an n-connector with depth at most k. It was shown by Pippenger and Yao [14] that

$$f_k(n)=O(n^{1+1/k}(\log n)^{1/k})$$

and that

$$f_k(n)=\Omega(n^{1+1/k}).$$

* Received by the editors May 14, 1984.
† IBM Research Laboratory, San Jose, California 95193.
‡ University of British Columbia, Vancouver, British Columbia, V6T 1W5, Canada.

(Throughout this paper, the constants implicit in the notation $O(\cdots)$ are independent of n, but may depend on k.) The upper bound here is proved by a probabilistic construction; the best explicit construction known gives

$$f_{2k-1}(n) = O(n^{1+1/k})$$

(see Pippenger [12]).

Let $g_k(n)$ denote the minimum possible size of a generalized n-connector with depth at most k. Dolev et al. [4] showed that

$$g_k(n) = O((n \log n)^{1+1/k})$$

by a probabilistic construction; they also showed that

$$g_{3k-2}(n) = O(n^{1+1/k})$$

by an explicit construction. Masson and Jordan [8] and Nassimi and Sahni [9] showed that

$$g_3(n) = O(n^{5/3})$$

by two quite different explicit constructions. Attempts to extend these constructions to depths greater than 3 do not give results competitive with the construction for depth $3k-2$ mentioned above.

In this paper we shall show that

$$g_3(n) = O(n^{3/2}(\log n)^{1/2})$$

by an explicit construction. We extend this to

$$g_{2k-1}(n) = O(n^{1+1/k}(\log n)^{(k-1)/2}),$$

which differs merely by logarithmic factors from the corresponding bound for $f_{2k-1}(n)$.

It seems unlikely that the construction of the present paper will be useful in practice; to compare it with the competing constructions of Masson and Jordan [8], Nassimi and Sahni [9] and Dolev et al. [4], we observe that $n^{3/2}(\log_2 n)^{1/2} = n^{5/3}$ for n about 10^3, $n^{4/3} \log n = n^{3/2}$ for n about 6×10^8 and $n^{5/4}(\log n)^{3/2} = n^{4/3}$ for n about 6×10^{37}. Nor is this result asymptotically the best; probabilistic constructions give sharper upper bounds for all fixed depths. We can, however, say that it is asymptotically the best explicit construction known, when the depth is fixed and the number of inputs and outputs is large.

Edge-colouring in bipartite graphs provides a vivid and convenient language for discussing connectors and their control algorithms. This relationship was observed by Lev, Pippenger and Valiant [7] (who used it to describe parallel control algorithms) and later by Hwang [6] (who did not, however, give any new results). We extend this method in the present paper by using hyperedge-colouring of bipartite hypergraphs to discuss generalized connectors. For the application of yet another combinatorial colouring problem to networks, see Dolev et al. [4].

2. Networks and graph-colouring problems. A (p, q)-*network* is an acyclic digraph with p distinguished vertices called *inputs* and q other distinguished vertices called *outputs*. Vertices that are neither inputs nor outputs will be called *links*.

A *request* is a pair comprising an input and an output. An *assignment* is a set of requests, no two of which have an input or output in common. A *generalized assignment* is a set of requests, no two of which have an output in common.

A *route* is a directed path from an input to an output. A *state* is a set of routes, no two of which have a vertex in common. A *generalized state* is a set of routes, any two of which have at most an initial segment of their vertices in common.

An assignment or generalized assignment is *realized* by a state or generalized state, respectively, if, for each request in the assignment, there is a route in the state from the input of the request to the output of the request.

A (p, q)-*connector* is a (p, q)-network for which every assignment is realized by a state. Let $f_k(p, q)$ denote the minimum possible size of a (p, q)-connector with depth at most k. A *generalized* (p, q)-*connector* is a (p, q)-network for which every generalized assignment is realized by a generalized state. Let $g_k(p, q)$ denote the minimum possible size of a generalized (p, q)-connector with depth at most k. A (p, q)-*crossbar* is a (p, q)-network with depth 1 and size pq, with an edge joining each input to each output.

We shall be particularly concerned with the three-stage construction shown in Fig. 1. The first stage comprises r (a, c)-subnetworks, the second stage comprises c (r, s)-subnetworks and the third stage comprises s (c, b)-subnetworks. An output of each subnetwork in the first stage is identified with an input of each subnetwork in the second stage to form a link and an output of each subnetwork in the second stage is identified with an input of each subnetwork in the third stage to form a link. The resultant is a (p, q)-network, where $p = ar$ and $q = bs$.

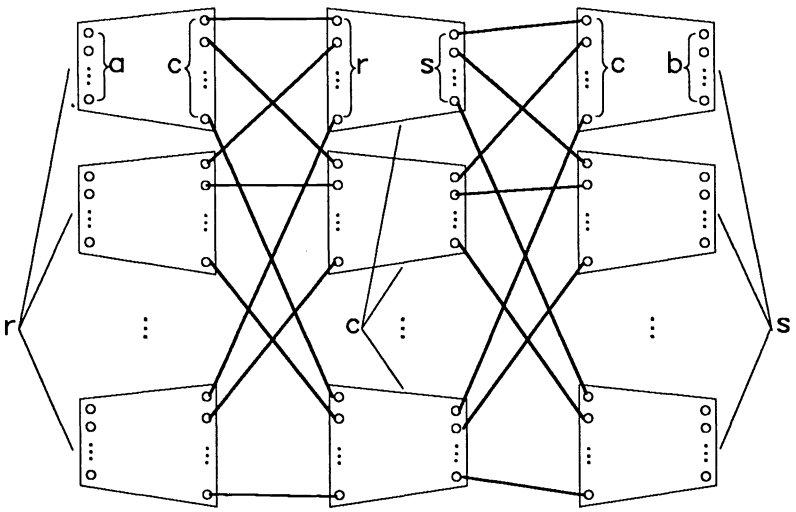


FIG. 1. *The interconnection of subnetworks in three stages to form a network. The thick lines joining subnetworks represent identifications of inputs and outputs to form links, not edges.*

We shall be concerned with conditions under which, if various subnetworks are connectors or generalized connectors, the resultant is a connector or generalized connector. We shall systematically obtain such conditions by reducing them to graph-colouring problems.

An (r, s, a, b)-*graph* is a bipartite graph (R, S, E) with vertices $R = \{1, \cdots, r\}$ and $S = \{1, \cdots, s\}$ and (possibly multiple) edges E such that at most a edges are incident with each vertex in R and at most b edges are incident with each vertex in S.

A c-*colouring* of a bipartite graph is an assignment of the colours $C = \{1, \cdots, c\}$ to the edges of the graph such that the edges incident with any vertex are assigned distinct colours.

An (r, s, a, b)-*hypergraph* is a bipartite hypergraph (R, S, E) with vertices $R = \{1, \cdots, r\}$ and $S = \{1, \cdots, s\}$ and (possibly multiple) hyperedges E such that each hyperedge is incident with exactly one vertex in R, at most a hyperedges are incident with each vertex in R and at most b hyperedges are incident with each vertex in S.

A *c-colouring* of a bipartite hypergraph is an assignment of the colours $C = \{1, \cdots, c\}$ to the hyperedges of the graph such that the hyperedges incident with any vertex are assigned distinct colours.

A *c-hypercolouring* of a bipartite hypergraph (R, S, E) is an assignment of subsets of the colours $C = \{1, \cdots, c\}$ to the hyperedges of the graph such that the hyperedges incident with any vertex in R are assigned disjoint sets of colours and the hyperedges incident with any vertex in S are assigned sets of colours possessing a system of distinct representatives.

The reductions of network conditions to graph-colouring problems are very trite. Rather than present four such reductions in detail, we shall describe the paradigm once. Given an assignment or generalized assignment, we construct a graph or hypergraph by associating a vertex in R with each subnetwork in the first stage, a vertex in S with each subnetwork in the third stage and an edge or hyperedge in E with each request or maximal set of requests having an input in common. We then apply a graph-colouring theorem to obtain a colouring or hypercolouring of this graph or hypergraph. We associated a colour in C with each subnetwork in the second stage. From the colouring or hypercolouring, we construct assignments or generalized assignments for the subnetworks. By hypothesis, there are states or generalized states realizing these assignments or generalized assignments. We then patch together these states or generalized states to obtain a state or generalized state for the resultant. In each case, the graph-colouring theorem is a transparent paraphrase of the network condition.

3. Some graph-colouring theorems. The following proposition is perhaps the oldest result concerning connecting networks (see Beneš [2, Thm. 3.1]).

PROPOSITION 1. *If all the subnetworks are connectors, then the resultant is a connector if*

$$c \geqq \max\{a, b\}.$$

This proposition reduces to the following well-known graph-colouring theorem (see Berge [3, Chap. 12, Thm. 2]).

THEOREM 2. *Every (r, s, a, b)-graph has a c-colouring if*

$$c \geqq \max\{a, b\}.$$

Taking all subnetworks in Proposition 1 to be crossbars and choosing $a = b = r = s = \lceil n^{1/2} \rceil$ yields n-connectors with depth 3 and size $O(n^{3/2})$. Taking the subnetworks in the first and third stages to be crossbars and constructing the subnetworks in the second stage recursively yields n-connectors with depth $2k-1$ and size $O(n^{1+1/k})$. This is asymptotically the best explicit construction known for connectors with limited depth.

The following proposition is due to Masson and Jordan [8].

PROPOSITION 3. *If the subnetworks in the first and third stages are generalized connectors and those in the second stage are connectors, then the resultant is a generalized connector if*

$$c \geqq \max\{as, b\}.$$

This proposition reduces to Theorem 2 in the same manner as Proposition 1. Taking all subnetworks to be crossbars and choosing $a = s = \lceil n^{1/3} \rceil$ and $b = r = \lceil n^{2/3} \rceil$ yields generalized n-connectors with depth 3 and size $O(n^{5/3})$.

PROPOSITION 4. *If the subnetworks in the second and third stages are generalized connectors and those in the first stage are connectors, then the resultant is a generalized connector if*

$$c \geqq a + (b(b-1)s)^{1/2}.$$

This proposition reduces to the following theorem.

THEOREM 5. *Every (r, s, a, b)-hypergraph has a c-colouring if*

$$c \geqq a + (b(b-1)s)^{1/2}.$$

Proof. Let G be an (r, s, a, b)-hypergraph. Let G^* be the hypergraph containing those hyperedges of G that are incident with more than $(bs/(b-1))^{1/2}$ vertices in S. There are at most $(b(b-1)s)^{1/2}$ hyperedges in G^*. Let each hyperedge in G^* be assigned a distinct colour. It will suffice to show that this c-colouring of G^* can be extended to a c-colouring of G.

We shall prove this by induction on the number of hyperedges that are in G but not in G^*. If there are no such hyperedges, then we are done. If there is at least one such hyperedge, let G' be a hypergraph obtained by deleting one such hyperedge H from G. By inductive hypothesis, the c-colouring of G^* can be extended to a c-colouring of G'. It will suffice to show that it can be extended to a c-colouring of G.

Let τ be a vertex in R or S. Let us say that a colour is *good* at τ if it is not assigned to a hyperedge incident with τ in G'. Let H be incident with the vertex ρ in R. All but at most $a-1$ colours are good at ρ. The hyperedge H is incident with at most $(bs/(b-1))^{1/2}$ vertices in S. For each such vertex σ, all but at most $b-1$ colours are good at σ. Since $c > (a-1) + (b(b-1)s)^{1/2}$, at least one colour is good at every vertex with which H is incident. Assigning this colour to H yields a c-colouring of G. □

Taking all subnetworks in Proposition 4 to be crossbars and choosing $a = s = \lceil n^{2/3} \rceil$ and $b = r = \lceil n^{1/3} \rceil$ again yields generalized n-connectors with depth 3 and size $O(n^{5/3})$.

Remark. It is not hard to see that Theorem 5 is, to within a constant factor, the best possible. Clearly at least a colours may be necessary. We shall show that at least $\frac{1}{2}(b(b-1)s)^{1/2}$ colours may be necessary. It will follow that at least $\max\{a, \frac{1}{2}(b(b-1)s)^{1/2}\} \geqq \frac{1}{2}(a + (b(b-1)s)^{1/2})$ colours may be necessary.

To show that at least $\frac{1}{2}(b(b-1)s)^{1/2}$ colours may be necessary, it will suffice to construct a hypergraph (V, E) with at most s vertices in V and at least $\frac{1}{2}(b(b-1)s)^{1/2}$ hyperedges in E such that there are at most b hyperedges incident with any vertex and every pair of hyperedges is incident with a common vertex. It will in fact suffice to construct a hypergraph (V, H) with at most s vertices in V and at least $s^{1/2}$ hyperedges in H such that there are at most 2 hyperedges incident with any vertex and every pair of hyperedges is incident with a common vertex, for then we may take E to contain $\lfloor b/2 \rfloor \geqq \frac{1}{2}(b(b-1))^{1/2}$ copies of each hyperedge in H.

Let $t = \lceil s^{1/2} \rceil$. Then $t(t-1)/2 \leqq s$. Let K be a complete graph with t vertices and $t(t-1)/2$ edges. Construct (V, H) by associating a vertex in V with each edge in K and a hyperedge in H with each vertex in K (a vertex in V is incident with a hyperedge in H if and only if the associated edge is incident with the associated vertex in K). □

PROPOSITION 6. *If all subnetworks are generalized connectors, then the resultant is a generalized connector if*

$$c \geqq (a-1)\lfloor \log_2 (2s) \rfloor + 2b.$$

This proposition reduces to the following theorem.

THEOREM 7. *Every (r, s, a, b)-hypergraph has a c-hypercolouring if*

$$c \geq (a-1)\lfloor \log_2 (2s) \rfloor + 2b.$$

Proof. Let G be an (r, s, a, b)-hypergraph. We shall consider two cases. If $\lfloor \log_2 (2s) \rfloor \geqq 2b$, then $c \geqq (a-1)\lfloor \log_2 (2s) \rfloor + 2b \geqq ab$. We can assign disjoint sets of b colours to each of the a or fewer hyperedges incident with each vertex in R. A vertex in S is incident with at most b hyperedges, so the sets of colours assigned to these hyperedges will possess a system of distinct representatives.

Suppose, then, that $\lfloor \log_2 (2s) \rfloor \leqq 2b$. We shall prove that G has a c-hypercolouring in which $\lfloor \log_2 (2s) \rfloor$ colours are assigned to each hyperedge. The proof is by induction on the number of hyperedges in G. If G has no hyperedges, then we are done. If G has at least one edge, let G' be a hypergraph obtained by deleting one hyperedge H incident with a vertex ρ in R. By inductive hypothesis, G' has a c-hypercolouring in which $\lfloor \log_2 (2s) \rfloor$ colours are assigned to each hyperedge. It will suffice to show that this c-hypercolouring can be extended to a c-hypercolouring of G by assigning a set of $\lfloor \log_2 (2s) \rfloor$ colours to H.

Let us say that a colour is *good* at ρ if it does not belong to the union of the sets of colours assigned to the hyperedges incident with ρ in G'. All but at most $(a-1)\lfloor \log_2 (2s) \rfloor$ colours are good at ρ, so at least $2b$ colours are good at ρ. Let σ be a vertex in S. Let us say that a colour is *good* at σ if the sets of colours assigned to the hyperedges incident with σ possess a system of distinct representatives not containing that colour. For each σ in S, all but at most b colours are good at σ. Let K be a set of colours that are good at ρ. Assigning the set K to the hyperedge H will yield a c-hypercolouring of G if, for every σ in S, some colour in K is good at σ. There are at least $(2b) \cdots (2b-k+1)/k!$ ways of choosing $k \leqq 2b$ colours that are good at ρ. For each σ in S, at most $(b) \cdots (b-k+1)/k!$ of these do not contain a colour that is good at σ. Thus at most $s(b) \cdots (b-k+1)/k!$ of them do not yield a c-hypercolouring of G. If $k = \lfloor \log_2 (2s) \rfloor \leqq 2b$, then

$$s < 2^k \leqq (2b) \cdots (2b-k+1)/(b) \cdots (b-k+1),$$

and there exists a set K of k colours that yields a c-hypercolouring of G. □

Taking all subnetworks in Proposition 6 to be crossbars and choosing $a = s = \lceil (n/\log_2 n)^{1/2} \rceil$ and $b = r = \lceil (n \log_2 n)^{1/2} \rceil$ yields generalized n-connectors with depth 3 and size $O(n^{3/2}(\log n)^{1/2})$. More generally, we can prove by induction on k that $g_{2k-1}(n) = O(n^{1+1/k}(\log n)^{(k-1)/2})$. If $k = 1$, and n-crossbar provides the basis $g_1(n) = O(n^2)$. If $k \geqq 2$, we apply Proposition 6 with $a = \lceil n^{1/k}(\log n)^{(k-3)/2} \rceil$, $b = \lceil n^{1/k}(\log n)^{(k-1)/2} \rceil$, $r = \lceil n^{(k-1)/k}/(\log n)^{(k-3)/2} \rceil$ and $s = \lceil n^{(k-1)/k}/(\log n)^{(k-1)/2} \rceil$. We have $r \geqq s$, and $g_{2k-3}(r, s) \leqq \lceil r/s \rceil g_{2k-3}(s)$, as can be seen by identifying the outputs of $\lceil r/s \rceil$ disjoint generalized s-connectors. Thus, by inductive hypothesis, we have $g_{2k-3}(r, s) = O(rs^{1/(k-1)}(\log s)^{(k-2)/2})$. Taking the subnetworks in the first and third stages to be crossbars and constructing the subnetworks in the second stage recursively completes the induction. This is asymptotically the best construction known for generalized connectors with limited depth. It matches, to within logarithmic factors, the best explicit construction known for connectors.

Remark. It is possible to improve Theorem 7 as regards constant factors but we do not know whether it is, to within a constant factor, best possible. As we saw in the proof of the Theorem, ab colours are always sufficient. We shall show that $(a-1)b+1)$ colours may be necessary, if r and s are large enough.

Let $[x, y]$ denote the binomial coefficient $x(x-1) \cdots (x-y+1)/y!$. Given a and b, construct the hypergraph H as follows. There are $r = b[(a-1)b, b-1]$ vertices in R, each incident with a hyperedges, for a total of ar hyperedges. For each set of b hyperedges, there is a vertex in S incident with just those b hyperedges, for a total of

$s = [ar, b]$ vertices in S. Suppose that H can be hypercoloured with $c = (a-1)b$ colours. We shall derive a contradiction.

Because of the vertices in R, each of the c colours can be assigned to at most r hyperedges. It follows that at most $cr/b = (a-1)r$ hyperedges are assigned b or more colours, and thus that at least $ar-(a-1)r = r$ hyperedges are each assigned at most $b-1$ colours. There are $[c, b-1]$ ways of choosing $b-1$ colours; thus some set of $r/[c, b-1] = b$ hyperedges are assigned colours from some set of $b-1$ colours. These b hyperedges are incident with some vertex in S, but their colours cannot possess a system of distinct representatives. This contradiction shows that at least $(a-1)b+1$ colours are necessary. □

The foregoing example excludes the possibility that $c = O(\max\{a, b\})$ is sufficient (as is the case for edge-colouring bipartite graphs), but it does not exclude the possibility that $c = O(\max\{r, s, a, b\})$ is sufficient. If this were true, then taking all subnetworks to be crossbars and choosing $a = b = r = s = \lceil n^{1/2} \rceil$ would yield generalized n-connectors with depth 3 and size $O(n^{3/2})$. Taking the subnetworks in the first and third stages to be crossbars and constructing the subnetworks in the second stage recursively would yield generalized n-connectors with depth $2k-1$ and size $O(n^{1+1/k})$. This would match, to within a constant factor, the best explicit construction known for connectors.

REFERENCES

[1] L. A. BASSALYGO, *Asymptotically optimal switching circuits*, Prob. Info. Trans., 17 (1981), pp. 206–211.

[2] V. E. BENEŠ, *Mathematical Theory of Connecting Networks and Telephone Traffic*, Academic Press, New York, 1965.

[3] C. BERGE, *Graphs and Hypergraphs*, North-Holland, Amsterdam, 1973.

[4] D. DOLEV, C. DWORK, N. PIPPENGER AND A. WIGDERSON, *Superconcentrators, generalizers and generalized connectors with limited depth*, 15th ACM Symposium on Theory of Computing, 1983, pp. 42–51.

[5] O. GABBER AND Z. GALIL, *Explicit constructions of linear-sized superconcentrators*, J. Comput. Sys. Sci., 22 (1981), pp. 407–420.

[6] F. K. HWANG, *Control algorithms for rearrangeable close networks*, IEEE Trans. Comm., 31 (1983), pp. 952–954.

[7] G. LEV, N. PIPPENGER AND L. G. VALIANT, *A fast parallel algorithm for routing in permutation networks*, IEEE Trans. Comput., 30 (1981), pp. 93–100.

[8] G. M. MASSON AND B. W. JORDAN, JR., *Generalized multi-stage connection networks*, Networks, 2 (1972), pp. 191–209.

[9] D. NASSIMI AND S. SAHNI, *Parallel permutation and sorting algorithms and a new generalized connection network*, J. Assoc. Comput. Mach., 29 (1982), pp. 642–667.

[10] Yu. P. OFMAN, *A universal automaton*, Trans. Moscow Math. Soc., 14 (1965), pp. 200–215.

[11] N. PIPPENGER, *Generalized connectors*, SIAM J. Comput., 7 (1978), pp. 510–514.

[12] ———, *On rearrangeable and non-blocking switching networks*, J. Comput. System Sci., 17 (1978), pp. 145–162.

[13] ———, *A new lower bound for the number of switches in rearrangeable networks*, this Journal, 1 (1980), pp. 164–167.

[14] N. PIPPENGER AND A. C.-C. YAO, *Rearrangeable networks with limited depth*, this Journal, 3 (1982), pp. 411–417.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 4, October 1985

006

ON SUBCODES OF GENERALIZED SECOND ORDER REED-MULLER CODES*

H. J. TIERSMA†

Abstract. We study sets of quadratic forms in m variables over $GF(q)$ such that the difference of any two quadratic forms in the set has rank m. It is easily seen that the cardinality of such a set is at most q^m, and we shall give four constructions of such sets which have cardinality exactly q^m (where q is an odd prime power). The first construction works only in the case when m is odd, and uses a BCH code. The second is an ad hoc construction in the case when $m=2$. The third construction is due to H. A. Wilbrink, and generalizes the second (though not in an obvious way). The fourth uses symmetric representations of finite fields and is due to G. Serousi and A. Lempel. These sets correspond to subcodes of the generalized second order Reed-Muller code, and we give the weight distribution of these codes.

AMS(MOS) subject classifications. 94B05, 94B15

Introduction and history. Reed-Muller codes and especially first and second order Reed-Muller codes over $GF(2)$ have been studied extensively by Kasami, Lin, Peterson, Goethals and Delsarte (see for instance [5, Chaps. 13, 14 and 15]).

The weight distribution of the binary second order Reed-Muller code was obtained in the following way: First we notice that the second order Reed-Muller code is a union of cosets of the first order Reed-Muller code. Then we observe that these cosets are in 1-1 correspondence with symplectic forms, and that the weight distribution in a coset is uniquely determined by the rank of the corresponding bilinear form. Now by counting symplectic forms of given rank, the weight distribution of the second order Reed-Muller code can be obtained. In [7] McEliece obtained the weight distribution of a generalized second order Reed-Muller code over $GF(q)$ (where q is a prime power), by using almost the same method. The only difference is that he uses quadratic forms instead of symplectic forms.

The next step is the observation that the minimum weight of a coset increases if the rank of the corresponding form increases. In the binary case, one obtains subcodes of the second order Reed-Muller code which have a large minimum distance by constructing maximal sets of sympletic forms with the property that the rank of the difference of any two symplectic forms in the set is maximal. If m is an integer, and 2^m is the length of the (binary) Reed-Muller code, then it appears that for m odd these subcodes are linear (in fact extended BCH codes), but for m even these subcodes are the nonlinear Kerdock codes.

In the following sections we shall extend these results to generalized Reed-Muller codes. In § 1 we establish our notation. Then in § 2 we shall say something about the weight distribution. Section 3 contains the constructions of maximal sets of quadratic forms. Finally in § 4 we give the weight distributions of the corresponding codes and conclude with some remarks.

1. Generalized Reed-Muller codes. In this section we establish our notation and give some results on generalized Reed-Muller codes. For the proofs of the theorems we refer to [1]. Let q be an odd prime power and let x_i be a variable taking values in $GF(q)$, $1 \leqq i \leqq m$, $(m \in \mathbb{N})$. Let $\mathbf{x} := (x_1, \cdots, x_m)$. P_ν denotes the set of polynomials of total degree less than or equal to ν, in the variables $x_1, \cdots, x_m$, with coefficients from $GF(q)$. V_j denotes a j-dimensional vector space over $GF(q)$.

* Received by the editors March 1, 1983, and in revised form May 15, 1984.
† Schoemekerstraat 23, 5688D Oirschot, The Netherlands.

We consider $GF(q^m)$ as an m-dimensional vector space over $GF(q)$. If α is a primitive element of $GF(q^m)$ then $1, \alpha, \cdots, \alpha^{m-1}$ is a basis for $GF(q^m)$, and we can write:

$$\alpha^j = \sum_{i=1}^{m} v_{j,i}\alpha^{i-1} \quad \text{where } v_{j,i} \in GF(q), \quad 0 \leqq j \leqq q^m - 2. \tag{1}$$

With $f \in P_\nu$ there corresponds a vector $\mathbf{v}(f)$, having q^m components; The ith component of $\mathbf{v}(f)$ is:

$$v_i(f) := f(v_{i-2,1}, \cdots, v_{i-2,m}), \qquad i = 2, \cdots, q^m. \tag{2}$$

The initial component is: $v_1(f) = f(0, \cdots, 0)$. So $\mathbf{v}(f)$ can be viewed as a table giving the values of f in the points of $GF(q^m)$ in some fixed order.

DEFINITION 1.1. Let $0 \leqq \nu \leqq (q-1)m$. The q-ary νth order generalized Reed-Muller code $GRM_\nu(q, m)$ consists of the vectors: $\{\mathbf{v}(f) | f \in P_\nu\}$.

DEFINITION 1.2.

$$k_\nu(q, m) := \left| \left\{ (s_1, \cdots, s_m) \middle| 0 \leqq s_i \leqq q-1, \sum_{i=1}^{m} s_i \leqq \nu \right\} \right|.$$

THEOREM 1.3. *$GRM_\nu(q, m)$ is a linear code of dimension $k_\nu(q, m)$ over $GF(q)$.*

DEFINITION 1.4. The code $GRM_\nu^*(q, m)$ consists of the words of $GRM_\nu(q, m)$ from which the first coordinate is deleted.

Remark 1.5. Each vector of $GRM_\nu(q, m)$ satisfies an overall parity check, so the words of $GRM_\nu(q, m)$ are in 1-1 correspondence to those of $GRM_\nu^*(q, m)$.

DEFINITION 1.6. For h in the range $0 \leqq h < q^m$ we define $w(h)$ by: $w(h) := \sum_{i=0}^{m-1} \delta_i$, where $h = \sum_{i=0}^{m-1} \delta_i q^i$, $0 \leqq \delta_i \leqq q-1$.

THEOREM 1.7. *The code $GRM_\nu^*(q, m)$ is a cyclic code having zeros: $\{\alpha^h | 0 < w(h) < m(q-1) - \nu\}$.*

2. Weight distributions. In this section we shall give an outline of a method for finding the weight distribution of the generalized first and second order Reed-Muller code. The detailed calculations are omitted, but can be found in [7].

From the definition of $GRM_\nu(q, m)$ it is easily seen that $GRM_\nu(q, m) \subset GRM_{\nu+1}(q, m)$. Therefore $GRM_2(q, m)$ is a union of cosets of $GRM_1(q, m)$. Obviously with each quadratic form $Q(\mathbf{x})$ (i.e. $Q(\mathbf{x}) = \sum_{i,j} x_i Q_{i,j} x_j$ where Q is a symmetric matrix over $GF(q)$), corresponds the following coset of $GRM_1(q, m)$ in $GRM_2(q, m)$: $\{\mathbf{v}(Q(\mathbf{x}) + \sum_{i=1}^{m} a_i x_i + \varepsilon) | \varepsilon \in GF(q), a_i \in GF(q), 1 \leqq i \leqq m\}$. Conversely with each coset of $GRM_1(q, m)$ in $GRM_2(q, m)$ corresponds a unique quadratic form.

THEOREM 2.1. *Every quadratic form of rank r in V_m can be transformed into precisely one of the following standard forms by a linear nonsingular transformation of the variables:*

If r is odd:

type 1: $Q(\mathbf{x}) = x_1x_2 + \cdots + x_{r-2}x_{r-1} + x_r^2$,
type 2: $Q(\mathbf{x}) = x_1x_2 + \cdots + x_{r-2}x_{r-1} + \gamma x_r^2$;

If r is even:

type 3: $Q(\mathbf{x}) = x_1x_2 + \cdots + x_{r-1}x_r$,
type 4: $Q(\mathbf{x}) = x_1x_2 + \cdots + x_{r-3}x_{r-2} + x_{r-1}^2 - \gamma x_r^2$;

where γ is a fixed nonsquare element of $GF(q)$.

For the proof of this theorem we refer to [6] or to [7].

DEFINITION 2.2. We shall call a quadratic form to be of rank r and type i, if it is equivalent to the standard form of rank r and type i. A coset of $GRM_1(q, m)$ in $GRM_2(q, m)$ is said to be of rank r and type i, if the corresponding quadratic form is of rank r and type i.

THEOREM 2.3. *The weight distribution of a coset of* $GRM_1(q, m)$ *in* $GRM_2(q, m)$ *depends only on its rank and type.*

For a proof of this theorem we refer to [7].

Using [7, Tables 4, 7 and 8], the weight distribution of a coset of rank r and type i can be obtained. The complete weight distribution of the second order generalized Reed-Muller code is given in [7, Table 10].

3. Maximal m-sets and subcodes. If we look at the weight distribution of cosets, we see that the minimum weight in a coset increases if the rank of the coset increases (Cf. [7]). We shall now try to find sets of quadratic forms on V_m satisfying:

(a) $0 \in S$,

(b) for all $Q_1, Q_2 \in S$: $r(Q_1 - Q_2) = m$.

These sets correspond to subcodes of $GRM_2(q, m)$ (taking the union of the cosets corresponding to the forms of S). By (a) the corresponding subcode contains $GRM_1(q, m)$. By (b) the corresponding subcode has a high minimum distance (and minimum weight). We shall try to make the sets S as large as possible. We have the following theorem.

THEOREM 3.1. *If* S *satisfies* (a) *and* (b), *then* $|S| \leqq q^m$.

Proof. Look at the first row of the matrices corresponding to the quadratic forms in S, and observe that every m-tuple over $GF(q)$ can appear at most once. □

DEFINITION 3.2. If S is a set of quadratic forms satisfying (a), (b) and the bound of Theorem 3.1 with equality (i.e. $|S| = q^m$), then S is called a maximal m-set.

In the following we will construct subcodes of $GRM_2(q, m)$ which correspond to maximal m-sets. Also we will construct maximal m-sets.

Construction 1. In this construction we shall use the theory of cyclic codes, which can be found in [5, Chaps. 7, 8].

We shall use for instance cyclotomic cosets C_i (cf. [5, p. 197]) and minimal idempotents θ_i^* (cf. [5, p. 221]).

We also require the following lemmas and remarks.

LEMMA 3.3. *Let* $a \in \mathbb{N}$, $a \leqq m(q-1)$. *The minimal positive integer satisfying* $w(h) \geqq a$ *is*: $h = (\mu+1)q^\rho - 1$, *where* μ, ρ *are such that* $a = \rho(q-1) + \mu$ $(0 \leqq \mu < q-1)$.

Proof. Clearly $h = (q-1) + (q-1)q + \cdots + (q-1)q^{\rho-1} + \mu q^\rho$. □

Let $S(\mathbf{x})$ be a quadratic function (corresponding to a codeword $\mathbf{v}$ of $GRM_2(q, m)$). We can consider the coordinates of $\mathbf{v}$ to be the values $S(\xi)$ for $\xi \in GF(q^m)$. S corresponds to a quadratic form and therefore to a symmetric bilinear form:

$$B(\xi, \eta) = S(\xi + \eta) - S(\xi) - S(\eta) + S(0).$$

We now observe that $S(\xi)$ is equal to $1/n$ times the Mattson-Solomon polynomial corresponding to $\mathbf{v}^*$. (This follows from [2, p. 417] since

$$S(\xi) = S(T_m(\lambda_1 \xi), \cdots, T_m(\lambda_m \xi)) = \frac{1}{n} F(\xi),$$

where T_m is the trace function (cf. [5, Ch. 4]), F is the Mattson-Solomon polynomial corresponding to $\mathbf{v}^*$, and $\lambda_1, \cdots, \lambda_m$ is the trace dual basis of $1, \alpha, \cdots, \alpha^{m-1}$ where α is a primitive element of $GF(q^m)$.)

The rank of the bilinear form is equal to the rank of the quadratic form.

LEMMA 3.4. *A bilinear form can be written by means of the trace-function as*: $B(\xi, \eta) = T_m(\eta L_B(\xi))$, *where* L_B *is an endomorphism of* V_m. *Then*: $\text{rank}(B) = m - \dim(\text{Ker}(L_B))$.

For a proof of this lemma we refer to [3].

Now we are ready to give the first construction. Let $\mathscr{C}$ be the *BCH*-code over $GF(q)$, with designed distance $\delta = q^m - q^{m-1} - q^{(m-1)/2} - 1$ and generator polynomial $g(x) = \text{l.c.m.}(m_1(x), \cdots, m_{\delta-1}(x))$, where m_j is the minimal polynomial of α^j, $1 \leqq j \leqq \delta - 1$.

Let $\hat{\mathscr{C}}$ be the extended code ($\mathscr{C}$ extended with an overall parity check).

THEOREM 3.5.

(1) $GRM_1^*(q, m) \subset \mathscr{C} \subset GRM_2^*(q, m)$, $GRM_1(q, m) \subset \hat{\mathscr{C}} \subset GRM_2(q, m)$.

(2) *The dimension of* $\mathscr{C}$ *is* $2m+1$.

(3) $\mathscr{C}$ *has nonzeros*: $\{\alpha^h | h \in C_0 \cup C_{-1} \cup C_{-(q^{\frac{1}{2}}(m-1)_{+1})}\}$.

(4) *The* q^m *quadratic forms which correspond to the* q^m *different cosets of* $GRM_1(q, m)$ *in* $\hat{\mathscr{C}}$, *form a maximal m-set.*

Proof.

(1) Let $h \in \{1, 2, \cdots, q^m - q^{m-1} - q^{(m-1)/2} - 2\}$ (i.e. α^h is a zero of $\mathscr{C}$). Then by Lemma 3.3: $0 < w(h) < m(q-1) - 1$ (since otherwise $h \geqq (q-1)q^{m-1} - 1 > (q-1)q^{m-1} - q^{(m-1)/2} - 2$). According to Theorem 1.7 we now have that every zero of $\mathscr{C}$ is a zero of $GRM_1^*(q, m)$ and thus that $GRM_1^*(q, m) \subset \mathscr{C}$. Let h be such that $0 < w(h) < m(q-1) - 2$, and let $(\delta_0(h), \cdots, \delta_{m-1}(h))$ be the q-ary representation of h. Then with some cyclic shift of this q-ary representation there corresponds an h^* for which $h^* < (q-2)q^{m-1} - 1$ (from Lemma 3.3). Since h^* is in the same cyclotomic coset as h, and $h^* < q^m - q^{m-1} - q^{m-1} - 1 < q^m - q^{m-1} - q^{(m-1)/2} - 2$, α^h is a zero of $\mathscr{C}$, and therefore we have that every zero of $GRM_2^*(q, m)$ is a zero of $\mathscr{C}$ and thus that $\mathscr{C} \subset GRM_2^*(q, m)$. From Remark 1.5, it now follows that $GRM_1(q, m) \subset \hat{\mathscr{C}} \subset GRM_2(q, m)$.

(2), (3) The q-ary representation of $\delta - 1$ is:

$$[d_{m-1}, \cdots, d_0] := (q-2, \underbrace{q-1, \cdots, q-1}_{\frac{1}{2}(m-3)}, q-2, \underbrace{q-1, \cdots, q-1}_{\frac{1}{2}(m-3)}, q-2).$$

The nonzeros of $\mathscr{C}$ are the α^h such that h has a q-ary representation $[\delta_{m-1}(h), \cdots, \delta_0(h)]$ for which every cyclic shift is lexicographically greater than $[d_{m-1}, \cdots, d_0]$. So h has a q-ary representation which is a cyclic shift of one of the following sequences:

$(q-1, \cdots, q-1)$, giving one possibility for h: $h \in C_0$;

$(q-1, \cdots, q-1, q-2)$, giving m possibilities for h: $h \in C_{-1}$;

$$(\underbrace{q-1, \cdots, q-1}_{\frac{1}{2}(m-1)}, q-2, \underbrace{q-1, \cdots, q-1}_{\frac{1}{2}(m-3)}, q-2),$$

giving m possibilities for h: $h \in C_{-(q^{(m-1)/2}+1)}$.

(4) From (3) the idempotent of the code $\mathscr{C}$ is $\theta_0 + \theta_1^* + \theta_k^*$, where $k = q^{(m-1)/2} + 1$ (cf. [5, p. 221, 222]), so the code consists of the codewords $c\theta_0 + a(x)\theta_1^* + b(x)\theta_k^*$. The Mattson–Solomon polynomial of such a word is (cf. [5, p.248]): $c + T_m(\beta_1 z) + T_m(\beta_2 z^k)$, (where β_1 and β_2 are elements of $GF(q^m)$ depending on $a(x)$ and $b(x)$).

From the remark preceding Lemma 3.4 the corresponding bilinear form is equal to:

$$B(\xi,\eta)=\frac{1}{n}\{(c+T_m(\beta_1[\xi+\eta])+T_m(\beta_2[\xi+\eta]^k))$$
$$-(c+T_m(\beta_1\xi)+T_m(\beta_2\xi^k))$$
$$-(c+T_m(\beta_1\eta)+T_m(\beta_2\eta^k))+(c+T_m(0)+T_m(0))\}$$
$$=\frac{1}{n}(T_m(\beta_2[(\xi+\eta)^k-\xi^k-\eta^k])),$$

where $\beta_2\in GF(q)^*$ if $b(x)\neq 0$.

Remembering that $k=q^{(m-1)/2}+1$, we find (after some calculations) that

$$B(\xi,\eta)=\frac{1}{n}T_m(\xi(\beta_2\eta^{q^{(m-1)/2}}+\beta_2^{q^{(m+1)/2}}\eta^{q^{(m+1)/2}}))=\frac{1}{n}T_m(\xi L_B(\eta)),$$

where

$$L_B(\eta)=\beta_2\eta^{q^{(m-1)/2}}+\beta_2^{q^{(m+1)/2}}\eta^{q^{(m+1)/2}}.$$

We now have $B(\xi,\eta)$ in the form of Lemma 3.4, and the rank of B is equal to $m-\dim(\mathrm{Ker}(L_B))$. But $\mathrm{Ker}(L_B)=\{0\}$, since: $L_B(\eta)=0$ iff

$$(\beta_2^{q^{(m+1)/2}}\eta+\beta_2^q\eta^q)^{q^{(m-1)/2}}=0,$$

i.e. iff

$$\beta_2\eta(\beta_2^{q^{(m+1)/2-1}}+\beta_2^{q-1}\eta^{q-1})=0.$$

Suppose η is a solution of the last equation. If $\eta\neq 0$, then $\eta=\alpha^u$ for some $u\in\mathbb{Z}$, since α is a primitive element of the field $GF(q^m)$. Also $\beta_2=\alpha^g$ for some $g\in\mathbb{Z}$, for the same reason.

Since $\eta\neq 0$, it follows (after substitution of $\eta(=\alpha^u)$ and $\beta_2(=\alpha^g)$) that $\alpha^{u(q-1)}=-\alpha^{g(q^{(m+1)/2}-q)}$. Therefore we have the congruence $u(q-1)\equiv g(q^{(m+1)/2}-q)+\frac{1}{2}(q^m-1) \bmod (q^m-1)$.

Let $u'\in\mathbb{Z}$ be such that

$$u=u'+g\frac{q^{(m+1)/2}-q}{q-1}.$$

Then this u' must satisfy the congruence $(q-1)u'\equiv\frac{1}{2}(q^m-1) \bmod (q^m-1)$. Therefore we have that $(q-1)u'=\frac{1}{2}(q^m-1)+j(q^m-1)$ for some $j\in\mathbb{Z}$. This yields the equation:

$$2u'-2j\frac{q^m-1}{q-1}=\frac{q^m-1}{q-1}.$$

The left-hand side of this equation is obviously an even number, while the right-hand side of this equation is odd (since m is odd). This yields a contradiction.

Conclusion. $\mathrm{Ker}(L_B)=\{0\}$, and so: the rank of the form is m. □

Construction 2. For $m=2$ we define the following maximal m-set:

$$S_\gamma:=\left\{\begin{pmatrix}x_1 & x_2\\ x_2 & \gamma x_1\end{pmatrix}\middle|(x_1,x_2)\in GF(q)^2\right\},$$

where γ is a nonsquare element of $GF(q)$.

THEOREM 3.6. *S_γ is a maximal 2-set.*

Proof. S_γ forms an additive group. $|S_\gamma|=q^2$. If $(x_1,x_2)\neq(0,0)$ then

$$\det\left[\begin{pmatrix}x_1 & x_2\\ x_2 & \gamma x_1\end{pmatrix}\right]=\gamma x_1^2-x_2^2\neq 0.$$

THEOREM 3.7.

$$\#(\textit{type 3 matrices in } S_\gamma) = \#(\textit{type 4 matrices in } S_\gamma) = \tfrac{1}{2}(q^2-1).$$

Proof. The determinant of a type 3 matrix is $-r^2$, for some $r \in GF(q)^*$. The determinant of a type 4 matrix is $-\gamma r^2$, for some $r \in GF(q)^*$ (where γ is a nonsquare).

If (x_1, x_2) runs through $GF(q)^2 \backslash \{(0,0)\}$, $\gamma x_1^2 - x_2^2$ takes equally often as values a square and a nonsquare (see [6]). □

Construction 3. (Wilbrink's construction.) We shall now give a general construction method for maximal m-sets, which is due to H. A. Wilbrink.

Let $a(x)$ be an irreducible polynomial of degree m,

$$a(x) = a_0 + a_1 x + \cdots + a_m x^m, \qquad a_m = 1.$$

Now: $GF(q)[x]/(a(x)) \cong GF(q^m)$, say with isomorphism ψ. Let C be an $m \times m$ matrix which has $a(x)$ as minimal polynomial. (For instance we can take for C the companion matrix of $a(x)$.)

LEMMA 3.8. *If $f(x) \in GF(q)[x]/(a(x))$, $f \neq 0$, then $f(C)$ is a nonsingular matrix.*

Proof. Let $f(x) \in GF(q)[x]/(a(x))$, $f \neq 0$. Then $\psi(f(x)) \neq 0$ and therefore $\psi(f(x))$ has an inverse. Define $g(x)$ to be: $g(x) := \psi^{-1}(\psi(f(x))^{-1})$. Then: $g(x)f(x) = f(x)g(x) = \psi^{-1}(\psi(f(x))\psi(f(x))^{-1}) = \psi^{-1}(1) = 1$. So $f(C)g(C) = I = g(C)f(C)$ and $f(C)$ is nonsingular. □

Now it is possible to find matrices M such that:

(A) $M^T = M$.

(B) $MC = C^T M$.

Since these two equations are in fact $\binom{m}{2} + \binom{m}{2} = m^2 - m$ linear homogeneous equations in the m^2 variables $(M)_{i,j}$, in fact there are at least q^m solutions for M.

THEOREM 3.9. *If M is a solution of* (A) *and* (B) *then $M = 0$ or M is nonsingular.*

Proof. Suppose that M is singular and let $\mathbf{x} \in N(M)$, $\mathbf{x} \neq 0$, where $N(M) = \{\mathbf{x} | M\mathbf{x} = 0\}$. Since $MC = C^T M$ it follows that $MC^i = C^{iT} M$ (by induction), and therefore $Mf(C) = f(C)^T M$ for every polynomial f. Now $Mf(C)\mathbf{x} = f(C)^T M\mathbf{x} = 0$; So $f(C)\mathbf{x} \in N(M)$ for every $f(x) \in GF(q)[x]/(a(x))$. We conclude that $\dim(N(M)) = m$ and therefore that $M = 0$. □

Let M_0 be a solution of (A) and (B), $M_0 \neq 0$.

DEFINITION 3.10.

$$S_a := \{M_0 f(C) | f(x) \in GF(q)[x]/(a(x))\}.$$

THEOREM 3.11. *S_a is a maximal m-set.*

Proof. $0 \in S_a$. The matrices $M_0 f(C)$ are symmetric:

$$(M_0 f(C))^T = f(C)^T M_0^T = f(C)^T M_0 = M_0 f(C).$$

The matrices of S_a are nonsingular: M_0 is nonsingular, $f(C)$ is nonsingular for every $f(x) \in GF(q)[x]/(a(x))$ unequal to 0, so $M_0 f(C)$ is nonsingular.

The difference of two matrices of S_a is an element of S_a, and therefore nonsingular. $|S_a| = q^m$, so S_a is maximal. □

LEMMA 3.12. *S_a is the set of solutions of equations* (A) *and* (B).

Proof. The matrices $M_0 f(C)$ are symmetric: $M_0 f(C) = (M_0 f(C))^T$. So equation (A) is satisfied.

$$(M_0 f(C))C = M_0(f(C)C) = (f(C)C)^T M_0 = C^T(f(C)^T M_0) = C^T(M_0 F(C)).$$

So equation (B) is satisfied.

Since the solutions are nonsingular or 0 and form an m-set the number of solutions is less than or equal to q^m, so the number of solutions is $q^m = |S_a|$. □

COROLLARY 3.13. *If M_1 and M_2 are both solutions of the equations* (A) *and* (B), *then there is a polynomial f_1 such that $M_2 = M_1 f_1(C)$.*

LEMMA 3.14. *There exists a polynomial $g(x) \in GF(q)[x]/(a(x))$, such that* $\det(g(C))$ *is a nonsquare element of $GF(q)$.*

Proof. The set of matrices $\{f(C) | f \in GF(q)[x]/(a(x))\}$ is a field, which is isomorphic to $GF(q^m)$, with isomorphism: $\varphi(f(C)) = \psi(f)$. (Where ψ is defined as before.)

Let ξ be a primitive element of the field $GF(q^m)$. Let $b(x)$ be the minimal polynomial of ξ i.e.:

$$b(x) = (x-\xi)(x-\xi^q)\cdots(x-\xi^{q^{m-1}}) = b_0 + b_1 x + \cdots + b_m x^m.$$

Let $g(x) := \psi^{-1}(\xi)$, and define D to be: $D := g(C)$. Then $b(x)$ is the minimal polynomial of D, since:

1) $0 = \varphi^{-1}(0) = \varphi^{-1}(b(\xi)) = b(\varphi^{-1}(\xi)) = b(D)$.

2) If $c(x)$ is a polynomial such that $c(D) = 0$, then

$$0 = \varphi(0) = \varphi(c(D)) = c(\varphi(D)) = c(\varphi(g(C))) = c(\psi(g)) = c(\xi),$$

and so we have that $b(x) | c(x)$. Now

$$\det(D) = b_0 = \xi\xi^q \cdots \xi^{q^{m-1}} = \xi^{(q^m-1)/(q-1)}.$$

Since ξ is a primitive element of $GF(q^m)$, $\xi^{(q^m-1)/(q-1)}$ is a primitive element of $GF(q)$, and therefore is a nonsquare element of $GF(q)$. □

THEOREM 3.15. *Let M_i denote the set of type i solutions of the equations* (A) *and* (B), *$i = 1, 2, 3, 4$. If m is even then $M_1 = M_2 = \varnothing$, $|M_3| = |M_4|$. If m is odd then $M_3 = M_4 = \varnothing$, $|M_1| = |M_2|$.*

Proof. Suppose that m is even. Then obviously there are no type 1, 2 solutions. So the set $\mathcal{M}$ of solutions can be written as $\{0\} \cup M_3 \cup M_4$.

Let T_1 and T_2 be two solutions of the equations (A) and (B). Then

$$\det(T_1)\det(T_2) = \begin{cases} \text{nonsquare} & \text{if type } (T_1) \neq \text{type } (T_2), \\ \text{square} & \text{if type } (T_1) = \text{type } (T_2). \end{cases}$$

Let $g(x) \in GF(q)[x]/(a(x))$ be such that $\det(g(C))$ is a nonsquare. (Such a $g(x)$ exists, see Lemma 3.19.) Consider $M_3' = \{Tg(C) | T \in M_3\}$, $M_4' = \{Tg(C) | T \in M_4\}$. Since $\det(T)\det(Tg(C)) = \det(T)^2 \det(g(C))$ is a nonsquare, we have that: $M_3' \subset M_4$, and $M_4' \subset M_3$.

Therefore we have that $|M_3| = |M_3'| \leqq |M_4| = |M_4'| \leqq |M_3|$, from which we conclude that $|M_3| = |M_4|$. If m is odd one can show in an analogous way that $M_3 = M_4 = \varnothing$, $|M_1| = |M_2|$.

Construction 4. Another general construction method for maximal m-sets is due to G. Seroussi and A. Lempel.

THEOREM 3.16. *Every finite extension $GF(q^m)$ of a finite field $GF(q)$ ($m \in \mathbb{N}$), has a symmetric representation (i.e., there exists a set of symmetric matrices of order m, over $GF(q)$ which is isomorphic to the field $GF(q^m)$).*

The proof of this theorem can be found in [4]. Now such a set forms a maximal m-set.

4. Parameters and weight distribution of the corresponding codes, some remarks. Using the results about the weight distribution of cosets of rank r and type i (cf. [7]), one can calculate the weight distribution of the codes corresponding to the maximal m-sets.

The extended *BCH*-code $\hat{\mathscr{C}}$ of construction 1 has parameters $[q^m, 2m+1, q^m - q^{m-1} - q^{(m-1)/2}]$, and weight distribution:

wt	0	$q^m - q^{m-1} - q^{(m-1)/2}$	$q^m - q^{m-1}$	$q^m - q^{m-1} + q^{(m-1)/2}$	q^m
#	1	$\frac{1}{2}(q-1)(q^m-1)q^m$	$(q^m-1)(q^m+q)$	$\frac{1}{2}(q-1)(q^m-1)q^m$	$q-1$

The codes corresponding with the sets S_γ of Construction 2 have parameters: $[q^2, 5, q^2 - 2q+1]$; and weight distribution:

wt	0	q^2-2q+1	q^2-q-1	q^2-q	q^2-q+1	q^2-1	q^2
#	1	$\frac{1}{2}(q^2-1)q^2$	$\frac{1}{2}(q-1)q^2(q^2-1)$	q^3-q	$\frac{1}{2}q^2(q-1)(q^2-1)$	$\frac{1}{2}(q^2-1)$	$q-1$

The codes corresponding with S_a of Construction 3 and the codes corresponding to the symmetric representation of $GF(q^m)$ over $GF(q)$ have the same parameters and weight distribution.

These codes have:

1) the same parameters and weight distribution as the extended *BCH*-codes of Construction 1, when m is odd.

2) parameters: $[q^m, 2m+1, q^m - q^{m-1} - (q-1)q^{m/2-1}]$; and weight distribution:

wt	0	$q^m - q^{m-1} - (q-1)q^{m/2-1}$	$q^m - q^{m-1} - q^{m/2-1}$	$q^m - q^{m-1}$	$q^m - q^{m-1} + q^{m/2-1}$
#	1	$\frac{1}{2}(q^m-1)q^m$	$\frac{1}{2}(q^m-1)(q-1)q^m$	$q^{m+1}-q$	$\frac{1}{2}(q^m-1)(q-1)q^m$

wt	$q^m - q^{m-1} + (q-1)q^{m/2-1}$	q^m
#	$\frac{1}{2}(q^m-1)q^m$	$q-1$

when m is even.

Remarks.

1. The codes of Construction 3 when m is odd are in general not equivalent to the extended *BCH*-codes of construction 1. If we take for example $q=3$ and $m=3$ and we look at the maximal 3-set which corresponds to the extended *BCH*-code $\hat{\mathscr{C}}$, then we find:

$$S = \left\{ \begin{pmatrix} x_1 - x_2 & x_1 - x_2 + x_3 & -x_1 \\ x_1 - x_2 + x_3 & x_1 & x_2 \\ -x_1 & x_2 & x_3 \end{pmatrix} \middle| (x_1, x_2, x_3) \in GF(3)^3 \right\}.$$

The only matrices C such that for all $M \in S$: $MC = C^T M$, are of the form cI, where I is the identity matrix. Since these matrices do not have an irreducible minimal polynomial, the set S can not come from Construction 3 and therefore the codes are inequivalent. From this example it also can be seen that the codes of Construction 4 are in general not equivalent to those of Construction 1, since S is not closed w.r.t. matrix multiplication.

2. Take $m=2$, q an odd prime power and $\gamma \in GF(q)$, a nonsquare. Define $f(x) := x^2 - 2x + 1 - \gamma^{-1} = (x-1)^2 - \gamma^{-1}$. Then f is irreducible (γ^{-1} is a nonsquare).

Take

$$C = \begin{pmatrix} 1 & 1 \\ \gamma^{-1} & 1 \end{pmatrix}.$$

Then f is the minimal polynomial of C.

Take

$$M = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}.$$

Then MC is a symmetric matrix. Now from Construction 3: $\{M(aC+bI) | (a, b) \in GF(q)^2\}$ is a maximal 2-set. Since

$$M(aC+bI) = \begin{pmatrix} \gamma^{-1}a & a+b \\ a+b & a \end{pmatrix},$$

the above maximal 2-set is equal to

$$\left\{ \begin{pmatrix} x_1 & x_2 \\ x_2 & \gamma x_1 \end{pmatrix} \middle| (x_1, x_2) \in GF(q)^2 \right\} = S_\gamma,$$

if we take $a = \gamma x_1$, $b = -\gamma x_1 + x_2$.

From this we conclude that Construction 3 is a generalization of Construction 2.

REFERENCES

[1] T. KASAMI, S. LIN AND W. W. PETERSON, *Generalized Reed-Muller codes*, Electronics and Communications in Japan, 51 (1968), pp. 96-103.

[2] P. DELSARTE, J.-M. GOETHALS AND F. J. MACWILLIAMS, *On generalized Reed-Muller codes and their relatives*, Inform. and Control, 16 (1970), pp. 403-442.

[3] P. DELSARTE AND J.-M. GOETHALS, *Alternating bilinear forms over GF(q)*, J. Combin. Theory, 19A (1975), pp. 26-50.

[4] G. SEROUSSI AND A. LEMPEL, *On symmetric representations of finite fields*, this Journal, 4 (1983), pp. 14-20.

[5] F. J. MACWILLIAMS AND N. J. A. SLOANE, *The Theory of Error Correcting Codes*, North-Holland, Amsterdam, 1977.

[6] L. E. DICKSON, *Linear Groups with an Exposition of the Galois Field Theory*, Dover, New York, 1958.

[7] R. J. MCELIECE, *Quadratic forms over finite fields and second-order Reed-Muller codes*, JPL Space Programs Summary 37-58 III (1969), pp. 28-33.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 4, October 1985

VOLTERRA MULTIPLIERS I*

RAY REDHEFFER†

Abstract. If p is a real square matrix, a *Volterra multiplier* is a positive diagonal matrix a such that, in the sense of quadratic forms, $ap \leqq 0$. Ever since the pioneering work of Volterra over half a century ago, it has been known that these multipliers are a significant aid in the study of stability. However, the utility of the method is diminished by the difficulty of deciding whether the multiplier exists. Here we give a number of new criteria for existence, usually under the hypothesis that p is combinatorially symmetric; that is, $p_{ij} = 0$ implies $p_{ji} = 0$. This is much weaker than the condition "$p_{ij}p_{ji} < 0$ unless $p_{ij} = p_{ji} = 0$" which has been used by Volterra and others, and greatly increases the scope of the results. Although the primary emphasis is on sufficient conditions that are easy to use, we give necessary and sufficient conditions for several cases of practical interest.

AMS(MOS) subject classifications. 15A45, 15A48

1. Introduction. An important class of interactive systems can be described by differential equations of the form

$$\dot{u}_i = u_i\left(e_i + \sum_{j=1}^{m} p_{ij}u_j\right), \qquad i = 1, 2, \cdots, m \tag{1}$$

where $u_i(0) > 0$ and all variables are real. This system was introduced by Lotka and Volterra around 1930 and has been extensively studied ever since. Together with its analogues and extensions it occurs in such diverse problems as pest control, the management of fish populations, the propagation of genetic traits, the spread of epidemics, and the kinetics of chemical reactions. In most applications one wants to know the long-term behavior: Does each u_i eventually settle down to a limiting value? If so, is the value independent of the initial conditions? If not, is there a periodic or an almost periodic solution? Do any of the populations (if it is a question of ecology) ever become extinct?

Conditions on the matrix $p = (p_{ij})$ necessary and sufficient to guarantee any one of the above behaviors are not known. However, a significant and useful hypothesis, due to Volterra, is that there exist positive multipliers a_i such that, in the sense of quadratic forms, $(a_ip_{ij}) \leqq 0$. Volterra's hypothesis is verified in many cases of practical interest and it leads to a great deal of added insight. The usefulness of this hypothesis is not confined to the system (1) but extends to systems of the form

$$\dot{u}_i = N(u)f_i(u_i)\left(e_i + \sum_{j=1}^{m} p_{ij}g_j(u_j)\right) \tag{2}$$

and to other generalizations of (1), some of which involve time delay or diffusion. A sample of such investigations can be found in [12], [13], [15], [20], [33], [34], [35], [36], [37], [39], [41].

The problem of obtaining usable criteria for the existence of Volterra's multipliers a_i has been open ever since the hypothesis $(a_ip_{ij}) \leqq 0$ was introduced by Volterra about a half century ago. Not until 1978 were satisfactory necessary and sufficient conditions known even for the case $m = 3$, and the discovery of such conditions by Cross [7] marks what is perhaps the most significant advance until that time. The result of Cross

* Received by the editors May 24, 1983, and in revised form May 15, 1984.
† Department of Mathematics, University of California, Los Angeles, California 90024.

is used effectively in [41] but the lack of simple sufficient conditions in the general case remains an obstacle to further progress.

Such conditions are obtained in [38] but only under the supplementary hypothesis that $p_{ij}p_{ji} < 0$ whenever $p_{ij}(i-j) \neq 0$. This inequality agrees with the Volterra hypothesis for the prey-predator case in (1) but is not needed for the study of (2) and its generalizations as outlined above; the differential equations aspect of the theory goes through equally well with no hypothesis on the sign of $p_{ij}p_{ji}$, provided the multipliers a_i are known to exist. For this reason it seems desirable to extend the theory of [38], [39] by dropping the condition $p_{ij}p_{ji} < 0$ and such is a principal goal of Part I. The main result [38] is summarized in § 18 and is generalized in §§ 19-22. An extension of the theorem of Cross to matrices of arbitrary order is given in Part II (this issue, pp. 590-598) together with a number of illustrative examples.

To avoid unnecessary clutter in the statement of our theorems we agree once and for all that p is a real m by m matrix with $m \geqq 2$. Since m is regarded as fixed, we do not carry m as a separate parameter.

2. The classes A_0 and A_1. The following definition is important in the sequel:

DEFINITION 1. It is said that $p \in A_0$ or $p \in A_1$ if there exists a positive diagonal matrix a such that, in the sense of quadratic forms, $ap \leqq 0$ or $ap < 0$, respectively.

When our results pertain to both classes A_i we generally formulate the proofs for one class only, leaving the other case to the reader. At first glance the condition $ap \leqq 0$ associated with A_0 seems more restrictive than $bpc \leqq 0$ where b and c, like a, are positive diagonal matrices. However the identity

$$x^t c^{-1} bpx = y^t bpcy, \qquad x = cy,$$

shows that in fact $bpc \leqq 0$ implies $ap \leqq 0$ with $a = c^{-1}b$ and hence it implies $p \in A_0$. Taking $b = I$, the identity matrix, we see that the definition of the class A_0 could be based on the inequality $pa \leqq 0$ rather than $ap \leqq 0$. This in turn shows that $p \in A_0$ if, and only if, $p^t \in A_0$. Similar remarks apply to A_1. These well-known observations are used below.

3. The classes P_0, P_1 and P. Some further classes of matrices with which we shall be concerned are defined as follows:

DEFINITION 2. It is said that $p \in P_0$ if all the principal minors are nonnegative, $p \in P$ if all the principal minors are positive, and $p \in P_1$ if the leading principal minors are positive.

The distinction between the classes P and P_1 is important in applications, because to test for $p \in P$ one would have to examine

$$\binom{m}{1} + \binom{m}{2} + \cdots + \binom{m}{m} = 2^m - 1$$

determinants, while for $p \in P_1$ we need examine only m determinants. If p is symmetric the Hurwitz criterion shows $P = P_1$, since $p \in P$ and $p \in P_1$ are both necessary and sufficient for $p > 0$. However, no such result is true in general and the distinction between P and P_1 must be carefully observed. We shall return to this matter in § 25.

There is an extensive literature pertaining to the relation among classes such as P, A_1 and several others [1], [4], [5], [6], [7], [9], [17], [28], [32]; cf. also [11], [12], [14], [16], [18], [27]. In §§ 4, 5, 6 we summarize those results that are most germane to the present investigation. Although some of the proofs are different from those in the references cited, the results are known, and this part of the paper should be regarded as expository.

4. Triangular matrices. If p is a triangular matrix with $p_{ii} < 0$ for $i = 1, 2, \cdots, m$, then $p \in A_1$ automatically [1], [17]. To see this, let p be upper triangular, so that all elements below the diagonal are 0. Let us then form the matrix $r = ap + p^t a$ together with its principal diagonal determinants

$$D_1 = r_{11}, \qquad D_2 = \det \begin{pmatrix} r_{11} & r_{12} \\ r_{12} & r_{22} \end{pmatrix}, \cdots.$$

The hypothesis $p_{11} < 0$ gives $D_1 < 0$. The term a_2 occurs in D_2 only in r_{22}, and is readily checked that $D_2 > 0$ if a_{22} is large enough. Likewise a_3 occurs in D_3 only in the element r_{33}, and we have $D_3 < 0$ if a_3 is large enough. Continuing in this way, we see that we can choose first $a_1 = 1$, then a_2, then a_3, and so on, in such a way that the Hurwitz criterion for r is satisfied. This shows $r < 0$ and hence $ap < 0$ also. The argument is not quite as obvious if p is lower triangular, but $p \in A_1 \Leftrightarrow p^t \in A_1$ as seen above.

5. A necessary condition. That $p \in A_1$ implies $-p \in P$ is deduced in [7] from a well-known theorem of Lyapunov, to the effect that the matrix equation $ap + p^t a < 0$ has a positive definite symmetric solution a if, and only if the characteristic values of p have negative real parts. It was pointed out by one of the referees that the result also follows from a theorem of Ostrowsky and Taussky [30] which is quoted without proof in [2]; namely, if $A = B + C$, where B is symmetric, C is skew symmetric and $B > 0$, then $\det A > \det B$. This theorem not only gives the desired result virtually by inspection, but also sheds light on other aspects of the problems considered here.

Although the two proofs sketched above are both short and illuminating, the theorems upon which they are based are not quite elementary. Actually the desired result, $p \in A_1$ implies $-p \in P$, can be deduced from the fact that if any matrix A satisfies $A > 0$, then $\det A > 0$. An extremely simple proof of this for symmetric matrices is given by Beckenbach and Bellman [2] and their proof applies to general matrices A without change. In the case at hand we get first $\det(-ap) > 0$, then $\det(-p) > 0$, and finally a similar result for each principal submatrix by the well-known method of specializing the quadratic form. (Take some of the variables to be 0.) This gives $-p \in P$ and completes the proof. Applying the result to $p - \varepsilon I$ and letting $\varepsilon \to 0+$, we see that $p \in A_0$ implies $-p \in P_0$.

The condition $p \in P$ is far from sufficient for $p \in A_1$ or even for $p \in A_0$, but if the off-diagonal elements have the same sign, it is sufficient for a condition of "quasi diagonal dominance" that has been used with success, chiefly by Ladde, in the study of stability [22]–[26]. Examples of other conditions leading to stability are in [8], [10], [14], [29], [34]; we do not want to give the impression that every stability study involves the Volterra hypothesis.

6. Symmetric matrices. Let p be symmetric and $p \in A_0$. Then $-p \in P_0$, as seen above, and hence $p \leqq 0$ by the Hurwitz criterion. In other words, a symmetric matrix cannot be improved by any multiplier a. (As pointed out by one of the referees, this also follows from the theorem of Ostrowsky and Taussky cited above.) Since the result is important in the sequel, we give two additional proofs. The first was obtained jointly with Ernst Straus; the second is due to Robert Steinberg.

Let p be symmetric and suppose $px = \lambda x$ where $\lambda > 0$ and $x \neq 0$. Then for any positive diagonal matrix we have $x^t apx = \lambda x^t ax > 0$ and hence p does not belong to A_0. Thus $p \in A_0$ implies that all characteristic values $\lambda \leqq 0$, and this in turn implies $p \leqq 0$ since p is symmetric.

Steinberg's proof is as follows: If $p = p^t$ and $ap \leqq 0$, where a is diagonal, then

$$\sum_{i,j=1}^{m} p_{ij}(a_i + a_j)y_i y_j \leqq 0$$

for all y_i. Under the further hypothesis that $a > 0$ let $y_i = x_i t^{a_i}$, multiply by dt/t, and integrate from $t = 0$ to $t = 1$. The resulting inequality $x^t p x \leqq 0$ shows $p \leqq 0$.

The first of the two arguments given above can be generalized. Let $px = bx$ where b is diagonal, $b \geqq 0$, and $bx \neq 0$. Then $x^t apx = x^t abx > 0$ for every positive diagonal matrix a and hence p does not belong to A_0. A similar condition is given in [1]. Naturally, the above results extend to A_1.

7. The labeled graph and the class V. As in [16], [37] the matrix p is described by a labeled graph $G(p)$ having m vertices, as follows: The vertices i and j are joined if $p_{ij} \neq 0$ or $p_{ji} \neq 0$, $i \neq j$, and vertex i has a black dot if $p_{ii} < 0$, an open circle if $p_{ii} = 0$. The case $p_{ii} > 0$ does not arise since it excludes the possibility that $p \in A_0$.

Although the individual signs of p_{ij} and p_{ji} for $i \neq j$ do not matter, the sign of the product $p_{ij}p_{ji}$ is crucial. For want of a better terminology we describe the two possibilities as follows:

$p_{ij}p_{ji} < 0$, prey-predator, symbol *pp*,

$p_{ij}p_{ji} > 0$, competitive-symbiotic, symbol *cs*.

Thus, in addition to the labeling introduced above, we have a further labeling of the edges of the graph as *pp* or *cs*.

The terminology "prey-predator" is classical and needs no apology. The case $p_{ij} < 0$, $p_{ji} < 0$ in (1) suggests competition and $p_{ij} > 0$, $p_{ji} > 0$ suggests symbiosis. Without affirming that either case is actually important in ecology, we have preferred to stay within the terminology of that discipline because it is in the field of ecology that the study of (1) has its origin.

Any given system can have *pp* links, *cs* links, and also the case of no coupling, $p_{ij} = p_{ji} = 0$. A fourth possibility is the partial coupling described by $p_{ij} = 0$, $p_{ji} \neq 0$. In this case the vertices (i, j) in $G(p)$ are still adjacent, but $[i, j]$ is referred to as a "partial link". For reasons which will become clear in the course of the work, most results in Part I are formulated for the class V of the following definition:

DEFINITION 3. We denote by V_0 the class of matrices p such that $p_{ii} \leqq 0$ for all i and by V the subclass of matrices $p \in V_0$ such that $p_{ij} = 0 \Leftrightarrow p_{ji} = 0$ for all i, j.

The letter V is in honor of Volterra. It may be mentioned that matrices such that $p_{ij} = 0 \Leftrightarrow p_{ji} = 0$ are termed *combinatorially symmetric* [27].

8. A sufficient condition when the graph is a tree. We shall establish:

THEOREM 1. *Let $p \in V$, let the graph of p be a tree, and let $n(i)$ denote the number of cs edges incident on vertex i in this graph. Suppose the inequality*

$$p_{ij}p_{ji} \leqq \frac{p_{ii}p_{jj}}{n(i)n(j)}$$

holds whenever $i \neq j$, $n(i)n(j) \geqq 1$. Then $p \in A_0$.

It might be thought that strict inequality in Theorem 1 would ensure $p \in A_1$ but this is false even when $m = 2$. For example, the matrix

$$p = \begin{pmatrix} -1 & a \\ -a & 0 \end{pmatrix}$$

satisfies the hypothesis with strict inequality if $a \neq 0$, but does not belong to A_1 for any a. Actually the strict inequality gives $p \in A$ where the class A, discussed in § 26, is more revelant to applications than either A_0 or A_1. A similar remark applies to the analogues and extensions of Theorem 1 which are given in §§ 19–22.

9. Proof of Theorem 1. Let i and j be adjacent vertices, and consider the quadratic form

$$a_i p_{ii} x_i^2 + (a_i p_{ij} + a_j p_{ji}) x_i x_j + a_j p_{jj} x_j^2$$

obtained from $x^t apx$ when all x_k except those for $k = i$ or j are 0. If $r^2 = a_i/a_j$, the above expression is $\leqq 0$ for all x_i, x_j if, and only if,

$$(rp_{ij} + r^{-1} p_{ji})^2 \leqq 4 p_{ii} p_{jj}.$$

If (i, j) is a *pp* link, the left side is minimum when $r^2 = -p_{ji}/p_{ij}$ and in this case the condition holds automatically, since the left side is then 0. If (i, j) is a *cs* link, the left side is minimum when $r^2 = p_{ji}/p_{ij}$ and the condition holds if, and only if,

$$p_{ij} p_{ji} \leqq p_{ii} p_{jj}. \tag{3}$$

In either case the optimum choice of a_k satisfies an equation which was first used by Volterra for *pp* links and was used in the general case in [3], namely,

$$\frac{a_i}{a_j} = \left| \frac{p_{ji}}{p_{ij}} \right|. \tag{4}$$

Since the graph is a tree it is readily checked that one can start with $a_1 = 1$ and determine the a_i recursively in such a way that (4) holds on every edge.

The only remaining problem is that the same vertex i, with coefficient p_{ii}, may be involved in several links of *cs* type. This problem is dealt with in [38]. Namely, let the coefficient p_{ii} be parceled out as

$$p_{ii} = \frac{p_{ii}}{n(1)} + \frac{p_{ii}}{n(i)} + \cdots + \frac{p_{ii}}{n(i)} \quad \text{(to } n(i) \text{ terms)}$$

and similarly for p_{jj}. This has the effect of replacing p_{ii} by $p_{ii}/n(i)$ and p_{jj} by $p_{jj}/n(j)$ in (3), and Theorem 1 follows. It is not necessary to allow any part of p_{ii} for a *pp* link, since the cross product $x_i x_j$ is absent.

If there are no *cs* edges, the hypothesis of Theorem 1 is vacuously fulfilled and we conclude that $p \in A_0$ automatically. This fact was first established by Volterra [39]. In the following section we give a necessary and sufficient condition for $p \in A_0$, in the case of a tree graph, which again generalizes the theorem of Volterra.

10. A necessary and sufficient condition. If we set $\sigma_{ij} = \operatorname{sign} p_{ij} p_{ji}$, the condition (4) to determine the a_j is

$$a_i p_{ij} = a_j p_{ji} \sigma_{ij}. \tag{5}$$

When all edges are of type *cs*, this asserts that the matrix ap is symmetric. As seen in § 6, a symmetric matrix cannot be improved by any multiplier; in other words, we cannot have $bap \leqq 0$ for any multiplier b unless already $ap \leqq 0$. Since an arbitrary multiplier c could be written as $c = ba$, this shows that the choice given by (5) is optimum. Thus we are led to the following necessary and sufficient condition:

THEOREM 2. *Let $p \in V$ have a tree graph and let q denote the matrix obtained from p when every element p_{ij} satisfying $p_{ij} p_{ji} < 0$ is replaced by 0. Then $p \in A_0$ or A_1 if, and only if, $-q \in P_0$ or P_1, respectively.*

The matrix q can also be described as follows: Let $G(p)$ be the graph of p and let $H(p)$ be the graph obtained from $G(p)$ when all *pp* links are removed. Then q is the corresponding matrix, where the word "corresponding" means not only that $G(q) = H$, but that $q_{ij} = p_{ij}$ for all elements p_{ij} that were not altered in the passage from G to H.

As an illustration let p be a 3 by 3 matrix with $p_{13} = p_{31} = 0$ and with $p_{ij}p_{ji} > 0$ for the remaining coefficients, $i \neq j$. Then all links are of *cs* type, the graph is a tree, and Theorem 2 shows $p \in A_0$ if and only if $p_{ii} < 0$ for $i = 1, 2, 3$ and

$$\frac{p_{12}p_{21}}{p_{11}p_{22}} + \frac{p_{23}p_{32}}{p_{22}p_{33}} \leqq 1. \tag{6}$$

To see that the condition of Theorem 2 is sufficient, let a be defined by (4), with $a_1 = 1$, and let $r = ap$. Then $r_{ij} = r_{ji}$ whenever $p_{ij}p_{ji} > 0$ and hence the symmetric part of r is aq. This shows that $r \leqq 0$ if the Hurwitz criterion holds for aq, and only then. Since $\det(aq) = \det(a)\det(q)$, and since a similar relation holds for all the principal subdeterminants of q, the determinants which figure in the Hurwitz criterion for aq have the same sign as those that figure in the criterion for q. This completes the proof of sufficiency.

If all links in p are of type *cs*, the matrix $ap = aq$ is symmetric; hence the choice of a given by (4) cannot be improved, and this shows that the condition in Theorem 2 is necessary. The following lemma gives necessity also when p has some *pp* links.

LEMMA 1. *Let p_1 and p_2 be two matrices in V whose graphs have no vertices in common, and let a new graph be formed by joining a vertex i of p_1 to a vertex j of p_2 by means of a pp edge. If p is the matrix corresponding to that new graph, we have $p \in A_0$ if, and only if, $p_1 \in A_0$ and $p_2 \in A_0$.*

By definition, the matrix p in the lemma is $p = p_1 + p_2 + p_3$, where p_3 has the two new elements p_{ij}, p_{ji} and the remaining elements 0. For proof, if $p \in A_0$, it is evident that $p_1 \in A_0$ and $p_2 \in A_0$, since some of the x_i in $x^t apx$ could be chosen to be 0. Conversely, if $p_1 \in A_0$ and $p_2 \in A_0$, let multipliers a_i be obtained for both matrices. Next multiply all a_i associated with p_2 by a positive constant λ so chosen that

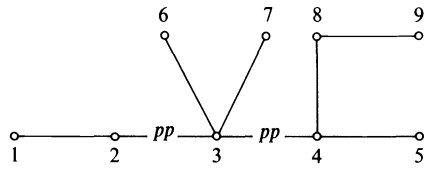
$$\lambda a_j p_{ji} + a_i p_{ij} = 0$$

at the particular i, j pertaining to the new edge. Then this new edge has no effect and the lemma follows.

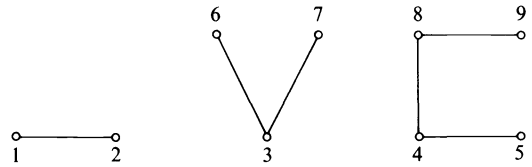
Returning to Theorem 2, suppose the graph of p has at least one *pp* edge and at least one *cs* edge; this is the only case that needs consideration. Remove the *pp* edges to get a set of disjoint trees each of which has *cs* edges only. If p_i are the matrices corresponding to those disjoint trees, repeated use of Lemma 1 shows that $p \in A_0$ if, and only if, each $p_i \in A_0$. But the choice of a used in the proof of Theorem 2 makes each p_i symmetric. Hence a cannot be improved, and the necessity follows.

11. Disconnected graphs. A graph free of loops is a collection of disjoint trees and is called a *forest* [3]. For the most part we assume $G(p)$ connected, so that we consider a tree rather than a forest, though the results apply without change in the latter case. The reason for assuming connectedness is that if $G(p)$ has several components, one can always simplify the analysis by considering each component separately. (Although we have introduced zeros to make all matrices of the same order, m, in the theoretical development, this is not necessary when the results are used for computation.) Even if $G(p)$ is connected, the graph $G(q)$ in Theorem 2 need not be. Introduction of q often allows us to separate the graph by removal of all *pp* links.

As an illustration suppose $G(p)$ is as follows, where the edges not labeled are of type *cs*:



To test whether $p \in A_0$ it suffices to examine matrices corresponding to the graphs



The relevant coefficients are the same as those in p, but the orders are 2 by 2, 3 by 3, and 4 by 4, respectively.

Although there is no loss of generality in assuming $G(p)$ connected for the primary results, as we have seen, there is some loss when these results are used in the proof of further theorems. The reason is that a principal submatrix of p need not have a connected graph even when $G(p)$ is connected. This difficulty arises in connection with Lemma 2 below and in some other cases. Let us state, therefore, that Theorem 2 (and other results of a similar format) apply to a forest as well as to a tree, and they will be so used without further comment. Lemma 2 below is explicitly formulated for a forest, but this is only because the assumption that $G(p)$ is a tree leads to trouble in the inductive proof.

12. The theorem of Berman and Herschkowitz. Theorem 2 is related to an interesting theorem of Berman and Herschkowitz [3] which appeared after this paper was submitted for publication. Their result states that, when $G(p)$ is a forest, the condition $-p \in P$ is necessary and sufficient for $p \in A_1$. Replacement of p by q often separates the graph, leading to simpler calculations, but that is not the main difference in the two theorems. The main difference is that Theorem 2 is based upon the class P_1 rather than P. That the distinction is essential is shown by the example

$$p = \begin{pmatrix} -1 & 0 & a \\ 0 & -1 & 1 \\ -a & 2 & -1 \end{pmatrix}, \qquad q = \begin{pmatrix} -1 & 0 & 0 \\ 0 & -1 & 1 \\ 0 & 2 & -1 \end{pmatrix}.$$

Here $-p \in P_1$ whenever $|a| > 1$ but $-p$ does not belong to P for any a and hence p does not belong to A_1 for any a. (This agrees with Theorem 2, since $-q$ is not in P_1.) The practical significance of the distinction is seen when we take a value such as $m = 20$, which is entirely realistic in the context of differential equations. With $m = 20$ one would have to check over a million determinants for the criterion $-p \in P$, but only 20 determinants (some of which may be simpler than those in p) for the criterion $-q \in P_1$.

There is one respect, however, in which the theorem of Berman and Herschkowitz is superior to Theorem 2. Namely, we have assumed $p \in V$, while no such assumption is needed in [3]. (The ingenious proof is based on a result of Berman, Varga and Ward

[4], [5] to the effect that $p \in A_1$ if, and only if, for every nonzero symmetric positive definite matrix s, the matrix sp has a negative diagonal element.) It will be seen that their result, together with our Theorem 2, yields the extension embodied in Theorem 3 below. We also sketch a proof which is independent of [3].

13. Omission of the hypothesis $p \in V$. We shall establish:

THEOREM 3. *Let $G(p)$ be a tree and let q be formed from p by replacing all elements p_{ij} with $p_{ij}p_{ji} \leqq 0$ by 0. Then $-q \in P_1 \Rightarrow p \in A_1 \Rightarrow -q \in P$.*

In other words, a partial link can be treated like a pp link when Theorem 2 is used as a criterion for $p \in A_1$. It should be observed that Theorem 3 says nothing about the classes P_0 or A_0. The example

$$p = \begin{pmatrix} -1 & 1 \\ 0 & 0 \end{pmatrix}, \qquad q = \begin{pmatrix} -1 & 0 \\ 0 & 0 \end{pmatrix}$$

has $-q \in P_0$ while p is not in A_0. Hence, Theorem 3 would be false if extended to P_0 and A_0 after the fashion of Theorem 2.

14. A lemma. Theorem 3 is based upon the following lemma, which is of independent interest:

LEMMA 2. *Let $G(p)$ be a forest and let $s = p_{ij}$, $t = p_{ji}$ where $i \neq j$ and where s and t are regarded as variables. Then $\det p = Ast + D$ where A and D are independent of s and t. Hence, if one of the variables s or t is 0, the determinant is independent of the other variable.*

Regardless of the nature of the graph, it is readily checked that the determinant has the form

$$\det p = Ast + Bs + Ct + D$$

where the coefficients A, B, C, D are independent of s and t. The lemma asserts that the linear terms are absent if $G(p)$ has no loops.

Lemma 2 can be deduced from Theorem 2 stated for a forest. If there is a counterexample, that is, a matrix p with $G(p)$ a forest and B or C different from 0, we can make a small perturbation and ensure $p \in V$. Considering $p - \lambda I$, where λ is a large constant, we can also make the matrix q associated with p for $s = t = 0$ satisfy the condition $-q \in P_1$ of Theorem 2. (Here we use the fact that the nonzero coefficient is a polynomial in λ, which, being nonzero for $\lambda = 0$, must also be nonzero for large λ.) In short: If there is a counterexample, there is one that satisfies the hypothesis of Theorem 2 when $s = t = 0$. Theorem 2 then shows that $p \in A_1$ whenever $st < 0$, hence $\det p$ has one and the same sign when $st < 0$, and this is impossible in the presence of linear terms.

The above proof is brief and arises naturally out of the subject of this paper, but it uses analysis to get a result which is obviously algebraic. A simple algebraic proof was found by Prof. Robert Steinberg and, with permission, his proof is presented next. Since $G(p)$ is a forest, it has a free end. Without loss of generality let the free end have the label 1 and the single vertex adjacent thereto the label 2; if no vertex is adjacent, then $p_{12} = p_{21} = 0$. In any case $p_{1j} = p_{j1} = 0$ for $j \geqq 3$ and the matrix p has the form

$$\begin{pmatrix} p_{11} & p_{12} & 0 & 0 & \cdots \\ p_{21} & p_{22} & p_{23} & p_{24} & \cdots \\ 0 & p_{32} & p_{33} & p_{34} & \cdots \\ 0 & p_{42} & p_{43} & p_{44} & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \end{pmatrix}.$$

We assume that the lemma is known for all matrices of order less than m with a loop-free graph and we use mathematical induction. There are two cases.

Case I. $s=p_{12}$, $t=p_{21}$. In this case expansion on the first row followed by expansion of the second determinant so obtained on the first column gives a result of the form $\det p = D + stA$ by inspection.

Case II. s, t are in the submatrix obtained when the first row and first column of p are deleted. Here expansion as above leads to two determinants, of orders $m-1$ and $m-2$, respectively, to which the induction hypothesis applies. The result is, an obvious notation,

$$\det p = p_{11}(A_1 st + D_1) - p_{12}p_{21}(A_2 st + D_2).$$

Since this has the form $Ast + D$, Lemma 2 follows by induction.

15. Proof of Theorem 3. To obtain Theorem 3, let us form a matrix $\tilde{p}$ intermediate between p and q, as follows: in $\tilde{p}$, every pp link is left unchanged, but every partial link is removed. Thus, $\tilde{p}_{ij}=0$ whenever $p_{ij}p_{ji}=0$ and otherwise $\tilde{p}_{ij}=p_{ij}$. In an obvious notation

$$-q \in P_1 \Rightarrow -\tilde{q} \in P_1 \Rightarrow \tilde{p} \in A_1 \Rightarrow -\tilde{p} \in P \Rightarrow -p \in P \Rightarrow p \in A_1$$

where the implications are justified, in order, by: $\tilde{q}=q$, Theorem 2, § 5, Lemma 2, and [3]. To see that $p \in A_1 \Rightarrow -q \in P$, let us form a matrix $\bar{p}$ close to p by making each partial link into a link of pp type. Then $\bar{p} \in V$ and

$$p \in A_1 \Rightarrow \bar{p} \in A_1 \Rightarrow -\bar{q} \in P \Rightarrow -q \in P$$

where the justification now is: $\bar{p}$ close to p, Theorem 2, and $\bar{q}=q$.

It is of some interest that the first of the above results can be deduced in the form $-q \in P_1 \Rightarrow -q \in P \Rightarrow p \in A_1$ without any use of Lemma 2 or [3], and the method gives a new approach to the result from [3] used above. The first of these implications follows from Theorem 2 and § 5 applied to q instead of p. To get the second, we assume existence of at least one partial link $[i,j]$ and we use induction. Removing the link $[i,j]$ separates the graph and produces two matrices which, by the induction hypothesis, both belong to A_1. This remains true if we replace p_{ii} by $(1-\varepsilon)p_{ii}$ and p_{jj} by $(1-\varepsilon)p_{jj}$ where $\varepsilon > 0$ is sufficiently small. Restoring the link then leads to the additional quadratic form

$$a_i \varepsilon p_{ii} x_i^2 + (a_i p_{ij} + a_j p_{ji}) x_i x_j + a_j \varepsilon p_{jj} x_j^2$$

and the proof is completed much as in the proof of Lemma 1.

16. Tridiagonal matrices. Perhaps the simplest tree graph is a single strand, that is, a graph in which every node except the two ends has the order 2. Such a graph corresponds to a tridiagonal matrix. We assume all links of *cs* type since in the contrary case the graph breaks up into the graphs of smaller tridiagonal matrices as explained above. The case $m=2$ is trivial and $m=3$ leads to (6). If $m=4$ and all links are of *cs* type, it is readily checked that $p_{ii}<0$ is necessary for $i=1,2,3,4$ and hence we can introduce the expressions

$$\bar{B}_{ij} = \frac{p_{ij}p_{ji}}{p_{ii}p_{jj}} \tag{7}$$

for $i,j=1,2,3,4$, each of which is positive. Theorem 2 with $q=p$ gives the following condition which, with $p_{ii}<0$, is necessary and sufficient for $p \in A_0$:

$$\bar{B}_{12}<1, \quad \bar{B}_{34}<1, \quad \bar{B}_{23} \leqq (1-\bar{B}_{12})(1-\bar{B}_{34}). \tag{8}$$

It should be observed that (8) implies $\bar{B}_{12}+\bar{B}_{23}<1$ and $\bar{B}_{23}+\bar{B}_{34}<1$. This agrees with (6) applied to the principal 3 by 3 submatrices of p.

The technique used in the proof of Theorem 1 leads to (8) without any use of determinants, as seen next. Let us set

$$p_{ii}=c_i p_{ii}+(1-c_i)p_{ii}, \qquad 0\leqq c_i\leqq 1, \tag{9}$$

noting that Theorem 1 is obtained by taking each $c_i=0, 1/2$ or 1. Using (3) as in the proof of Theorem 1, we get the sufficient condition

$$\bar{B}_{12}\leqq c_1, \quad \bar{B}_{23}\leqq(1-c_1)c_2, \quad \bar{B}_{34}\leqq(1-c_2) \tag{10}$$

under the hypothesis $p_{ii}<0$, which is always needed. A moment's thought shows that the optimum choice of c_i is obtained by requiring equality in all relations (10) except the last [38]. This leads again to (8).

A similar procedure applies to the general m by m tridiagonal matrix and leads to the same necessary and sufficient condition as that given by Theorem 2. For numerical calculation, it is best not to solve for the c_i algebraically but to determine them recursively starting with c_1. Theorem 1 gives the sufficient condition

$$\bar{B}_{12}\leqq\tfrac{1}{2}, \quad \bar{B}_{23}\leqq\tfrac{1}{4}, \cdots, \bar{B}_{m-2\,m-1}\leqq\tfrac{1}{4}, \quad \bar{B}_{m-1\,m}\leqq\tfrac{1}{2}$$

whereas introduction of the c_i gives a refinement which is necessary and sufficient.

When the above technique using c_i is applied to a matrix with a more general tree graph, difficulty is encountered at each node of order $\geqq 3$. The optimum determination of the c_i cannot be accomplished recursively, but depends on the solution of a system of simultaneous equations. For a necessary and sufficient condition, the criterion of Theorem 2 is simpler. However a tolerably simple sufficient condition, sharper than that of Theorem 1, can be obtained as follows: At any node i of order $k\geqq 3$ write $p_{ii}=k(p_{ii}/k)$ and use the value p_{ii}/k instead of p_{ii} on each branch which terminates at i. On each branch the c_i are then determined recursively as above. Although a formal statement will not be given here, it can be said that the resulting criterion is intermediate between that of Theorem 1 and that of Theorem 2, both in sharpness and in computational complexity.

17. Balanced matrices. The distinct vertices $1, 2, \cdots, k$ of $G(p)$ form a loop if i and $i+1$ are adjacent for $i=1, 2, \cdots, k-1$ and $k, 1$ are also adjacent. The loop is denoted by $[1, 2, \cdots, k]$. It is said that the loop is balanced if

$$|p_{12}p_{23}\cdots p_{k\,k-1}p_{k1}|=|p_{21}p_{32}\cdots p_{k-1\,k}p_{1k}|.$$

A similar definition applies to any loop, with a more elaborate use of subscripts, and the matrix p is said to be balanced if every loop in $G(p)$ is balanced. Further discussion of the history and use of this condition can be found in [38], where it is also shown that, if two loops are balanced, so is the loop formed by their union. Thus, the criterion need be verified only for some basic set of loops in $G(p)$.

The above formulation involving products has been taken as primary partly because of its historical origin in the work of Volterra and partly because it provides a specific decision procedure. But a simple and illuminating interpretation of this condition was given by one of the referees for this paper; namely, p is admissible if, and only if, the matrix $\tilde{p}=(|p_{ij}|)$ is diagonally similar to a symmetric matrix. To see this, suppose $b^{-1}\tilde{p}b=s$ where b is a nonsingular diagonal matrix and s is symmetric. The equation $s_{ij}=s_{ji}$ gives $a_i|p_{ij}|=a_j|p_{ji}|$ with $a_i=(b_i)^{-2}$. Hence p is balanced. Conversely, if p is balanced, we can determine $a_i>0$ as above and the choice $b_i=(a_i)^{-1/2}$ gives the matrix b.

We shall establish the following:

THEOREM 4. *If $p \in V$ is balanced, the sufficient conditions of Theorem 1 for $p \in A_0$ and of Theorem 2 for $p \in A_0$ or $p \in A_1$ remain sufficient even when $G(p)$ is not a tree. If, in addition, all links are of cs type, the conditions in Theorem 2 with $p = q$ are both necessary and sufficient.*

The proof is essentially unchanged, since the hypothesis of being balanced is equivalent to the hypothesis that positive multipliers satisfying (4) can be found. In the special case of a tree there are no loops and the matrix is balanced vacuously. Thus Theorem 4 contains the sufficiency part of Theorems 1 and 2.

The proof of necessity breaks down, because there is no analogue of Lemma 1 when $G(p)$ has loops. (The trouble is that *pp* links can be removed without separating the graph.) As an illustration, let p be the balanced matrix p_0 or p_1 where

$$p_0 = \begin{pmatrix} -2 & 2 & 16 \\ 2 & -2 & 1 \\ -16 & 1 & -2 \end{pmatrix}, \qquad p_1 = \begin{pmatrix} -8 & 7 & 64 \\ 7 & -8 & 4 \\ -64 & 4 & -8 \end{pmatrix}.$$

Replacing the elements 16, −16, 64, −64 by 0 gives the matrices q_0 and q_1, respectively, and it is readily checked that $\det q_i > 0$, so that the condition $-q \in P_0$ (and of course $-q \in P_1$) of Theorem 2 fails. Nevertheless the multiplier

$$a = \operatorname{diag}(15, 15, 17)$$

gives $r = ap + p^t a$ in the form

$$r_0 = \begin{pmatrix} -60 & 60 & -32 \\ 60 & -60 & 32 \\ -32 & 32 & -68 \end{pmatrix}, \qquad r_1 = \begin{pmatrix} -240 & 210 & -128 \\ 210 & -240 & 128 \\ -128 & 128 & -272 \end{pmatrix}$$

respectively. By a short calculation $-r_0 \in P_0$ and $-r_1 \in P_1$, so that $p_0 \in A_0$ and $p_1 \in A_1$ by the Hurwitz criterion.

The genesis of these examples is as follows. The matrix p_0 together with its multiplier a was constructed by use of Theorem 8 below. Then p_1 was obtained by a slight modification of $4p_0$. We need p_1 to see that $p \in A_1$ is possible for a balanced matrix even if the corresponding matrix q does not satisfy $-q \in P_0$. No such example can be obtained from Theorem 8 directly, since the theorem pertains to the class A_0.

18. A class of sufficient conditions. A connected graph $G(p)$ can always be reduced to a tree by removing a suitable set of links $[i, j] = L_{ij}$. A sufficient condition for $p \in A_0$ is that the matrix $\tilde{p}$ corresponding to that tree shall satisfy $\tilde{p} \in A_0$, with enough slack so that we do not leave the class A_0 when restoring the links L_{ij}. This point of view leads to a number of conditions for $p \in A_0$ which were introduced in [38].

Since we propose to generalize the main result of [38], a brief summary is given here. If the set $\{L_{ij}\}$ of links is removed to reduce $G(p)$ to a tree, let $N(i)$ denote the number of times the index i appears as a subscript in an enumeration of the elements L_{ij}. Then a sufficient condition for $p \in A_0$ is

$$A_{ij} \leqq \frac{4B_{ij}}{N(i)N(j)} \quad \text{for each } L_{ij}$$

where A_{ij} and B_{ij} are certain numerical quantities which are easily computed from knowledge of p. Precise definitions of these quantities are given later; at the moment, we wish only to give the general flavor of the theorem.

The analysis in [38] makes essential use of the hypothesis that all links are of type *pp*. When this condition is dropped, we encounter several new problems, the most important of which is that the matrix $\tilde{p}$ belonging to the tree need not satisfy $\tilde{p} \in A_0$ automatically. (It is, in part, in anticipation of this problem that we have presented the analysis leading to Theorems 1, 2, 3.) The extension of the main theorem [38] to allow *cs* links is discussed next. We begin with the case in which $G(p)$ is a loop, since that case forms the basis for further developments.

19. A single loop. Let the graph of p consist of the single loop $[1, 2, \cdots, m]$. If $p \in V$, as now assumed, we can determine a_i by $a_i = 1$ and

$$\frac{a_1}{a_2} = \left|\frac{p_{21}}{p_{12}}\right|, \quad \frac{a_2}{a_3} = \left|\frac{p_{32}}{p_{23}}\right|, \quad \cdots, \quad \frac{a_{m-1}}{a_m} = \left|\frac{p_{mm-1}}{p_{m-1m}}\right|.$$

If only x_1 and x_m are different from 0, the condition $x^t apx \leqq 0$ is equivalent to

$$(a_1 p_{1m} + a_m p_{m1})^2 \leqq 4 a_1 a_m p_{11} p_{mm}.$$

Dividing both sides by $a_1 a_2 |p_{1m} p_{m1}|$, we get the equivalent condition

$$R + \frac{1}{R} + 2\sigma \leqq 4 \frac{p_{11} p_{mm}}{|p_{1m} p_{m1}|} \tag{11}$$

where $R = (a_1/a_m)(|p_{1m}/p_{m1}|)$ and $\sigma = \text{sign}\,(p_{1m} p_{m1})$; clearly

$$R = \left|\frac{p_{21} p_{32} \cdots p_{m\,m-1} p_{1m}}{p_{12} p_{23} \cdots p_{m-1\,m} p_{m1}}\right|.$$

Let us denote the left side of (11) by A and the right side by $4B$. Thus

$$A = R + \frac{1}{R} - 2 \quad \text{or} \quad A = R + \frac{1}{R} + 2$$

according as $[1, m]$ is a link of type *pp* or *cs*. The former value of A vanishes when the loop is balanced, since $R = 1$ in that case, and is referred to in [38] as the *measure of asymmetry* of the loop. By contrast, the latter value is 4, and not 0, when the loop is balanced. Nevertheless, it is minimum when $R = 1$, and is referred to again as a measure of asymmetry. The measure changes its value, according as the link $[1, m]$ is of type *pp* or of type *cs*.

The quantity B agrees with that in [38] and is referred to as the *measure of strength* of the link $[1, m]$. We mention in passing that B here and B_{ij} introduced later are reciprocals of the quantities $\bar{B}_{ij}$ introduced in § 16.

Let us now discuss the condition $A \leqq 4B$ of (11) in more detail. If the two links $[1, 2]$ and $[m-1, m]$ adjacent to $[1, m]$ are both of type *pp*, the above choice of the a_i makes the associated cross product terms disappear and the link $[1, m]$ is effectively isolated. Thus, a sufficient condition for $p \in A_0$ is that $A \leqq 4B$, and that the graph obtained when p_{11}, p_{mm}, p_{1m}, p_{m1} are all replaced by 0 shall belong to A_0. The latter is a tree, in fact a tree corresponding to a tridiagonal matrix, and is covered by the preceding discussion. The condition is both necessary and sufficient if the only possible choice of the a_i, to make the tree belong to A_0, is that used in the derivation of (11).

If one or both of the links adjacent to $[1, m]$ are of type *cs*, some part of p_{11}, p_{mm} or both must be used with the tree if the tree is to belong to A_0. As above we write

$$p_{11} = c_1 p_{11} + (1 - c_1) p_{11}, \qquad p_{mm} = c_m p_{mm} + (1 - c_m) p_{mm}$$

with $0 \leqq c_j \leqq 1$, and we require

$$A = 4B(1-c_1)(1-c_m) \tag{12}$$

instead of $A \leqq 4B$. Here we prescribe = rather than $\leqq$ because it is permissible, and is optimum for the rest of the calculation involving the tree. The tree is now obtained by replacing p_{11}, p_{mm}, p_{1m}, p_{m1} respectively by $c_1 p_{11}$, $c_m p_{mm}$, 0 and 0. If this tree belongs to A_0, then (12) is sufficient for $p \in A_0$.

When only one of the links adjacent to $[1, m]$ is of type *cs*, the value c_j for the other link can be taken to be 0, and the above procedure is both specific and easy to implement. If both links are of type *cs*, however, there is a difficulty in making an optimum choice of the c_j, since only the product $(1-c_1)(1-c_m)$ is determined by (12). A simple sufficient condition is obtained by the choice $c_1 = c_m = 1/2$, corresponding to the choice used in Theorem 1.

20. Generalization. We now discuss what is involved when the theory developed in [38] is extended to allow *cs* links. Since the main features were illustrated in the discussion of § 19, we shall be brief.

For any loop having i, j as adjacent vertices we define a ratio of products R_{ij} analogous to R of § 19 (see [38]) and the measure of asymmetry is then

$$A_{ij} = R_{ij} + \frac{1}{R_{ij}} + 2 \operatorname{sign}(p_{ij}p_{ji}).$$

Although R_{ij} does not depend on the particular choice of adjacent vertices, the quantity A_{ij} does depend on this choice, in general, having one value if $[i, j]$ is a *pp* link and another if $[i, j]$ is a *cs* link. The first major change is that this more elaborate *measure of asymmetry* must be used instead of the A_{ij} in [38]. The latter was defined for a *pp* link only. On the other hand the *measure of strength*

$$B_{ij} = \frac{p_{ii}p_{jj}}{|p_{ij}p_{ji}|},$$

like the definition of R_{ij}, remains unchanged.

As in [38] we remove links L_{ij} from the graph $G(p)$ until the resulting graph is a tree, T. If T has some *cs* links, the matrix q associated with T does not belong to A_0 automatically, and $q \in A_0$ must be imposed as a separate condition. This is the second major change from the theory as presented in [38].

Finally, if some of the links adjacent to the L_{ij} are of type *cs*, we must borrow from the coefficients p_{ii} and p_{jj} as in the previous discussion, so that the measure of strength B_{ij} is replaced by

$$B_{ij}(1-c_i)(1-c_j).$$

This is the third major change.

When all these matters are taken into account, one gets a generalization of the results [38] to allow *cs* links. We shall give a formal statement only when the above constants c_k can be taken to be 0, and to this end we introduce the following definition:

DEFINITION 3. A link or set of links is said to be isolated if all other links adjacent thereto are of type *pp*.

Here two links are considered to be *adjacent* if they have a vertex in common.

THEOREM 5. *Let $p \in V$ and let $\{L_{ij}\}$ be a set of disjoint, isolated links whose removal changes $G(p)$ to a tree. Let q be the matrix corresponding to this tree. Then $p \in A_0$ if $q \in A_0$ and if the inequality $A_{ij} \leqq 4B_{ij}$ holds for each link L_{ij} in the set $\{L_{ij}\}$.*

If i occurs as a subscript on an L in the set $\{L_{ij}\}$, then we take $q_{ii}=0$. This choice does not affect the question whether $q \in A_0$. The quantities A_{ij} are determined from the tree graph, as follows: Let the single link L_{ij} be restored to the tree graph, so that L_{ij} is now part of a unique loop. Then A_{ij} is found by the procedure of § 19 applied to this loop.

21. A necessary and sufficient condition. For a broad class of matrices the condition of Theorem 5 is necessary as well as sufficient. Description of this class depends on the following:

DEFINITION 4. A *pp* link L_{ij} is critical if $p_{ii}p_{jj}=0$, and a *cs* link is critical if $p_{ii}p_{jj}=p_{ij}p_{ji}$. The matrix $q \in V$ is critical if every link in $G(q)$ is critical and all the *cs* links are isolated.

A moment's thought shows that a critical matrix belongs to A_0 if, and only if, all its loops are balanced. In particular, a critical matrix with a tree graph always belongs to A_0.

We shall establish the following:

THEOREM 6. *Under the hypothesis of Theorem* 5 *suppose the matrix q is critical. Then the condition $A_{ij} \leqq 4B_{ij}$ there given is both necessary and sufficient for $p \in A_0$.*

For proof, consideration of adjacent vertices $[i, j]$ in $G(q)$ shows that (4) is necessary for $p \in A_0$, and hence a must agree with the multiplier used in § 19. The fact that a is essentially unique, and is determined over the tree by (4), has two effects. First, it shows that the coefficients a_ip_{ij} and a_jp_{ji} associated with the link L_{ij} have the values that lead to the A_{ij}. Second, it shows that the cross-product term associated with any *pp* link in the tree for *ap* is missing. Since each L_{ij} is connected to the tree only by *pp* links, it is effectively isolated from the tree. The L_{ij} are disjoint from each other by hypothesis. Thus, if we set $x_k=0$ for $k \neq i$ or j, the resulting quadratic form is precisely the form that led to the condition $A_{ij} \leqq 4B_{ij}$. If $ap \leqq 0$, this form is $\leqq 0$, and the necessity follows.

22. Adjacent links. So far, it has been assumed that the links L_{ij} are disjoint. If this is not the case, we can account for the overlapping as in [38, Thm. 1]. Namely, let the links in the set L_{ij} be arranged in a list, each link appearing just once, and let $N(i)$ denote the number of times the index i occurs in this list as a subscript on L. Then Theorem 5 continues to hold if the measure of strength B_{ij} is divided by $N(i)N(j)$. A particularly neat version is obtained if the tree graph for q is analyzed by Theorem 1 instead of by the necessary and sufficient condition given in Theorem 2. For the reader's convenience the definition of $n(i)$ in Theorem 1 is repeated here, namely, $n(i)$ is the number of *cs* edges incident on the vertex i. We then have the following:

THEOREM 7. *Let $p \in V$ and let $\{L_{ij}\}$ be an isolated set of links whose removal changes $G(p)$ to a tree. Then $p \in A_0$ if the inequalities*

$$A_{ij} \leqq \frac{4B_{ij}}{N(i)N(j)}, \qquad 4 \leqq \frac{4B_{ij}}{n(i)n(j)}$$

hold for the links L_{ij} in the set L_{ij} and for the cs links of $G(p)$ which are not among the L_{ij}, respectively.

Theorems 5 and 7 give conditions under which a stable system with matrix q, where $G(q)$ is a tree, remains stable when links are added to form loops. The results are easy to apply, even when p is large, provided $G(p)$ has few loops and plenty of *pp* links. But when these conditions are not fulfilled it is better to bypass the theorems and go back to the underlying idea. Namely, reduce $G(p)$ to a tree, determine a_i for the tree as in § 9, and see if the original matrix p with this multiplier satisfies

$ap+p^ta \in P_1$. The choice of a is motivated by Theorems 2, 3, 6 and by § 9, and is not wholly random. But it takes account of the 2 by 2 submatrices only and is far from coming to grips with the main problems.

To put these remarks into perspective consider the matrices

$$-p=\begin{pmatrix} 2 & 3 & 0 & 1 \\ 1 & 2 & 1 & 3 \\ 0 & 1 & 3 & 3 \\ -1 & 0 & 1 & c \end{pmatrix}, \qquad -q=\begin{pmatrix} 2 & 3 & 0 & 1 \\ 1 & 2 & 1 & 0 \\ 0 & 1 & 3 & 0 \\ -1 & 0 & 0 & c \end{pmatrix}$$

where q is obtained by dropping the links [4, 2] and [3, 4]. By Theorem 2 we have $q \in A_1$ if $c>0$ and we ask: How large must c be if $p \in A_1$? The above procedure leads quickly to the multiplier $a=\operatorname{diag}(1,3,3,1)$ and to the sufficient condition $c \geqq 13.7$. The much more laborious methods of Part II show that $c>11$ is necessary, $c \geqq 11.7$ is sufficient, and the multiplier is $a=\operatorname{diag}(1, 2.6, 5, 2.8)$.

23. Matrices with a critical link. If a matrix $p \in V$ has a critical link $[i, j]$ then, in general, the problem of deciding whether $p \in A_0$ can be effectively solved, and the solution is of about the same difficulty as direct application of the Hurwitz criterion to decide whether merely $p \geqq 0$. Here the phrase "in general" means that the elements of p do not satisfy any other special equality beyond the condition of criticality occurring in the hypothesis. The precise nature of the excluded equalities will be clear from the following discussion.

Let us recall first that if E is the elementary matrix such that premultiplication by E interchanges two rows of p, then postmultiplication by E interchanges the corresponding two columns, and furthermore, $E=E^t=E^{-1}$. When two rows and also the corresponding two columns are interchanged, p is changed to $q=EpE$ and a to $b=EaE$. The identity

$$ap+p^ta=E(bq+q^tb)E$$

shows that $q \in A_0$ if, and only if, $p \in A_0$. Thus we can make such a simultaneous interchange of rows and columns as often as we like, and move any principal matrix of p into the upper left-hand corner. Hence, without loss of generality, the critical link in the hypothesis can be taken to be [1, 2].

The case in which this link is of *pp* type is trivial but is presented for completeness. Since $p_{11}p_{22}=0$ for such a link, we can interchange two rows and two columns again, if necessary, and assume $p_{11}=0$. The matrix $r=ap+p^ta$ then has $r_{11}=0$, and consideration of appropriate 2 by 2 submatrices of r gives $r_{1j}=0$ if $r \geqq 0$. If no pair (p_{1j}, p_{j1}) reduces to $(0,0)$ for $2 \leqq j \leqq m$, then necessarily $p_{1j}p_{j1}<0$ and a_j is uniquely determined by $a_j=-p_{1j}/p_{j1}$, aside from the scale factor $a_1=1$. Thus $p \in A_0$ if, and only if, $r \geqq 0$ with this choice of a. Since the first row and column of r are 0, r can be replaced by the $m-1$ by $m-1$ matrix q obtained when the first row and column of r are deleted. Since q is symmetric, it can be tested by the Hurwitz criterion, and hence the above remarks give a complete solution to the problem of deciding whether $p \in A_0$ when p has a critical *pp* link.

24. Matrices with a critical link, continued. Let us assume next that [1, 2] is a critical link of *cs* type, so that $p_{11}p_{22}=p_{12}p_{21}>0$. If $r=ap+p^ta$ as above, then $r \geqq 0$ implies $a_1p_{12}=a_2p_{21}$ and hence $r_{11}r_{22}=r_{12}r_{21}$. This shows that the rows of

$$\begin{pmatrix} r_{11} & r_{12} \\ r_{21} & r_{22} \end{pmatrix}$$

are linearly dependent, so that $r_{11}+\lambda r_{21}=0$ and $r_{12}+\lambda r_{22}=0$ for some scalar λ. If we add λ times the second row of r to the first row, and λ times the second column of that matrix to the first column, the result is a matrix of form

$$P=\begin{pmatrix} 0 & 0 & c_3 & c_4 & \cdots \\ 0 & r_{22} & r_{23} & r_{24} & \cdots \\ c_3 & r_{32} & r_{33} & r_{34} & \cdots \\ c_4 & r_{42} & r_{43} & r_{44} & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \end{pmatrix}$$

where $c_j=r_{1j}+\lambda r_{2j}$. If E is the elementary matrix such that premultiplication by E adds λ times the second row to the first then, $P=ErE^t$ and hence $r\leqq 0$ if, and only if $P\leqq 0$. Consideration of suitable 2 by 2 submatrices gives $c_j=0$, if $P\leqq 0$, hence also if $r\leqq 0$. This is, therefore, a necessary condition.

By a short calculation $\lambda=-p_{11}/p_{12}$ and the fact that $a_1p_{12}=a_2p_{21}$ gives

$$p_{12}c_j=a_2(p_{21}p_{1j}-p_{11}p_{2j})+a_j(p_{12}p_{j1}-p_{11}p_{j2}).$$

Thus we have established the following:

THEOREM 8. *Let $p\in V_0$, let $m\geqq 3$, and let $p_{11}p_{22}=p_{12}p_{21}>0$. Suppose the expressions*

$$e_j=p_{11}p_{2j}-p_{21}p_{1j},\qquad f_j=p_{11}p_{j2}-p_{12}p_{j1}$$

do not vanish simultaneously for any j, $3\leqq j\leqq m$. Then $p\in A_0$ if, and only if, $e_jf_j<0$ for $3\leqq j\leqq m$ and the $m-1$ by $m-1$ matrix

$$q=\begin{pmatrix} q_{22} & q_{23} & \cdots & q_{2m} \\ q_{32} & q_{33} & \cdots & q_{3m} \\ \cdots & \cdots & \cdots & \cdots \\ q_{m2} & q_{m3} & & q_{mm} \end{pmatrix}$$

satisfies $q\leqq 0$ where $q_{2j}=p_{2j}$ and $q_{ij}=-(e_i/f_i)p_{ij}$, $i\geqq 3$. The choice of a making $ap\leqq 0$ is uniquely determined by $a_1=p_{21}/p_{12}$, $a_2=1$, $a_j=-e_j/f_j$ for $j\geqq 3$.

Theorem 8 leads to a sufficient condition for $p\in A_0$ even without the hypothesis $p_{11}p_{22}=p_{12}p_{21}$. If $p\in A_0$ and this hypothesis fails, the only possibility is that $p_{11}p_{22}>p_{12}p_{21}$. This follows from the fact that p must satisfy the Hurwitz condition, as noted in § 5, but also follows from the more elementary remark [12] that, in the 2-by-2 case, the Hurwitz inequalities are both necessary and sufficient for $p\in A_0$. Let us therefore replace p_{11} by αp_{11} and p_{22} by βp_{22} where $\alpha\beta p_{11}p_{22}=p_{12}p_{21}$, $0<\alpha\leqq 1$, $0<\beta\leqq 1$. Theorem 8 applies to the new matrix $\tilde{p}$ so obtained and gives a sufficient condition for $p\in A_0$, since

$$x^tapx=x^ta\tilde{p}x+(1-\alpha)a_1p_{11}x_1^2+(1-\beta)a_2p_{22}x_2^2.$$

The use of two parameters α and β, rather than one, is sometimes helpful to ensure that $\tilde{e}_j$ and $\tilde{f}_j$ do not vanish simultaneously for $\tilde{p}$.

25. A closure property. In these concluding sections we give two properties of the class V that bear on the appropriateness of the restriction $p\in V$ in practical problems. The first pertains to a distinction which we have already emphasized; namely, $p\in P$ requires examination of 2^m-1 determinants, in general, while only m determinants are needed for $p\in P_1$. If p is symmetric or satisfies the hypothesis of Kotelanskii's theorem [9], [11], the problem disappears, but these are very special cases. Furthermore, even these cases lead to no simplification of the weak inequalities associated with P_0. Thus, to test for $p\in A_0$ by the criterion $-q\in P_0$ of Theorem 2 requires examination of

$2^m - 1$ determinants, and to test for $p \in A_0$ by the criterion $-(q+q') \in P_0$ of Theorem 8 requires examination of $2^{m-1} - 1$ determinants.

The question arises whether we could test $p - \varepsilon I$ for membership in A_1 and then deduce membership in A_0 by letting $\varepsilon \to 0+$. If so, the way is open to base the analysis upon the class P_1 rather than P_0. A significant advantage of the hypothesis $p \in V$ is that it does, in fact, allow such a procedure:

THEOREM 9. *Any limit point of A_0 which is in V is also in A_0.*

The precise meaning is that if $p_n \in A_0$ and $\lim p_n = p \in V$, then $p \in A_0$. Convergence is thought to be based on any of the usual topologies, or equivalently, the limit can be considered elementwise.

The hypothesis $p \in V$ is essential. For example, if p is a triangular matrix with zero diagonal, then $p - \varepsilon I \in A_1$ for all $\varepsilon > 0$, as seen in § 4, but p is not in A_0 unless $p = 0$.

If the graph $G(p)$ is not connected, one can associate a matrix $\tilde{p}$ to any component of $G(p)$ in an obvious way and construct a sequence $\tilde{p}_n \to \tilde{p}$, $\tilde{p}_n \in A_0$ from the given sequence $p_n \to p$. Hence, without loss of generality, we assume $G(p)$ connected. Under this assumption, Theorem 9 follows from:

LEMMA 3. *Let $P \in V$ and let $G(p)$ be spanned by a connected path γ. Let $A = \min |p_{ij}|$ and $B = \max |p_{ij}|$ over adjacent vertices i, j of γ, and let $C = \max |p_{ii}|$ over all i. Suppose $ap \leqq 0$ where a is a positive diagonal matrix normalized by $a_1 = 1$. Then $\lambda^{-m} \leqq a_i \leqq \lambda^m$ with $\lambda = (2B^2 + 4C^2)/A^2$.*

For proof, let i, j be adjacent vertices of λ. The necessary condition obtained in § 9 can be written

$$r^2(p_{ij})^2 + r^{-2}(p_{ji})^2 \leqq 4p_{ii}p_{jj} - 2p_{ij}p_{ji}$$

where $r^2 = a_i/a_j$. Since i and j are adjacent we have $|p_{ij}| \geqq A$, $|p_{ji}| \geqq A$ and hence both r^2 and $1/r^2$ are bounded by λ. The result follows from the fact that at most m steps of this kind are needed to get from a_1 to any a_i along the path γ.

To deduce Theorem 9 let us notice that the same path γ as that for p will do for p_n if n is large enough, and the ratio λ for p_n is arbitrarily close to that for p when n is large. Hence the corresponding vectors a_n for p_n are bounded by an inequality like that in Lemma 3, as $n \to \infty$, and an easy compactness argument gives Theorem 9.

26. The class *A*. In many respects the most appropriate class for the study of stability is neither A_0 nor A_1, but a class A defined as follows: An *admissible perturbation* of p is a matrix $\tilde{p}$ of the same size such that $\tilde{p}_{ij} = 0$ if, and only if, $p_{ij} = 0$. Then $p \in A$ if every admissible perturbation $\tilde{p}$, which is sufficiently close to p, is in A_0. It is readily checked that $A_0 \supset A \supset A_1$ and that both inclusions are strict. Stability theory based on the class A has a rich and satisfying structure, which sheds much light on the role of the "self-limiting condition" $p_{ii} < 0$ [35], [36], [37].

In general, a set of inequalities which give $p \in A_0$ will give $p \in A$ if those inequalities which involve nonzero elements of p in an essential way are required to be strict. This applies, for example, to the criteria in Theorems 1, 5 and 7. Other conditions of this sort are readily obtained by examining the effect of an admissible perturbation and will not be emphasized here. Our objective in mentioning the class A is to establish the following theorem:

THEOREM 10. *Let A^* be defined as A is defined, except that for A^* the term "admissible perturbation" allows perturbation both of the nonzero coefficients and of those coefficients $p_{ij} = 0$ for which $p_{ji} \neq 0$. Then $A^* = A$, and hence $A \cap V$ is dense in A.*

The concluding statement would tend to justify the restriction $p \in V$ in applied problems in which the coefficients are known only with limited precision.

The proof is complicated by the fact that we must allow an arbitrary perturbation of the nonzero elements, as well as the perturbation of zero elements which is of primary interest. Without loss of generality we assume that the graph $G(p)$ is connected and that $|\cdot|$ denotes the sup norm, $|p| = \max |p_{ij}|$. Throughout the proof, the term "admissible perturbation" is used in the sense of the class A rather than A^*. Since it is trivial that $A \supset A^*$, it suffices to show $A^* \supset A$. To this end, we assume $p \in A$ and we seek to show $p \in A^*$.

We distinguish three cases. If all $p_{ii} = 0$, then $p \in A$ implies $p \in V$ (it implies also that $G(p)$ is a tree) and the problem associated with Theorem 10 does not arise. Suppose next that all $p_{ii} < 0$. Replacing each p_{ii} by $p_{ii} + 2\varepsilon$ is then an admissible perturbation. We take $\varepsilon > 0$ but so small that there exists a multiplier a^* such that $a^* p^* \leqq 0$ for the corresponding matrix $p^* = p + 2\varepsilon I$. This is possible, since $p \in A$, and it gives $a^* p \leqq -2\varepsilon a^*$. Since the left-hand side is a continuous function of the p_{ij}, there exists $\delta > 0$ such that $a^* \tilde{p} \leqq -\varepsilon a^*$ whenever $|p - \tilde{p}| < \delta$. This goes beyond the assertion of Theorem 10 in several respects. First, we are allowed to perturb all the elements of p, and not just those elements p_{ij} for which $p_{ji} \neq 0$. Second, we can choose one and the same multiplier $\tilde{a} = a^*$ for all the perturbations $\tilde{p}$, $|p - \tilde{p}| < \delta$. Third, we got not only $\tilde{a}\tilde{p} \leqq 0$, as required by the definition of A, but even $\tilde{a}\tilde{p} \leqq \eta I$ where $\eta > 0$ is fixed.

The only other case requiring consideration is that in which some p_{ii} are 0 and others are not. By interchanging rows and columns we can assume $p_{ii} < 0$ for $i = 1, 2, \cdots, k$ and $p_{ii} = 0$ for $k < i \leqq m$. Let M denote the set of p_{ij} in the k by k matrix (p_{ij}), $1 \leqq i, j \leqq k$, so that p is decomposed into M and the remaining L-shaped region, L, containing those p_{ij} for which $i > k$ or $j > k$. Some properties of M and L are listed next.

(a) If $p_{ij} \in L$ and $p_{ij} \neq 0$, then $p_{ij} p_{ji} < 0$ and any multiplier a making $ap \leqq 0$ must satisfy $a_i p_{ij} + a_j p_{ji} = 0$ for $i > k$. This follows from the hypothesis $p_{ii} = 0$ for $i > k$.

(b) If all $p_{ij} \in M$ for $i \neq j$ are replaced by 0, this has the effect of removing all links (i, j) from $G(p)$ in which $i \leqq k$ and $j \leqq k$, the end points being, however, retained. The resulting graph must be free of loops and hence is a finite collection of disjoint trees, $T_1, T_2, \cdots, T_c$. This follows because every loop in $G(p)$ must have a strong link [28]; that is, a link $[i, j]$ with $p_{ii} p_{jj} > 0$. Here we use the hypothesis $p \in A$.

(c) Each tree T_i in (b) must have at least one vertex h_i with $h_i \leqq k$, since the original graph $G(p)$ was connected.

(d) Let J be the m by m matrix with diagonal elements $J_{ii} = 1$ for $1 \leqq i \leqq k$ and all other elements 0. Then there exists a positive multiplier a^* such that $a^* p \leqq -3\varepsilon J$ where $\varepsilon > 0$. This follows by making a perturbation of the nonzero elements p_{ii} and using the hypothesis $p \in A$.

After these preliminaries the proof of Theorem 10 is easily completed. Let the nonzero elements $p_{ij} \in L$ be perturbed to $\tilde{p}_{ij}$ so that $|p_{ij} - \tilde{p}_{ij}| < \delta$ where $\delta > 0$ is fixed. Start with $a_h = (a^*)_h$ at the distinguished vertex $h = h_1$ of the tree T_1, and determine a_i over T_1 by use of the relation $a_i \tilde{p}_{ij} + a_j \tilde{p}_{ji} = 0$ for adjacent vertices i, j. None of these vertices is involved in any other tree T_k since the trees are disjoint. Hence we can determine $\tilde{a}_j$ in a similar way over each of the trees, so that the desired relations hold on all trees simultaneously. If $\tilde{p} \neq p$, the value $\tilde{a}$ obtained by this process will not agree with a^*, and the disagreement affects M, in general, as well as L. Nothing can be done about this, since the relation $\tilde{a}_i \tilde{p}_{ij} + \tilde{a}_j \tilde{p}_{ji} = 0$ in L is essential, and must take precedence over any relations associated with M.

Nevertheless, if δ is sufficiently small, it is easily seen that $|a^* - \tilde{a}| < \eta$ in this process, where η is as small as may be desired. Since $a\tilde{p}$ is a continuous function of a, we have $\tilde{a}\tilde{p} \leqq -2\varepsilon J$ if η is sufficiently small, hence also if δ is sufficiently small.

Keeping the same multiplier $\tilde{a}$, we now perturb the elements of M (whether zero or not) to get a new matrix $\tilde{p}$ where again $|p-\tilde{p}|<\delta$. Since $\tilde{a}p$ is a continuous function of p, we shall have $\tilde{a}\tilde{p}\leqq -\varepsilon J$ if δ is sufficiently small, and Theorem 10 follows.

The foregoing proof gives considerable insight into the structure of matrices $p\in A$ in which some, but not all, of the p_{ii} are zero. It also gives the following: Let $p\in A$ and let $|p-\tilde{p}|<\varepsilon$ where $\tilde{p}$ is an admissible perturbation in the extended sense associated with A^*. Let the multiplier $\tilde{a}$ giving $\tilde{a}\tilde{p}\leqq 0$ be normalized by requiring that $\min \tilde{a}_i = 1$, and that $\max \tilde{a}_i$ be as small as possible. Then if ε is sufficiently small, the class of normalized multipliers so obtained is uniformly bounded independently of $\tilde{p}$. In other words, if the multipliers are chosen judiciously, a result like Lemma 3 holds when the explicit condition there given is replaced by the more subtle hypothesis $p\in A$.

REFERENCES

[1] G. P. BARKER, A. BERMAN AND R. J. PLEMMONS, *Positive diagonal solutions to the Lyapunov equations*, Linear and Multilinear Algebra, 5 (1978), pp. 249-256.

[2] EDWIN F. BECKENBACH AND RICHARD BELLMAN, *Inequalities*, Springer-Verlag, New York, 1961, pp. 58, 59.

[3] ABRAHAM BERMAN AND DANIEL HERSCHKOWITZ, *Matrix diagonal stability and its implications*, SIAM J. Alg. Disc. Meth., 4 (1983), pp. 377-381.

[4] A. BERMAN, R. S. VARGA AND R. C. WARD, *Matrices with nonpositive off-diagonal entries*, Lin. Alg. Appl., 21 (1978), pp. 163-174.

[5] ABRAHAM BERMAN AND ROBERT C. WARD, *Classes of stable and semipositive matrices*, Lin. Alg. Appl., 21 (1978), pp. 163-174.

[6] S. BIALAS AND J. GARLOFF, *Intervals of P-matrices and related matrices*, 1983, Lin. Alg. Appl., to appear.

[7] G. W. CROSS, *Three types of matrix stability*, Lin. Alg. Appl., 20 (1978), pp. 253-263.

[8] J. M. CUSHING, *Stable limit cycles of time dependent multispecies interactions*, Math. Biosci., 31 (1976), pp. 259-273.

[9] MIROSLAV FIEDLER AND VLASTIMIL PTAK, *On matrices with non-positive off-diagonal elements and positive principal minors*, Czech. Mat. Zh., 12 (1962), pp. 382-400.

[10] H. I. FREEDMAN, *Deterministic Mathematical Models in Population Ecology*, Marcel Dekker, New York, 1980.

[11] F. R. GANTMACHER, *The Theory of Matrices, Vol.* II, Chelsea, New York, 1959, pp. 70-74, 189.

[12] B. S. GOH, *Global stability in two species interactions*, J. Math. Biol., 3 (1976), pp. 313-318.

[13] ———, *Global stability in many species systems*, Amer. Naturalist, 111 (1977), pp. 135-143.

[14] K. P. HADELER, *On copositive matrices*, Lin. Alg. Appl., 49 (1983), pp. 79-89.

[15] M. IKEDA AND D. D. SILJAK, *Lotka-Volterra equations: Decomposition, stability and structure, Part* II: *Nonequilibrium analysis*, preprint.

[16] CLARK JEFFRIES, VICTOR KLEE AND PAULINE VAN DEN DRIESSCHE, *When is a matrix sign stable?*, Canad. J. Math., XXIX (1977), pp. 315-326.

[17] C. R. JOHNSON, *Sufficient conditions for D-stability*, J. Econ. Theory, 9 (1974), pp. 53-62.

[18] ———, *A local Lyapunov theorem and the stability of sums*, Lin. Alg. Appl., 13 (1976), pp. 37-43.

[19] HASSAN K. KHALIL, *The existence of positive diagonal P such that* $PA+A^TP<0$, IEEE Trans. Automat. Control, AC-27 (1982), pp. 181-184.

[20] N. KRIKORIAN, *The Volterra model for three species predator-prey systems: Boundedness and stability*, J. Math. Biol., 7 (1979), pp. 117-132.

[21] J. LASALLE, *Some extensions of Liapunov's second method*, IRE Trans. Circuit Theory, CT-7 (1960), pp. 520-527.

[22] G. S. LADDE, *Competitive processes and comparison differential systems*, Trans. Amer. Math. Soc., 221 (1976), pp. 391-402.

[23] ———, *Stability of model ecosystems with time delay*, J. Theoret. Biol., 61 (1976), pp. 1-13.

[24] ———, *Competitive processes*, I. *Stability of hereditary systems*, Nonlinear Anal., Theory, Methods and Appl., 1 (1977), pp. 607-631.

[25] ———, *Competitive processes*, II. *Stability of random systems*, J. Theor. Biol., 68 (1977), pp. 331-354.

[26] G. S. LADDE AND D. D. SILJAK, *Stability of multispecies communities in randomly varying environment*, J. Math. Biol., 2 (1975), pp. 165-178.

[27] J. S. MAYBEE, *Combinatorially symmetric matrices*, Lin. Alg. Appls., 8 (1974), pp. 529-537.

[28] J. S. MAYBEE AND J. QUIRK, *Qualitative problems in matrix theory*, SIAM Rev., 2 (1969), pp. 32-51.
[29] P. DE MOTTONI AND A. SCHIAFFINO, *Competition systems with periodic coefficients: A geometric approach*, J. Math. Biol., 11 (1981), pp. 319-335.
[30] A. OSTROWSKI AND O. TAUSSKY, *On the variation of the determinant of a positive definite matrix*, Neder. Akad. Wet. Proc., (A) 54 (1951), pp. 383-385.
[31] S. Y. AND J. W. T. YOUNGS, *The symmetrization of matrices by diagonal matrices*, J. Math. Anal. Appl., 4 (1962), pp. 102-110.
[32] J. PLEMMONS, *M-matrix characterizations*, I. *Nonsingular M-matrices*, Lin. Alg. Appl., 18 (1977), pp. 175-188.
[33] PAUL POLANSKY, *Invariant distributions for multi-population models in random environments*, Theor. Pop. Biol., 16 (1979), pp. 25-34.
[34] J. QUIRK AND R. RUPPERT, *Qualitative economics and the stability of equilibrium*, Rev. Econ. Stud., 32 (1965), pp. 311-325.
[35] RAY REDHEFFER AND WOLFGANG WALTER, *Solution of the stability problem for a class of generalized Volterra prey-predator systems*, J. Differential Equations, 52 (1984), pp. 245-263.
[36] ———, *On parabolic systems of the Volterra prey-predator type*, Nonlinear Anal., Theory, Methods and Appls., 7 (1983), pp. 333-347.
[37] RAY REDHEFFER AND ZHIMING ZHOU, *Global asymptotic stability for a class of many-variable Volterra prey-predator systems*, Nonlin. Anal., Theory, Methods and Appls., 5 (1981), pp. 1309-1329.
[38] ———, *A class of matrices connected with Volterra prey-predator equations*, this Journal, 3 (1982), pp. 122-134.
[39] V. VOLTERRA, *Leçons sur la théorie mathématique de la lutte pour la vie*, Gauthier-Villars et Cie., Paris, 1931.
[40] RICHARD R. VANCE, *Interspecies competition and the intermediate disturbance principle*, preprint.
[41] ANGELIKA WÖRZ-BUSEKROS, *Global stability in ecological systems with continuous time delay*, SIAM J. Appl. Math., 35 (1978), pp. 123-134.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 4, October 1985

008

VOLTERRA MULTIPLIERS II*

RAY REDHEFFER†

Abstract. If p is a real m by m matrix, a Volterra multiplier is a positive diagonal matrix a such that, in this sense of quadratic forms, $ap \leqq 0$. The usefulness of this condition has been well-known since it was introduced by Volterra around 1930, but the usefulness is diminished by the difficulty of deciding whether the multiplier exists. In 1978 the case $m=3$ was fully solved by Cross, but the case $m \geqq 4$ has remained open except under simplifying assumptions; e.g., that the matrix p is in some suitable graph-theoretic sense sparse. Results under such assumptions were given in Part I (SIAM J. Alg. Disc. Meth., 6 (1985), pp. 570–589. of this study, m being arbitrary. Here we present a general theorem, without supplementary hypotheses, which reduces the problem for m to two simultaneous problems for $m-1$, in the spirit of the 1978 work cited above. As a consequence we are able to give a complete theoretical solution when $m=4$ and an effective computational procedure for larger values. In this sense, the Volterra problem can be regarded as solved. The procedure is illustrated by examples which are accessible to previous methods only with difficulty, if at all.

AMS(MOS) subject classifications. 15A45, 15A48

1. Introduction. As in Part I (this issue, pp. 570–589), p denotes a real m by m matrix, a a positive m by m diagonal matrix, and inequalities such as $ap<0$ are interpreted in the sense of quadratic forms. It is said that $p \in A_0$ or A_1 if the Volterra multiplier a can be chosen so that $ap \leqq 0$ or $ap<0$, respectively. The class P denotes the class of matrices p for which all 2^m-1 principal minors are positive and P_1 denotes the class for which the leading principal minors are positive. We have $p+p^t \in P$ if, and only if, $p+p^t \in P_1$. When either condition holds, it is said that $p+p^t$ satisfies the *Hurwitz criterion.* The determinant of a square matrix M is here denoted by $|M|$. This notation was avoided in Part I, because several of the results involved absolute values.

The history and importance of the classes A_0 and A_1 were set forth in Part I together with a number of criteria for the existence of the Volterra multiplier. In general, these criteria were of two types. Either the conditions were necessary and sufficient, but were restricted to special matrices p; or the conditions were merely sufficient, but applied to general matrices. Our object here is to give necessary and sufficient conditions for the general case. Aside from an elementary remark of Goh [3] to the effect that $p \in A_1$ if, and only if, $-p \in P$ in the case $m=2$, the only such conditions available up to now are given by the theorem of Cross [1] for $m=3$. This theorem is as follows, M_{ij} being the minor formed from the ith and jth rows and columns:

THEOREM OF CROSS. *The real* 3 *by* 3 *matrix* p *satisfies* $-p \in A_1$ *if, and only if,* $p \in P$ *and the inequalities*

$$(p_{13}y+p_{31})^2 < 4p_{11}p_{33}y, \qquad (b_1y+b_2)^2 < 4M_{12}M_{23}y, \tag{1}$$

where $b_1 = p_{12}p_{23}-p_{22}p_{13}$ *and* $b_2 = p_{21}p_{32}-p_{22}p_{31}$, *are satisfied simultaneously.*

The ingenious proof [1] depends on a direct, *ad hoc* verification of a certain determinantal identity which is far from trivial even when $m=3$, and which gives no clue as to the form (or even the existence) of such an identity in the general case. Here

* Received by the editors August 2, 1983, and in revised form May 15, 1984. This research was supported in part by the National Science Foundation under grant MCS 79-03544.

† Department of Mathematics, University of California, Los Angeles, California 90024.

we give a new proof which involves little computation and applies to matrices of any size. The result is a corresponding extension of the theorem of Cross. Since the latter underlies our investigation, it is re-examined from the point of view of this paper in §§ 3 and 4. The analysis yields an extension to the class A_0 which is not considered in [1] and is here formulated as Theorem 1. Our main goal, however, is the generalization embodied in Theorem 2. Throughout Part II we give conditions for $-p \in A_1$ rather than for $p \in A_1$ as in Part I. The reason is that we shall have to consider matrices of various orders, and the condition $p \in P$ is easier to deal with than the condition $-p \in P$. This reformulation was already used in the above statement of the theorem of Cross.

Before concluding these introductory remarks, we mention an interesting work of Khalil [5] which treats the Volterra problem by an entirely different idea. The gist of his method is to use an iterative procedure to construct the multiplier a. It is shown that the process converges if the multiplier exists but no bound is given for the number of iterations needed. Unfortunately all of Khalil's examples are for the case $m = 2$, in which case the answer can be found by inspection [3]. In fact, in view of the complete solution found for $m = 3$ by Cross, the only interesting case is that in which $m \geqq 4$. Preliminary calculation suggests that Khalil's method can be usefully applied to such matrices only when programmed on a high-speed computer, and comparison with our examples (which were done by hand) would take us too far afield. It is hoped, however, that such a comparison will be made in a sequel to this paper, in which both methods are written up as a program.

2. The use of the inverse. If $p \in A_1$ then, as is well known, p is nonsingular and $p^{-1} \in A_1$. That p is nonsingular follows from the fact that, if $|p| = 0$, then $|ap| = 0$, and hence $apx = 0$ has a nontrivial solution x. Here we want to emphasize, not only that p^{-1} belongs to A_1, but that *the same multiplier a* works for p^{-1} as for p. This is clear from the identity

$$q^t(ap + p^t a)q = aq + q^t a \tag{2}$$

where $q = p^{-1}$. As seen below, the fact that the same multiplier works for q as for p quickly gives the necessity of the relations (1), and the proof goes through in higher dimensions. This is the first use of the inverse.

The second use is to construct an appropriate form of the identity for $m = 3$ mentioned above, when m is unrestricted. Without going into detail, the gist of the matter is discussed next. Let $r = ap + p^t a$, let R be obtained from r by dropping the last row and column, and let $a = x$, a variable. By a simple argument, given below, it is seen that $|r|$ is a quadratic in x with discriminant d, where d is divisible by $|R|$. Naturally, a corresponding statement holds for $\hat{s} = a\hat{q} + \hat{q}^t a$ where $\hat{q} = |p|q$ is the adjoint of p; that is, the quadratic form has a discriminant $\hat{e}$ which is divisible by $|\hat{S}|$. It follows from (2) that

$$p^t(a\hat{q} + \hat{q}^t a)p = |p|(ap + p^t a), \qquad \hat{q}^t(ap + p^t a)\hat{q} = |p|(a\hat{q} + \hat{q}^t a)$$

and hence that $\hat{e} = |p|^{2m-4}d$, $m \geqq 2$. Therefore $|p|^{2m-4}d$ is divisible by both $|R|$ and $|\hat{S}|$. Further analysis shows that it is divisible by the product, so that $|p|^{m-3}d = |R||\hat{S}|J$ where J is a rational function of degree 0. A rather difficult inductive proof gives $J = 1$, so that finally $|p|^{m-3}d = |R||\hat{S}|$. This reduces to the identity of Cross when $m = 3$.

In the first version of this paper the above procedure was carried out in full, giving Lemma 3 below. But a much simpler proof was found later by Professor Robert Steinberg of UCLA, and it is this proof which is presented here. We have sketched

the original argument nevertheless, because it shows where the result came from. The method of Steinberg is not a derivation of the identity, but a verification of it.

We conclude by mentioning a third use of the inverse; namely, it leads to a surprisingly simple proof of the known theorem that $-p \in A_1 \Rightarrow p \in P$. We use mathematical induction, noting that the case $m = 1$ is trivial and that, if a matrix is in A_1, then its principal submatrices are also in A_1 (with an appropriate m).

Suppose, then, that $-p \in A_1$ and that the desired conclusion $-p \in A_1 \Rightarrow p \in P$ is known for matrices of order $\leqq m-1$. The principal minors of p therefore have the correct sign, and it remains only to show $|p| > 0$. This follows, however, from the above relation $\hat{q} = |p|q$. Since $-q \in A_1$, we have $q_{11} > 0$ by the case $m = 1$. Since $\hat{q}_{11}$ is a principal minor of p of order $m-1$, we have $\hat{q}_{11} > 0$ by the induction hypothesis. The relation $\hat{q}_{11} = |p|q_{11}$ now gives $|p| > 0$, completing the proof.

3. The theorem of Cross. With $m = 3$ let $ap > 0$ for some positive diagonal multiplier a and introduce the adjoint

$$\hat{q} = |p|p^{-1} = \begin{pmatrix} p_{22}p_{33} - p_{23}p_{32} & p_{13}p_{32} - p_{12}p_{33} & p_{12}p_{23} - p_{13}p_{22} \\ p_{23}p_{31} - p_{21}p_{33} & p_{11}p_{33} - p_{13}p_{31} & p_{21}p_{13} - p_{11}p_{23} \\ p_{21}p_{32} - p_{22}p_{31} & p_{12}p_{31} - p_{11}p_{32} & p_{11}p_{22} - p_{12}p_{21} \end{pmatrix}. \tag{3}$$

By results of § 2, we have $|p| > 0$ and $a\hat{q} > 0$ with the same a. Let us then apply the Hurwitz criterion to the principal minors, not only of $ap + p^t a$, but also of $a\hat{q} + \hat{q}^t a$. If t denotes a suitable ratio a_i/a_j, different in each case, the result is the three necessary conditions

$$(p_{12} + tp_{21})^2 < 4p_{11}p_{22}t, \qquad (\hat{q}_{12} + t\hat{q}_{21})^2 < 4\hat{q}_{11}\hat{q}_{22}t, \tag{4a}$$

$$(p_{23} + tp_{32})^2 < 4p_{22}p_{33}t, \qquad (\hat{q}_{23} + t\hat{q}_{32})^2 < 4\hat{q}_{22}\hat{q}_{33}t, \tag{4b}$$

$$(p_{31} + tp_{13})^2 < 4p_{33}p_{11}t, \qquad (\hat{q}_{31} + t\hat{q}_{13})^2 < 4\hat{q}_{33}\hat{q}_{11}t. \tag{4c}$$

(The intent is that each pair of inequalities a, b, c must have a simultaneous solution, but the value of t need not be the same from one pair to the next.) Condition (4c) is the same as (1) with $t = y$, and hence (1) is necessary.

If p satisfies the additional condition $p \in P$ (which is necessary in any case), the theorem of Cross shows that any one of the three inequality-pairs (4) is sufficient to ensure $-p \in A_1$. An alternative proof of this sufficiency, involving little calculation, can be obtained by specializing the proof of Theorem 2 to the case $m = 3$.

It is of considerable interest that the extra hypothesis $p \in P$ could be replaced by

$$p_{ii} > 0, \quad q_{ii} > 0, \quad i = 1, 2, 3 \tag{5a}$$

or by

$$p_{ii} > 0, \quad \hat{q}_{ii} > 0, \quad i = 1, 2, 3. \tag{5b}$$

To see this, let us assume, without loss of generality, that (4a) holds. If the first inequality (4a) has a solution, then $\hat{q}_{33} > 0$ and the equation $\hat{q}_{33} = |p|q_{33}$ gives $|p| > 0$. Together with (5a) this in turn gives $p \in P$. Similarly, if the second inequality (4a) has a solution we get the inequality in

$$0 < \hat{q}_{11}\hat{q}_{22} - \hat{q}_{12}\hat{q}_{21} = |p|p_{33}. \tag{6}$$

Hence $|p| > 0$, and $p \in P$ follows from (5b). (The bearing of the well-known equality (6) on this problem was pointed out by Prof. Robert Steinberg.)

In the presence of either condition (5), then, any one of the conditions (4) is *sufficient* for $-p \in A_1$, and all five conditions (4), (5) are *necessary*. Since each condition (4) can be obtained from any one of them by cyclic permutation of the indices 1, 2, 3, and since the relations involve p and p^{-1} symmetrically, the above formulation has the invariance properties which were alluded to in the concluding paragraph of [1].

4. Extension to the class A_0. As stated in Part I, a matrix p is said to be *combinatorically symmetric* if $p_{ij} = 0 \Leftrightarrow p_{ji} = 0$. We shall establish the following.

THEOREM 1. *Let p be combinatorially symmetric, let $m = 3$, and let*

$$p_{ii} \geqq 0, \quad \hat{q}_{ii} \geqq 0, \quad i = 1, 2, 3 \tag{7}$$

where $\hat{q}$ is the matrix on the right of (3). *Let $*$ denote the three inequality-pairs* (4) *with $<$ replaced by $\leqq$ throughout. Then $-p \in A_0$ if one of the pairs $*$ has a simultaneous solution with $t > 0$. Conversely, if $-p \in A_0$, then* (7) *holds and each of the pairs $*$ has a simultaneous solution with $t > 0$.*

It is worth noting that Theorem 1 is false if p is not assumed to be combinatorially symmetric. For example, suppose the sole nonzero element of p is p_{13}. Then all but one of the six relevant inequalities hold for all t, and yet $-p$ is not in A_0. If a pair $*$ holds only for $t = 0$ the conclusion $-p \in A_0$ also does not follow, but in this case p is effectively a 2-by-2 matrix and the problem is trivial.

For proof let $\tilde{p} = p + \varepsilon I$ where $\varepsilon > 0$. Then $-p \in A_0$ implies $-\tilde{p} \in A_1$ and this in turn implies (4) for $\tilde{p}$ as seen above. It remains to show that the values $\tilde{t}$ associated with $\tilde{p}$ do not tend to 0 or ∞ as $\varepsilon \to 0$. To show this, let us fix attention on (4a), which is repeated for the reader's convenience:

$$(p_{12} + tp_{21})^2 \leqq 4p_{11}p_{22}t, \qquad (\hat{q}_{12} + t\hat{q}_{21})^2 \leqq 4\hat{q}_{11}\hat{q}_{22}t. \tag{8}$$

(We have replaced $<$ by $\leqq$ for later use.) If $p_{12} \neq 0$, then also $p_{21} \neq 0$, and hence $\tilde{t}$ is confined to a compact subset of $(0, \infty)$ as $\varepsilon \to 0$. This gives the desired value $t > 0$ for p. The same happens if $\hat{q}_{12} \neq 0$ and $\hat{q}_{21} = 0$, but we can no longer say that one of these inequalities implies the other.

Suppose, however, that $p_{12} = 0$. Then $p_{21} = 0$ and $\hat{q}_{12} = p_{13}p_{32}$, $\hat{q}_{21} = p_{23}p_{31}$. If $\hat{q}_{12} = 0$, then $p_{13} = 0$ or $p_{32} = 0$. Hence, since p is combinatorially symmetric, we have $\hat{q}_{21} = 0$. Thus both inequalities (8) are satisfied by any t, for instance, by $t = 1$. The only remaining possibility is that neither $\hat{q}_{12}$ nor $\hat{q}_{21}$ is 0, and this was dealt with above. Hence $p \in A_0$ implies (8) with $t > 0$. The same method applies to the other two inequalities $*$ and gives the second statement in Theorem 1.

Suppose next that one relation $*$ holds with $t > 0$. Without loss of generality, we assume that the given relation is (8). Replacing p by $\tilde{p}$ has the effect of replacing p_{ii} by $p_{ii} + \varepsilon$, $i = 1, 2, 3$. In view of the hypothesis $p_{ii} \geqq 0$, $\hat{q}_{ii} \geqq 0$ this in turn increases the magnitude of both p_{ii} and $\hat{q}_{ii}$. Thus (8) becomes a strict inequality for $\tilde{p}$, and the theorem of Cross gives $\tilde{p} \in A_1$. That $p \in A_0$ now follows from Theorem 9 of Part I.

5. Statement of the main theorem. With $m \geqq 2$ as in Part I, the principal result of this paper is as follows:

THEOREM 2. *Let p be nonsingular, with inverse $p^{-1} = q$, and let p^*, q^*, a^* denote the $m-1$ by $m-1$ matrices obtained from p, q, a, respectively, when the last row and last column are deleted. Then:*

(i) *If $ap > 0$ we must have $p_{mm} > 0$, $a^*p^* > 0$ and $a^*q^* > 0$.*

(ii) *If $p_{mm} > 0$, $a^*p^* > 0$ and $a^*q^* > 0$ it is possible to choose $a_m > 0$ in such a way that $ap > 0$.*

The theorem reduces the inequality $ap>0$ for matrices of order m to two other inequalities of the same type for matrices of order $m-1$. The case $m=3$ gives the theorem of Cross, as discussed in § 3.

6. The discriminant. Let $r_{ij}=a_ip_{ij}$, $w=p_{mm}$, $x=a_m$, and

$$u=(p_{m1},p_{m2},\cdots,p_{m,m-1}),\qquad v=(r_{1m},r_{2m},\cdots,r_{m-1,m}) \tag{9}$$

$$R=(r_{ij}+r_{ji}),\qquad 1\leqq i,j\leqq m-1. \tag{10}$$

Thus, z and w are scalars, u and v are $m-1$ dimensional row vectors, and the symmetric matrix R agrees with $a^*p^*+(a^*p^*)^t$ in the notation of Theorem 2. Hence

$$D(x)=\begin{vmatrix} R & v^t+xu^t \\ v+xu & 2xw \end{vmatrix}=|ap+p^ta| \tag{11}$$

where $D(x)$ is defined by this equation. Expanding on the last column as a binomial, we get

$$D(x)=\begin{vmatrix} R & v^t \\ v+xu & xw \end{vmatrix}+\begin{vmatrix} R & xu^t \\ v+xu & xw \end{vmatrix}.$$

Further expansion on the last row as a binomial gives

$$D(x)=d_0+2d_1x+d_2x^2 \tag{12}$$

where

$$d_0=\begin{vmatrix} R & v^t \\ v & 0 \end{vmatrix},\quad d_1=\begin{vmatrix} R & u^t \\ v & w \end{vmatrix},\quad d_2=\begin{vmatrix} R & u^t \\ u & 0 \end{vmatrix}. \tag{13}$$

When R is nonsingular we can multiply the top row by vR^{-1} or uR^{-1}, as the case may be, and subtract from the second row. The result is

$$d_0=-|R|vR^{-1}v^t,\quad d_1=|R|(w-vR^{-1}u^t),\quad d_2=-|R|uR^{-1}u^t. \tag{14}$$

If $|R|=0$, we can reduce the first row of R to 0 by elementary operations and a corresponding sequence of column operations reduces the first column to 0. Applying these operations to the determinants (13) we find $d_1^2=d_0d_2$, and from this one can conclude that the discriminant $d_1^2-d_0d_2$ is divisible by $|R|$. However, it was pointed out by Professor Alfred Hales that Sylvester's identity [4] gives a much stronger result, as follows:

LEMMA 1. *If $d=d_1^2-d_0d_2$ is the discriminant of the quadratic $D(x)$, then*

$$d=-|R|\begin{vmatrix} R & v^t & u^t \\ v & 0 & w \\ u & w & 0 \end{vmatrix}.$$

For those unfamiliar with Sylvester's identity a direct proof can be given with ease. Assuming without loss of generality that R is nonsingular, simplify the determinant on the far right by subtracting vR^{-1} or uR^{-1} times the first row from the second and third rows, respectively. The resulting determinant can be expanded by inspection and is seen to agree with $-d/|R|$ as computed from (14).

LEMMA 2. *If $R>0$, then $d_0\leqq 0$ and $d_2\leqq 0$. If, in addition, $d\geqq 0$ and $w>0$, then $d_1>0$.*

The hypothesis $R>0$ implies $|R|>0$. Hence, $d_0\leqq 0$ and $d_2\leqq 0$ follow from (14) and the inequalities are strict unless u is 0. To show that $d_1>0$ holds when, in addition,

$w>0$ and $d\geqq 0$, assume for contradiction that $d_1\leqq 0$. Then

$$-|R|vR^{-1}u^t<d_1\leqq 0 \tag{15}$$

by (6). On the other hand since $R^{-1}>0$ the formula $(u, v)=vR^{-1}u^t$ describes a positive definite inner product and the Cauchy-Schwarz inequality for that product leads to

$$(vR^{-1}u^t)^2\leqq(vR^{-1}v^t)(uR^{-1}u^t). \tag{16}$$

Multiplying by $|R|^2$ and using (7), we get $d_1^2<d_0d_2$, which contradicts the hypothesis $d\geqq 0$.

7. The inverse. If p is nonsingular, with $q=p^{-1}$, then, as stated in § 2, we have $aq>0$ if and only if $ap>0$. Let us now set

$$q=(q_{ij}),\quad s_{ij}=a_iq_{ij},\quad \text{and}\quad S=(s_{ij}+s_{ji}),\quad 1\leqq i,j\leqq m-1,$$

corresponding to the matrix R introduced previously. Lemma 3 enables us to determine the sign of the discriminant d:

LEMMA 3. *If $m\geqq 2$ and p is nonsingular, then $d=|p|^2|R||S|$.*

The following elegant proof, which is much simpler than the proof first obtained, is due to Professor Robert Steinberg. Referring to Lemma 1, let

$$T=\begin{pmatrix} R & v^t & u^t \\ v & 0 & w \\ u & w & 0 \end{pmatrix}.$$

Multiply the last row by a_m and add to the next-to-last row, and then do the same for columns, to get

$$|T|=\begin{vmatrix} ap+p^ta & b^t \\ b & 0 \end{vmatrix} \tag{17}$$

where $b=(u, w)$ is the bottom row of p. If $q=p^{-1}$, a short calculation using block multiplication gives

$$\begin{pmatrix} q^t & 0 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} ap+p^ta & b \\ b & 0 \end{pmatrix}\begin{pmatrix} q & 0 \\ 0 & 1 \end{pmatrix}=\begin{pmatrix} aq+q^ta & q^tb^t \\ bq & 0 \end{pmatrix}. \tag{18}$$

Since $pq=I$, the expression bq must be the bottom row of I, that is,

$$bq=(0, 0, \cdots, 0, 1)\qquad (m \text{ coordinates}).$$

This in turn gives

$$\begin{vmatrix} aq+q^ta & q^tb^t \\ bq & 0 \end{vmatrix}=-|S| \tag{19}$$

as seen when we expand on the bottom row and then expand the result on the right-hand column. Taking determinants in (18) and using (17) and (19), we get

$$|q||T||q|=-|S|$$

or $|T|=-|p|^2|S|$, since $qp=I$. Lemma 3 now follows from Lemma 1.

8. Proof of Theorem 2. If $ap>0$, then $p_{mm}>0$ and $aq>0$ where $q=p^{-1}$. Since the principal minors of a positive matrix are positive, we see that $a^*p^*>0$ and $a^*q^*>0$. This gives (i).

To get (ii) let us define

$$a = \begin{pmatrix} a^* & 0 \\ 0 & x \end{pmatrix}$$

where a^* is the given positive multiplier such that $a^*p^* > 0$, $a^*p^* > 0$, and let us reason as in [1]. The Hurwitz criterion gives $|R| > 0$ and $|S| > 0$ and, since p is nonsingular, $d > 0$ follows from Lemma 3. Lemma 2 shows then that the inequality $D(x) > 0$ has a positive solution x. This condition together with $R > 0$ ensures that $ap + p^t a$ satisfies the Hurwitz criterion, and hence $ap > 0$.

9. The case $m = 3$. When $m = 3$, $q = p^{-1}$ and $a = (1, t, x)$ we have

$$R = \begin{pmatrix} 2p_{11} & p_{12} + tp_{21} \\ p_{12} + tp_{21} & 2tp_{22} \end{pmatrix}, \qquad S = \begin{pmatrix} 2q_{11} & q_{11} + tq_{21} \\ q_{12} + tq_{21} & 2tq_{22} \end{pmatrix}.$$

The conditions $R > 0$ and $S > 0$ hold if, and only if, $p_{11} > 0$, $q_{11} > 0$, and

$$4tp_{11}p_{22} > (p_{12} + tp_{21})^2, \qquad 4tq_{11}q_{22} > (q_{12} + tq_{21})^2. \tag{20}$$

Thus, existence of a simultaneous solution $t > 0$ of (20), together with $p_{11} > 0$, $q_{11} > 0$ and $p_{33} > 0$, is necessary and sufficient for existence of $a > 0$ such that $ap > 0$. This is equivalent to the result of Cross, in which the conditions supplementary to (20) are that p shall satisfy the Hurwitz criterion.

10. The case $m = 4$. If $ap > 0$, then $p_{ii} > 0$ and $q_{ii} > 0$, where $q = p^{-1}$, and this obviously necessary condition is assumed in the sequel. A similar remark applies to all the submatrices introduced in the analysis; if any of them has a nonpositive diagonal element, the investigation stops at that point.

Let $m = 4$ and for any 4 by 4 matrix M let M^* denote the 3 by 3 matrix which is obtained from M when the last row and column are deleted. If $ap > 0$, then p is nonsingular and $aq > 0$ where $q = p^{-1}$. From this follows $a^*p^* > 0$, $a^*(p^*)^{-1} > 0$, $a^*q^* > 0$, and $a^*(q^*)^{-1} > 0$. Let $a = (1, t, s, x)$ and consider the leading 2 by 2 minor of each of these four matrices. Since the minor must be positive definite in each instance, we are led to four quadratic inequalities involving t alone, namely, to (20) and to two others of the same form for $(p^*)^{-1}$ and $(q^*)^{-1}$. Each of these inequalities must define a nonempty open interval of the positive real axis; otherwise the multiplier a does not exist. Furthermore, the four intervals must have a nonempty intersection. If the intervals are (α_i, β_i), this means that $\alpha < \beta$, where $\alpha = \max(\alpha_i)$ and $\beta = \min(\beta_i)$. Finally t must satisfy $\alpha < t < \beta$.

For each $t \in (\alpha, \beta)$ the quadratic inequality

$$d_0 + 2d_1 s + d_2 s^2 > 0 \tag{21}$$

determines a nonempty open interval of the real axis, where d_0, d_1 and d_2 are given by (5) as

$$\begin{vmatrix} 2p_{11} & p_{12} + t_{21} & p_{13} \\ p_{12} + tp_{21} & 2tp_{22} & tp_{23} \\ p_{13} & tp_{23} & 0 \end{vmatrix}, \qquad \begin{vmatrix} 2p_{11} & p_{12} + tp_{21} & p_{31} \\ p_{12} + tp_{21} & 2tp_{22} & p_{32} \\ p_{13} & tp_{23} & p_{33} \end{vmatrix},$$

$$\begin{vmatrix} 2p_{11} & p_{12} + tp_{21} & p_{31} \\ p_{12} + tp_{21} & 2tp_{22} & p_{32} \\ p_{31} & p_{32} & 0 \end{vmatrix}$$

respectively. Likewise, for each $t \in (\alpha, \beta)$ the inequality

$$e_0 + 2e_1 s + e_2 s^2 > 0 \tag{22}$$

determines a nonempty open interval of the real axis, where e_i is obtained from d_i by writing q in place of p in the defining determinants. This follows from Theorem 1 with $m = 3$ or from the theorem of Cross applied to p^* and to q^*, the role of x in the theorem being taken by s.

If $(p^*)^{-1} = (\bar{q}_{ij})$ then, aside from the positive factor $2|p^*|$,

$$\begin{aligned} d_1 &= t^2 p_{23}\bar{q}_{23} + t p_{13}\bar{q}_{13}, \quad d_2 = t p_{31}\bar{q}_{31} + p_{32}\bar{q}_{32}, \\ 2d_1 &= p_{21}\bar{q}_{21}t^2 + (|p^*| + p_{11}\bar{q}_{11} + p_{22}\bar{q}_{22} + p_{33}\bar{q}_{33})t + p_{12}\bar{q}_{12}. \end{aligned} \tag{23}$$

These formulas are useful in numerical work because $(p^*)^{-1} = (\bar{q}_{ij})$ must be found in the course of the computation in any case.

Since s is the same in both (21) and (22), a second necessary condition is that it must be possible to choose $t \in (\alpha, \beta)$ in such a way that (21) and (22) have a simultaneous solution. When this holds, x can be chosen so that $ap > 0$; in other words, the *necessary* conditions enumerated above are also *sufficient.* This last remark, which embodies the main mathematical content of the whole development, is Theorem 2 applied in the case $m = 4$. Note that $R = a^*p^* + (a^*p^*)^t$ and $S = a^*q^* + (a^*q^*)^t$, and that $R > 0$ and $S > 0$ are ensured by choice of t and s.

If every $t \in (\alpha, \beta)$ leads to a simultaneous solution, then we can make any particular choice, say $t = (\alpha + \beta)/2$, and the problem is solved. The only other case in which the multiplier a exists is that in which some values t work, but not all do. In this case as t traverses its interval (α, β) we must reach a point where the two s-intervals are just abutting. The quadratic equations defining the end points have a common root, and hence t at the point in question must satisfy the quartic equation

$$\begin{vmatrix} d_2 & e_2 \\ d_0 & e_0 \end{vmatrix}^2 = 4 \begin{vmatrix} d_2 & e_2 \\ d_1 & e_1 \end{vmatrix} \begin{vmatrix} d_1 & e_1 \\ d_0 & e_0 \end{vmatrix}. \tag{24}$$

Thus we are led to a specific decision procedure. First, find the interval (α, β), verifying that it is nonempty. Second, see if $t = (\alpha + \beta)/2$ allows a simultaneous solution of (21), (22). Third (if this fails) see if the quartic (24) has a root in (α, β). If the latter condition fails too, then the multiplier does not exist. If it succeeds, then (except in infinitely rare cases) the multiplier does exist. The only reservation is that the s-intervals could just barely abut, as t traverses its interval, and then move apart again without ever overlapping. This entails an algebraic relation among the elements p_{ij} which is not likely in any practical application. To rule it out, however, one would have to try values t near the root of the quartic and see whether the intervals overlap.

11. Remarks on the general case. An important difference between the cases $m = 3$ and $m = 4$ is that in the former case there is no free parameter in the inequalities leading to the multiplier a, while in the latter there is the free parameter $t = a_2$, the value a_1 being normalized as 1. This difficulty was overcome, in part, by the considerations leading to (24). However, for larger values of m the presence of free parameters becomes increasingly burdensome and an alternative, less precise procedure is suggested here.

Namely, instead of trying to optimize the choice of t in $a = (1, t, s, \cdots)$ just take $t = (\alpha + \beta)/2$, where (α, β) is the defining interval; if this interval is empty, the multiplier

a does not exist. Theorem 1 then reduces the problem to two others, with no free parameter, for matrices of order $m-1$. We take s to be at the mid-point of the defining s-interval given by these two problems, and so on. If the method does not abort, it will produce a multiplier a, and if no multiplier exists, the method will necessarily abort. It is not infallible, but is shown by examples to be surprisingly effective.

The reason for this effectiveness is, in part, the large number of independent inequalities for each variable, and in part the fact that these inequalities take account of the full structure of p. In general there are 2^{m-2} inequalities for the first variable, t, half as many for s, and so on, down to a single inequality (which need not be written down) for the last variable $x = a_n$. This is seen from the structure of Theorem 2, which reduces the problem with m to two similar problems with $m-1$.

Another aspect of the general case is that one must make far fewer matrix inversions than one might think from the number of inequalities. The reason is that all principal subdeterminants encountered must be positive; otherwise the procedure comes to a halt. Thus, if $q = p^{-1}$ is computed by the usual method, operating simultaneously on p and on I, the diagonalization of p can be achieved by row operations alone, and furthermore, it will not be necessary to interchange any two rows. The calculation can be arranged, therefore, so that it automatically inverts the leading submatrices of orders $2, 3, \cdots, m-1$, as well as p itself. The same applies to other inversions, e.g., to the computation of $(p^*)^{-1}$ and $(q^*)^{-1}$.

These general remarks are now illustrated in a specific example.

12. A numerical example. Let us consider the matrix

$$p = \begin{pmatrix} 2 & 3 & 2 & 1 \\ 1 & 2 & 1 & 3 \\ 1 & 2 & 2 & 3 \\ -1 & 1 & 1 & c \end{pmatrix}.$$

The problem is: For what values of the constant c is there a multiplier a such that $ap > 0$? It will be seen that no such multiplier exists when $c \leqq 12$, and that the multiplier always exists if $c \geqq 12.6$. It should be noticed that this matrix does not have favorable properties of the kind usually assumed in such problems. There is no condition of diagonal dominance, the off-diagonal elements are not of one sign, nor do they satisfy the Volterra condition $p_{ij}p_{ji} < 0$. The graph has no special simplicity, being the complete graph on four vertices. Furthermore, the multiplier $a = I$ does not work for any c, since $|p + p^t| = -112$ independently of c.

If p is followed by the 4-by-4 identity matrix I and the diagonalization is begun by use of the first three rows to simplify the first three columns only, the result is

$$\begin{matrix} 1 & 0 & 0 & -7 & 2 & -2 & -1 & 0 \\ 0 & 1 & 0 & 5 & -1 & 2 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & b & 3 & -3 & -2 & 1 \end{matrix} \tag{25}$$

where $b = c - 12$. It was not necessary to divide any row by a constant, because p was chosen so that the leading 2 by 2 and 3 by 3 minors are 1. The left-hand matrix (25) shows that $|p| = b$ and hence $b > 0$ is necessary for existance of the multiplier a. This gives $c > 12$. The leading 3 by 3 submatrix on the right gives the inverse of the corresponding submatrix in p; that is, it gives $(p^*)^{-1}$.

For simplicity we now take $b=1$, corresponding to $c=13$. Continuing the diagonalization in (17) gives I on the left, $q=p^{-1}$ on the right, and

$$q^*=\begin{pmatrix} 23 & -23 & -15 \\ -16 & 17 & 10 \\ 0 & -1 & 1 \end{pmatrix}, \qquad 13(q^*)^{-1}=\begin{pmatrix} 27 & 38 & 25 \\ 16 & 23 & 10 \\ 16 & 23 & 23 \end{pmatrix}. \tag{26}$$

Here the second result is obtained from the first by determinants or by diagonalization, as preferred. Together with the two matrices (26) we consider

$$p^*=\begin{pmatrix} 2 & 3 & 2 \\ 1 & 2 & 1 \\ 1 & 2 & 2 \end{pmatrix}, \qquad (p^*)^{-1}=\begin{pmatrix} 2 & -2 & -1 \\ -1 & 2 & 0 \\ 0 & -1 & 1 \end{pmatrix}, \tag{27}$$

the latter being obtained from (25). By Theorem 1, we can find a so that $ap>0$ if, and only if, each of the four matrices (26), (27) satisfies the corresponding condition with a^*.

Let us set $a^*=\operatorname{diag}(1, t, s)$. The leading 2-by-2 minors in (27) and (26) give, respectively,

$$(3+t)^2<16t, \quad (2+t)^2<16t, \quad (23+16t)^2<1564t, \quad (38+16t)^2<2484t.$$

The corresponding t-intervals are

$$(1,9), \quad (0.34, 11.66), \quad (0.88, 2.36), \quad (1.77, 3.18).$$

Hence, $1.77<t<2.36$ is a necessary and sufficient condition for all four inequalities to be satisfied.

Instead of carrying t as a parameter, let us now take $t=2$. By (23) applied first to (27) and then to (26) the inequalities for s are

$$s^2-6s+2<0, \qquad 23s^2-198s+350<0$$

with corresponding intervals

$$(0.35, 5.65), \qquad (2.59, 6.12).$$

Since these have a nonempty intersection, the multiplier a can be found. In fact, one can take $a^*=\operatorname{diag}(1,2,3)$, $(1,2,4)$, or $(1,2,5)$, if integral components are desired. That $a_4=x$ can be suitably chosen follows from Theorem 2.

As a check, one can readily verify, by the Hurwitz criterion, that $a^*q^*>0$ and $a^*p^*>0$ in (26), (27); this verification is much easier than is determination of the unknown multiplier a^*. A more elaborate verification allows us to narrow down the bounds on c. Let $a=(1,2,4,x)$ and compute $ap+p^ta$, keeping c arbitrary. It is seen that the Hurwitz criterion holds if, and only if, $|ap+p^ta|>0$, and this in turn is equivalent to

$$-79x^2+2x(24c-122)-412>0. \tag{28}$$

The discriminant is positive for $c>12.58$. Hence the multiplier a can be found if $c\geqq 12.6$. As a further check note that, when $c=13$, the value $|S|=|a^*q^*+q^{*t}a^*|$ can be read off from (26) and the discriminant is

$$d=(190)^2-(412)(79)=3552=(1)^2(24)(148)=|p|^2|R||S|$$

in agreement with Lemma 3. It may be mentioned incidentally that whenever x exists, we can take $2x=x_1+x_2$, where x_1 and x_2 are the roots of the corresponding quadratic

equation. This gives $x = -d_1/d_2$ in the notation of § 3; hence

$$x = \frac{24c - 122}{79}$$

in the present case. When $c = 12.6$, this gives $x = 2.28$, so that a possible multiplier is

$$a = \text{diag}\,(1, 2, 4, 2.28).$$

13. Semidefinite matrices. Much of the utility of Volterra multipliers stems from the fact that one can have $ap \geqq 0$ in realistic cases even when some $p_{ii} = 0$. In the context of differential equations, this means that the self-limiting term is absent from the ith equation. For such indices the nonzero off-diagonal elements must satisfy the Volterra condition $p_{ij}p_{ji} < 0$ and the ratio a_i/a_j is uniquely determined by $a_i p_{ij} + a_j p_{ji} = 0$. Nevertheless, it is often possible to allow small perturbation of the nonzero elements, so that $-p \in A$ in the notation [6]. Indeed, the presence of elements $p_{ii} = 0$ is rather a help than a hindrance, because it allows us to determine some of the ratios a_i/a_k a priori. For systematic discussion one can interchange rows and columns, so that all zero p_{ii} are at the lower end of the diagonal.

As an illustration let us consider the matrix

$$\begin{pmatrix} 2 & 3 & 2 & 1 & z_1 \\ 1 & 2 & 1 & 3 & z_2 \\ 1 & 2 & 2 & 3 & z_3 \\ -1 & 1 & 1 & c & z_4 \\ y_1 & y_2 & y_3 & y_4 & 0 \end{pmatrix}$$

with the multiplier $a = \text{diag}\,(1, t, s, x, y)$. A necessary condition is $y_i z_i \leqq 0$ and

$$yy_1 + z_1 = 0, \quad yy_2 + tz_2 = 0, \quad yy_3 + sz_3 = 0, \quad yy_4 + xz_4 = 0.$$

Another necessary condition is that the values $1, t, s, x$ shall yield a multiplier for the matrix considered in the preceding section. These two conditions together are sufficient.

For example let $y_1 z_1 < 0$ and $y_2 z_2 < 0$, the remaining variables being 0. Then $y = -z_1/y_1$ and $t = y_2 z_1/(y_1 z_2)$. It is necessary that this t shall satisfy the inequality obtained in § 12,

$$1.77 < \frac{y_2 z_1}{y_1 z_2} < 2.36.$$

If the value $t = y_2 z_1/(y_1 z_2)$ leads to a value of s in the remainder of the discussion, then a exists to make $ap \geqq 0$, and otherwise not. In particular, when $c = 13$ the condition

$$y_2 z_1 = 2 y_1 z_2 \tag{29}$$

is sufficient, since it was verified that the choice $t = 2$ is satisfactory. The new feature is that the extra coefficient y_i, z_i, with $p_{55} = 0$, tend to *take up the slack*; the choice of t is no longer subject to discretion.

Pursuing this example further, let us suppose that (29) holds and that $y_3 z_3 < 0$, the condition $y_4 = z_4 = 0$ being retained. Then the value $s = -yy_3/z_3$ must be compatible with the analysis of § 9, and we get the necessary and sufficient condition

$$2.59 < \frac{z_1 y_3}{y_1 z_3} < 5.65$$

when $c = 13$. Under the hypothesis $c = 13$ and (29) if the above double inequality holds, then a can be found, and otherwise not.

Continuing the discussion, let $y_4 z_4 < 0$ and let

$$z_1 y_3 = 4 y_1 z_3 \tag{30}$$

as well as (29), so that $t = 2$ and $s = 4$. Then we have the situation leading to (28). Since $x = -yy_4/z_4$, a necessary and sufficient condition is

$$x_1 < \frac{z_1 y_4}{y_1 z_4} < x_2$$

where x_1 and x_2 are the zeros of the quadratic (28). If $C = 24c - 122$, the bounds for $C \to \infty$ are, asymptotically,

$$\frac{206}{C} < \frac{z_1 y_4}{y_1 z_4} < \frac{2C}{79} - \frac{206}{C}.$$

For example, when $c = 20$, the exact and approximate intervals are, respectively, (0.62, 8.45) and (0.58, 8.48). Since the multiplier $a = (1, t, s, x)$ has a certain leeway in the analysis of § 9, the above conditions give $-p \in A$ and not only $-p \in A_0$, in the notation [6]. This is true even though (29) and (30) are exact equalities. A slight deviation in the y_i and z_i can be compensated by adjusting the multiplier a.

REFERENCES

[1] G. W. Cross, *Three types of matrix stability*, Lin. Alg. Appl., 20 (1978), pp. 253-263.

[2] F. R. Gantmacher, *The Theory of Matrices, Vol.* I, Chelsea, New York, 1959, p. 32.

[3] B. S. Goh, *Global stability in many species systems*, Amer. Naturalist, 111 (1977), pp. 135-143.

[4] Thomas Muir, *A Treatise of the Theory of Determinants*, Dover, New York, 1960, p. 166.

[5] Hassan K. Khalil, *The existence of positive diagonal P such that* $PA + A^T P < 0$, IEEE Trans. Automat. Control AC-27 (1982), pp. 181-184.

[6] Ray Redheffer, *Volterra multipliers* I, this Journal, this issue, pp. 570-589. Additional references are given there.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 4, October 1985

009

THE ASYMPTOTIC BEHAVIOR OF TOEPLITZ DETERMINANTS GENERATED BY THE LAURENT COEFFICIENTS OF A MEROMORPHIC FUNCTION*

ADHEMAR BULTHEEL†

Abstract. By combining known results on the asymptotic behavior of Hankel and Toeplitz determinants we obtain expressions for $\det (f_{m+i-j})_{i,j=0}^{n}$ as $n \to \infty$ and $m = 0, \pm 1, \pm 2, \cdots$. The f_k are Laurent coefficients of a meromorphic function. These expressions contain the zeros of the meromorphic function and are therefore interesting in proving results on the convergence properties of Laurent-Padé approximants.

Key word. Toeplitz determinants

AMS(MOS) subject classifications. 47B35, 41A60

1. Introduction. The *qd*-algorithm of Rutishauser [11] or the $\pi\zeta$-algorithm [5] are known to construct a table of numbers, given the Taylor series of a meromorphic function. By considering the column limits of these tables, we can compute the poles of that function. The main tool in the proof of these results is the study of the asymptotic behavior of Hankel determinants constructed with the Taylor series coefficients. By using a symmetry property of the tables constructed by the above algorithms for a Taylor series and its inverse one finds that row limits of these tables allow zeros of that function to be computed. If the meromorphic function is given by a Laurent series in a certain annular region, centered around the origin, it is possible to construct by the same algorithms a bi-infinite table of numbers [6, p. 62]. As in the case of a Taylor series one can prove that the downward limits can be used to compute the poles of the function around infinity. The upward limits of the columns can be used to find the poles around the origin. In [8] this was shown in the context of two point Padé approximants for an algorithm developed by McCabe and Murphy [10]. This algorithm is of the same type and closely related to the *qd*- and $\pi\zeta$-algorithms. We call it the FG-algorithm, referring to the notation used in [9]. The method of proof is based on an additive splitting of the Laurent series,

$$f(z) = \sum_{-\infty}^{\infty} f_k z^k = \sum_{0}^{\infty} f_k z^k + \sum_{1}^{\infty} f_{-k} z^{-k}.$$

The downward column limits depend only on the coefficients in the first sum. The upward column limits depend only on the coefficients in the second sum. Consequently the theory developed for a Taylor series can again be used. It is to be expected that the row limits will give information about the zeros. This is indeed the case [2], [3]. These row limits depend, however, on the complete Laurent series and there is no obvious symmetry property that can derive the results from the previous theory. It turns out that instead of the asymptotic behavior of the Hankel determinant $\det [f_{m+i+j}]_{i,j=0}^{n}$ as $m \to \infty$ we now have to investigate its behavior as $n \to \infty$, or what is the same, the behavior of Toeplitz determinants $\det [f_{m+i-j}]_{i,j=0}^{n}$ as $n \to \infty$. The latter is the subject of this paper. It will be clear at the end that the theory for a Taylor series can still be used but now for a multiplicative splitting of f: $f(z) = f_+(z) f_-(z)$, where

* Received by the editors January 11, 1984, and in final revised form June 5, 1984.

† Department of Computer Science, Katholieke Universiteit Leuven, Celestijnenlaan 200A, B-3030 Leuven, Belgium.

f_+ is analytic around the origin and $f_-(z)$ is analytic around infinity. Several expressions for the asymptotics of Toeplitz determinants exist in the literature. Usually they are constructed from the Fourier coefficients of a function that has certain properties. We shall use these and adapt them to our situation. In view of the application it is important that the zeros of the function are introduced in the expressions. It should be noted that for rational functions and under certain restricting conditions an explicit expression for Toeplitz determinants was given in [4].

We do not include the proofs on the row limits of the tables constructed by the rhombus rules because these were given elsewhere and because we think the expressions may be interesting in their own right.

2. Notation and main results. Let $f(z)$ be meromorphic in $\mathbb{C}_0 = \mathbb{C}\backslash\{0\}$ and suppose there exist r and R such that

$$f(z) = \sum_{-\infty}^{\infty} f_k z^k \tag{2.1}$$

is a Laurent series of $f(z)$, convergent in the annulus $r<|z|<R$.

We suppose further that

$$f(z) = G_\rho(\tilde{f}) f_+(z) f_-(z) z^\kappa \tag{2.2}$$

with

$$\tilde{f}(z) = z^{-\kappa} f(z)$$

and

$$G_\rho(\tilde{f}) = \exp\left[\frac{1}{2\pi}\int_0^{2\pi} \log \tilde{f}(\rho e^{i\theta})\, d\theta\right], \qquad r<\rho<R, \tag{2.3}$$

and

$$f_+(z) = \prod_{k\in Z^+}\left(1-\frac{z}{\zeta_k^+}\right)\Big/\prod_{k\in P^+}\left(1-\frac{z}{\pi_k^+}\right), \tag{2.4}$$

$$f_-(z) = \prod_{k\in Z^-}\left(1-\frac{\zeta_k^-}{z}\right)\Big/\prod_{\pi\in P^-}\left(1-\frac{\pi_k^-}{z}\right), \tag{2.5}$$

where the zeros and poles are numbered such that

$$\begin{aligned} &\cdots \leqq |\zeta_2^-| \leqq |\zeta_1^-| \leqq r < R \leqq |\zeta_1^+| \leqq |\zeta_2^+| \leqq \cdots, \\ &\cdots \leqq |\pi_2^-| \leqq |\pi_1^-| \leqq r < R \leqq |\pi_1^+| \leqq |\pi_2^+| \leqq \cdots. \end{aligned} \tag{2.6}$$

We suppose that ζ_k^+, $k\in Z^+$, are all finite and we set $\zeta_k^+ = \infty$ if $k > \# Z^+$ and similarly for π_k^+. Also $\zeta_k^- \neq 0$ for $k\in Z^-$ and $\zeta_k^- = 0$ for $k > \# Z^-$ and similarly for π_k^-.

f_+ and f_- are supposed to be irreducible. κ of (2.2) is the winding number

$$\kappa = \kappa_\rho = \operatorname{ind}_\rho(f) = \frac{1}{2\pi}[\arg f(\rho e^{i\theta})]_0^{2\pi}. \tag{2.7}$$

Note that $\operatorname{ind}_\rho(f_+) = \operatorname{ind}_\rho(f_-) = \operatorname{ind}_\rho(\tilde{f}) = 0$. For a rational function f, $\operatorname{ind}_\rho(f)$ = the number of zeros in $|z|<\rho$ − the number of poles in $|z|<\rho$. With the series (2.1) we associate the Toeplitz determinants

$$T_n^{(m)}(f) = \det(f_{m+i-j})_{i,j=0}^n, \qquad m, n = 0, \pm 1, \pm 2, \cdots.$$

Then for $n \to \infty$, we have the following asymptotic expansions:

THEOREM 1. *With the notation just introduced we have for* $n \to \infty$:

$$T_n^{(m)}(f) = c[G_\rho(\tilde{f})]^{n+1}\{1 + o(1)\} \quad \text{if} \quad m = \kappa,$$

$$T_n^{(m)}(f) = c[G_\rho(\tilde{f})]^{n+1}[(-\zeta_1^+)(-\zeta_2^+)\cdots(-\zeta_{m-\kappa}^+)]^{-n} \cdot \{1 + O(|\zeta_{m-\kappa}^+/\sigma|^n)\}$$

if $0 < m - \kappa < \#Z^+ + 1$ *and if* σ *satisfying* $|\zeta_{m-\kappa}^+| < \sigma < |\zeta_{m-\kappa+1}^+|$ *exists.*

$$T_n^{(m)}(f) = c[G_\rho(\tilde{f})]^{n+1}[(-\zeta_1^-)(-\zeta_2^-)\cdots(-\zeta_{\kappa-m}^-)]^n \cdot \{1 + O(|\sigma_{\kappa-m}^-|^n)\}$$

if $0 < \kappa - m < \#Z^- + 1$ *and if* σ *satisfying* $|\zeta_{\kappa-m+1}^-| < \sigma < |\zeta_{\kappa-m}^-|$ *exists.*

Here and in the following c is a constant not depending on n. It is seen from this that the asymptotic behavior of $T_n^{(m)}(f)$ is up to a factor completely defined by f_+ or f_-. We have thus the following theorem.

THEOREM 2.

$$T_n^{(m)}(f) \sim T_n^{(m-\kappa)}(f_+)[G_\rho(\tilde{f})]^{n+1}$$

where $m \geqq \kappa$, $\tilde{f}(z) = z^{-\kappa}f(z)$, *and*

$$T_n^{(m)}(f) \sim T_n^{(\kappa-m)}(\hat{f}_-)[G_\rho(\tilde{f})]^{n+1}$$

where $m \leqq \kappa$, $\hat{f}_-(z) = f_-(1/z)$, $\tilde{f}(z) = z^{-\kappa}f(z)$.

As a special case we reformulate Theorem 1 for a rational function $f(z)$.

THEOREM 3. *Suppose* $f(z)$ *is a rational function with irreducible form*

$$f(z) = K \cdot z^l \cdot \frac{(z-\zeta_1)(z-\zeta_2)\cdots(z-\zeta_N)}{(z-\pi_1)(z-\pi_2)\cdots(z-\pi_M)}.$$

Let $f(z) = \sum_{-\infty}^{\infty} f_k z^k$ *in* $r < |z| < R$ *and* $f(z) \neq 0$ *on* $|z| = \rho$ *with* $r < \rho < R$. *We order poles and zeros as follows*

$$0 < |\zeta_N| \leqq |\zeta_{N-1}| \leqq \cdots \leqq |\zeta_1| < \infty,$$

$$0 < |\pi_M| \leqq |\pi_{M-1}| \leqq \cdots \leqq |\pi_{P+1}| < \rho < |\pi_P| \leqq |\pi_{P-1}| \leqq \cdots \leqq |\pi_1| < \infty.$$

$M - P$ *is the number of nonzero poles in* $|z| < \rho$. *Then for* $0 \leqq k - m \leqq N$, *where* $k = N - M + P + l$, *we have*

$$T_n^{(m)}(f) = cK^n[(-\zeta_1)(-\zeta_2)\cdots(-\zeta_{k-m})]^n[(-\pi_1)(-\pi_2)\cdots(-\pi_P)]^{-n} \cdot \left\{1 + O\left(\left|\frac{\zeta_{k-m+1}}{\sigma}\right|^n\right)\right\}, \qquad n \to \infty,$$

where it is understood that ζ_0 *and* ζ_{N+1} *are arbitrary and defined by*

$$0 < \zeta_{N+1} < |\zeta_N| \quad \text{and} \quad |\zeta_1| < \zeta_0 < \infty$$

and on condition σ *exists satisfying* $|\zeta_{k-m+1}| < \sigma < |\zeta_{k-m}|$.

We prove Theorem 1 by taking results on the asymptotic behavior of Toeplitz determinants from the literature. By combining them and using some elementary transformations we get the desired results.

3. Proof of Theorem 1. We shall start by eliminating the geometric mean $G_\rho(\tilde{f})$ from the formula. To do this use the simple observation given in

LEMMA 1.

$$T_n^{(m)}(c\cdot f) = c^{n+1}T_n^{(m)}(f).$$

This allows us to set for the moment $G_\rho(\tilde{f})=1$ without loss of generality. The next step is to solve the problem for $\kappa=0$ and for $\rho=1$, so that the f_k may be considered to be the Fourier coefficients of $f(z)$.

We first quote the following theorem which was proved in a more general context in [1]. For our case it reads as follows.

THEOREM 4. *Let* $\rho=1$, $G_\rho(f)=1$ *and* $\operatorname{ind}_\rho(f)=0$. *Then, as* $n\to\infty$, *we have*

$$T_n^{(0)}(f) = c\{1+o(1)\},$$
$$T_n^{(m)}(f) = c(-1)^{(n+m)m}\{T_{m-1}^{(n+1)}(f_+^{-1})+o(1)\}, \qquad m>0,$$
$$T_n^{(-m)}(f) = c(-1)^{(n+m)m}\{T_{m-1}^{(-n-1)}(f_+/f_-)+o(1)\}, \qquad m>0.$$

In this case c is even specified to be

$$\exp\left(\sum_{k=1}^{\infty}(k(\log f)_{-k}(\log f)_k)\right)$$

with $\{(\log f)_k\}_{k=-\infty}^{\infty}$ *the Fourier coefficients of* $\log f$.

We shall now derive expressions for $T_{m-1}^{(n+1)}(f_+^{-1})$ and $T_{m-1}^{(-n-1)}(f_+/f_-)$. This will be done via known asymptotic expressions for finite Toeplitz determinants. The basic result is classical and given in [7, p. 596]. Similar results are obtained in [5, p. 45].

THEOREM 5. *Let* $f(z)=\sum_0^\infty f_k z^k$ *be the Taylor series of a meromorphic function, analytic at the origin, with poles* π_k, *ordered as*

$$0<|\pi_1|\leqq|\pi_2|\leqq\cdots\leqq|\pi_N|.$$

Let $\pi_{N+1}=\infty$ *and* $f_k=0$ *for* $k<0$. *Then*

$$T_{m-1}^{(n)}(f) = c(\pi_1\cdots\pi_m)^{-n}[1+O(|\pi_m/\sigma|^n)], \qquad n\to\infty,$$

if $0<m<N+1$ *and* σ *exists satisfying* $|\pi_m|<\sigma<|\pi_{m+1}|$.

To extend this result to Toeplitz determinants of a Laurent series we use the following corollary.

COROLLARY 1. *Let* $f(z)$ *and its Laurent series be as in* (2.1)-(2.6). *Then*

$$T_{m-1}^{(n)}(f) = c(\pi_1^+\cdots\pi_m^+)^{-n}[1+O(|\pi_m^+/\sigma|^n)], \qquad n\to\infty,$$

if $0<m<\#P^++1$ *and* $|\pi_m^+|<\sigma<|\pi_{m+1}^+|$, *and*

$$T_{m-1}^{(-n)}(f) = c(\pi_1^-\pi_2^-\cdots\pi_m^-)^n[1+O(|\sigma/\pi_m^-|^n)], \qquad n\to\infty,$$

if $0<m<\#P^-+1$ *and* $|\pi_{m+1}^-|<\sigma<|\pi_m^-|$.

Proof. Split $f(z)$ as $f(z)=g(z)+\hat{g}(z)$ with

$$g(z)=\sum_0^\infty f_k z^k \quad\text{and}\quad \hat{g}(z)=\sum_1^\infty f_{-k}z^{-k}.$$

Clearly $g(z)$ defines a function with poles π_k^+, $k\in P^+$, arranged as

$$\rho<|\pi_1^+|\leqq|\pi_2^+|\leqq\cdots.$$

Because $T_{m-1}^{(n)}(f)$ for $n\geqq 0$ depends only on the coefficients $f_0, f_1, \cdots$ we have $T_{m-1}^{(n)}(f)=T_{m-1}^{(n)}(g)$. Thus Theorem 5 applies, and this proves the first part of the corollary.

Set $h(z)=\hat{g}(1/z)$. This function has poles $(\pi_k^-)^{-1}$, $k\in P^-$, ordered like

$$\rho^{-1}<|\pi_1^-|^{-1}\leqq|\pi_2^-|^{-1}\leqq\cdots.$$

$T_{m-1}^{(-n)}(f)$ depends only on $f_{-1}, f_{-2}, \cdots$ if $n\geqq m$. Since $T_{m-1}^{(-n)}(f)=T_{1-m}^{(n)}(h)=T_{m-1}^{(n)}(h)$, we can again apply Theorem 5 which gives the second part of the statement. □

With this corollary we can prove that Theorem 1 is true for $\rho=1$ and $\kappa=\text{ind}_\rho(f)=0$. The general case can be shown by using the trivial transformations given in the following lemma.

LEMMA 2. *With the notation of* § 2, *set*

$$\tilde{f}(z)=z^{-\kappa}f(z) \quad \textit{and} \quad g(z)=\tilde{f}(\rho z)=\sum_{-\infty}^{\infty} g_k z^k$$

for $r'<|z|<R'$ *with* $g_k=\rho^k f_{k+\kappa}$, $r'=r/\rho<1$ *and* $R'=R/\rho>1$. *Then we have*:

(1) $G_\rho(\tilde{f})=G_1(g)$,

(2) $T_n^{(m)}(f)=T_n^{(m-\kappa)}(\tilde{f})=\rho^{-(m-\kappa)n}T_n^{(m-\kappa)}(g)$,

(3) *If* τ *is a pole (zero) of* f, *then* τ/ρ *is a pole (zero) of* $g(z)$, *with a possible exception for* $\tau=0$ *or* $\tau=\infty$.

4. Proof of Theorem 3. Note that $f(z)$ can be written in the form (2.1) if we take

$$\begin{array}{ccccccccc}
\zeta_1, & \zeta_2, & \cdots, & \zeta_{\#Z^+}, & \zeta_{\#Z^+1}, & \cdots, & \zeta_N, \\
\| & \| & & \| & \| & & \| \\
\zeta^+_{\#Z^+}, & \zeta^+_{\#Z^+-1}, & \cdots, & \zeta^+_1, & \zeta^-_1, & \cdots, & \zeta^-_{\#Z^-}
\end{array}$$

with $|\zeta_i^+|>\rho$ for $i=1,2,\cdots,\#Z^+$ and $|\zeta_i^-|<\rho$ for $i=1,2,\cdots,\#Z^-$.

Similarly for the poles.

$$\begin{array}{ccccccccc}
\pi_1, & \pi_2, & \cdots, & \pi_P, & \pi_{P+1}, & \cdots, & \pi_M, \\
\| & \| & & \| & \| & & \| \\
\pi^+_{\#P^+}, & \pi^+_{\#P^+-1}, & \cdots, & \pi^+_1, & \pi^-_1, & \cdots, & \pi^-_{\#P^-}
\end{array}$$

($\#P^+=P$ and $\#P^-=M-P$).

Clearly $\kappa=\#Z^--\#P^-+l$, and

$$G_\rho(\tilde{f})=K\cdot\frac{(-\zeta_1^+)\cdots(-\zeta^+_{\#Z^+})}{(-\pi_1^+)\cdots(-\pi^+_{\#P^+})}.$$

Theorem 3 now directly follows from Theorem 1 if you observe that $0<m-\kappa<\#Z^++1$ is equivalent with $0\leqq k-m<\#Z^+$, $0<\kappa-m<\#Z^-+1$ is equivalent with $\#Z^+<k-m\leqq N$ and $k-m=\#Z^+$ is equivalent with $m=\kappa$.

5. Conclusion. With the multiplicative splitting for f defined in (2.2)-(2.6), it is shown that the asymptotic behavior of the Toeplitz determinants $T_n^{(m)}(f)$ are mainly defined by $T_n^{(m-\kappa)}(f_+)$ or $T_n^{(\kappa-m)}(\hat{f}_-)$ depending on m being larger or smaller than κ. With these formulas it is possible to extend the results for a rational function given in [2] to the general meromorphic case.

REFERENCES

[1] A. BÖTTCHER AND B. SILBERMANN, *The asymptotic behavior of Toeplitz determinants for generating functions with zeros of integral orders*, Math. Nachr., 102 (1981), pp. 70-105.

[2] A. BULTHEEL, *Zeros of a rational function defined by its Laurent series*, in Padé Approximation and its Applications, Bad Honnef 1983, H. Werner and H. J. Bünger, eds., Springer-Verlag, Berlin, 1984, pp. 34-48.

[3] A. BULTHEEL, *Quotient-difference relations in connection with* AR *filtering*, Proc. ECCTD '83, VDE-Verlag, Berlin, 1983, pp. 395-399.
[4] K. M. DAY, *Toeplitz matrices generated by an arbitrary rational function*, Trans. Amer. Math. Soc., 206 (1975), pp. 224-245.
[5] W. B. GRAGG, *The Padé table and its relation to certain algorithms of numerical analysis*, SIAM Rev., 14 (1972), pp. 1-62.
[6] ———, *Laurent, Fourier, and Chebyshev-Padé tables*, in Padé and Rational Approximation, E. B. Saff and R. S. Varga, eds., Academic Press, New York, 1977, pp. 61-72.
[7] P. HENRICI, *Applied and Computational Complex Analysis*, Vol. 1, John Wiley, New York, 1974.
[8] W. B. JONES AND A. MAGNUS, *Computation of poles of two-point Padé approximants and their limits*, J. Comp. Appl. Math., 6 (1980), pp. 105-119.
[9] W. B. JONES AND W. J. THRON, *Continued Fractions, Analytic Theory and Applications*, Academic Press, New York, 1980.
[10] J. N. MCCABE AND J. A. MURPHY, *Continued fractions which correspond to power series expansions at two points*, J. Inst. Math. Appl., 17 (1976), pp. 233-247.
[11] H. RUTISHAUSER, *Der quotienten-differenzen-Algorithmus*, Z. Angew Math. Phys., 5 (1954), pp. 233-251.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 4, October 1985

010

MULTI-SPLITTINGS OF MATRICES AND PARALLEL SOLUTION OF LINEAR SYSTEMS*

DIANNE P. O'LEARY† AND R. E. WHITE‡

Abstract. We present two classes of matrix splittings and give applications to the parallel iterative solution of systems of linear equations. These splittings generalize regular splittings and P-regular splittings, resulting in algorithms which can be implemented efficiently on parallel computing systems. Convergence is established, rate of convergence is discussed, and numerical examples are given.

Key words. matrix splittings, iterative methods for linear systems, parallel computation, regular splittings

AMS(MOS) subject classifications. 65F10, 65N20

1. Introduction.

Consider the solution of a large linear system of equations

$$Ax = b$$

on a parallel computer. We assume that several processors are available and that they can execute different instruction sequences on their local data and can communicate with physically adjacent processors.

In this paper we consider the problem of solving linear systems for which the matrix A can either be split into many pieces or split into two pieces in many ways. An example of the first case is the assembly of a finite element matrix by elements. In that case A can be decomposed as

$$A = \sum_{k=1}^{K} A_k$$

where each matrix A_k has small rank. The second case arises from having several candidate iterative methods

$$B_k x_{i+1} = C_k x_i + b, \qquad i = 0, 1, \cdots,$$

where for $k = 1, 2, \cdots, K$, $A = B_k - C_k$.

We discuss ways of using these two kinds of decompositions of A in order to construct convergent iterative methods which are structured so that most operations can be performed in parallel. We base such iterative methods on *multi-splittings* of the matrix A.

In § 2 we define multi-splittings and prove some convergence results for these iterative methods. Section 3 provides a discussion of parallelism in the iterative methods, examples of problems for which multi-splittings can be used, and motivation for the definitions and results of § 2. Section 4 provides results of some numerical experiments on multi-splittings. It is possible to read §§ 3 and 4 before § 2 if a reader is so inclined.

2. Multi-splittings: definitions and theory.

We begin with a definition of a multi-splitting of a matrix A, discuss its use in an iterative method for solving linear systems, and prove some convergence results. For notational convenience we omit the lower and upper limits 1 and K on all sums and the indices $k = 1, \cdots, K$ on ordered triples (B_k, C_k, D_k).

* Received by the editors January 24, 1984, and in revised form April 19, 1984.

† Computer Science Department, University of Maryland, College Park, Maryland 20742. The work of this author was supported by the Air Force Office of Scientific Research under grant AFOSR-82-0078.

‡ Department of Mathematics, North Carolina State University, Raleigh, North Carolina 27695-8205.

DEFINITION. Let A, B_k, C_k, and D_k be $n \times n$ matrices. Then (B_k, C_k, D_k) is called a *multi-splitting of* A if

i) $A = B_k - C_k$, $k = 1, \cdots, K$, where each B_k is invertible.

ii) $\sum_k D_k = I$ where the matrices D_k are diagonal and $D_k \geqq 0$.

We will use the notation

$$\mathbf{H} = \sum_k D_k B_k^{-1} C_k \quad \text{and} \quad \mathbf{G} = \sum_k D_k B_k^{-1}.$$

We are interested in the convergence of an iterative method based on $\mathbf{H}$ and $\mathbf{G}$ for solving $Ax = b$. Using (i) above, $Ax = b$ may be written as

$$B_k x = C_k x + b, \qquad k = 1, \cdots, K$$

or

$$x = B_k^{-1} C_k x + B_k^{-1} b, \qquad k = 1, \cdots, K.$$

We use the weighting matrices D_k to combine these K equations as

$$\sum_k D_k x = \sum_k D_k B_k^{-1} C_k x + \sum_k D_k B_k^{-1} b,$$

which, by (ii) and the definitions of $\mathbf{H}$ and $\mathbf{G}$ yields the following algorithm.

ALGORITHM 1

Choose x_0 arbitrarily.

For $i = 0, 1, 2, \cdots$, until convergence

$x_{i+1} = \mathbf{H} x_i + \mathbf{G} b$.

The parallelism in a variant of this algorithm will be discussed in § 3.

It would be convenient if it were true that whenever the iterative methods based on each of the splittings $A = B_k - C_k$ converged, then Algorithm 1 produced a convergent sequence, too. Unfortunately, the situation is more complicated than that, as the following trivial example shows.

Example. Let $K = 2$, $n = 2$, and consider

$$A = \begin{bmatrix} \frac{3}{4} & 0 \\ 0 & \frac{3}{4} \end{bmatrix} = B_1 - C_1 = B_2 - C_2,$$

where

$$B_1 = \begin{bmatrix} .5 & -1 \\ 1 & 4 \end{bmatrix}, \qquad C_1 = \begin{bmatrix} -.25 & -1 \\ 1 & 3.25 \end{bmatrix},$$

and

$$B_2 = \begin{bmatrix} 4 & 1 \\ -1 & .5 \end{bmatrix}, \qquad C_2 = \begin{bmatrix} 3.25 & 1 \\ -1 & -.25 \end{bmatrix}.$$

Then

$$B_1^{-1} C_1 = \begin{bmatrix} 0 & -.25 \\ .25 & .875 \end{bmatrix}, \qquad B_2^{-1} C_2 = \begin{bmatrix} .875 & .25 \\ -.25 & 0 \end{bmatrix}.$$

The spectral radius ρ for both matrices is .7965, so iterations based on both splittings

are convergent. But, with the choice

$$D_1 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}, \qquad D_2 = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix},$$

the resulting iteration matrix is

$$\mathbf{H} = \begin{bmatrix} .875 & .25 \\ .25 & .875 \end{bmatrix}$$

for which the spectral radius is 1.125, and therefore Algorithm 1 would not be convergent. Other choices of D_1 and D_2 will change this situation, of course. For example, if the definitions of D_1 and D_2 above are interchanged, the resulting matrix $\mathbf{H}$ has spectral radius equal to $\frac{1}{4}$.

Recall that (B, C) is a weak regular splitting of A if $A = B - C$, $B^{-1} \geqq 0$, and $H \equiv B^{-1}C \geqq 0$. Similarly, (B, C) is called a P-regular splitting of A if B is nonsingular and the symmetric part of $B + C$ is positive definite. From standard results in the theory of iterative methods (see, for example, [1] and [6]),

(1) If (B, C) is a weak regular splitting of a matrix A satisfying $A^{-1} \geqq 0$, and $H = B^{-1}C$, then $\rho(H) < 1$.

(2) If (B, C) is a P-regular splitting of a symmetric positive definite matrix A, and $H = B^{-1}C$, then $\rho(H) < 1$.

(3) If $\|H\| < 1$ for any matrix norm, then $\rho(H) < 1$.

We seek conditions on the multi-splitting (B_k, C_k, D_k) which will ensure that analogous results apply to the splitting resulting in $\mathbf{H}$. Consequently, these conditions will ensure that Algorithm 1 is convergent. The example above shows that the second result does not have a direct analogue: it is not enough that each splitting in the multisplitting is a P-regular splitting of a symmetric positive definite matrix. The other two results do generalize without additional hypotheses.

THEOREM 1. (a) *If, for* $k = 1, 2, \cdots, K$, (B_k, C_k) *is a weak regular splitting of a matrix* A *satisfying* $A^{-1} \geqq 0$, *then Algorithm* 1 *is convergent.*

(b) *If, for* $k = 1, 2, \cdots, K$, (B_k, C_k) *is a* P*-regular splitting of a symmetric positive definite matrix* A *and* $D_k = \alpha_k I$, *then Algorithm* 1 *is convergent.*

(c) *If, for* $k = 1, 2, \cdots, K$, $\|B_k^{-1}C_k\|_\infty < 1$, *then Algorithm* 1 *is convergent.*

Proof. (a) The proof parallels the proof for convergence of weak regular splittings found, for example, in Ortega [6]. From the definitions of $\mathbf{H}$ and weak regular splitting we have the following three facts:

1. $\mathbf{H} \geqq 0$ and therefore $\mathbf{H}^j \geqq 0$, $j = 0, 1, \cdots$.
2. $I - \mathbf{H} = \sum_k D_k B_k^{-1} A$.
3. $(I + \mathbf{H} + \cdots + \mathbf{H}^m)(I - \mathbf{H}) = I - \mathbf{H}^{m+1}$.

Now, using these facts in order,

$$\begin{aligned} 0 &\leqq (I + \mathbf{H} + \cdots + \mathbf{H}^m) \sum_k D_k B_k^{-1} \\ &= (I + \mathbf{H} + \cdots + \mathbf{H}^m)(I - \mathbf{H}) A^{-1} \\ &= (I - \mathbf{H}^{m+1}) A^{-1} \leqq A^{-1}. \end{aligned}$$

Therefore, the elements of $\mathbf{H}^m$ must remain bounded, and therefore $\mathbf{H}$ is convergent.

(b) Again, the proof parallels a standard proof of convergence, that for P-regular splittings [6]. It is sufficient to show that $A - \mathbf{H}^T A \mathbf{H}$ is positive definite; then the result

that $\rho(\mathbf{H})<1$ follows from a theorem of Stein [9]. We use the notation $B^{-T}\equiv(B^T)^{-1}=(B^{-1})^T$.

$$
\begin{aligned}
A-\mathbf{H}^TA\mathbf{H} &= A-\Big(I-\sum_k D_kB_k^{-1}A\Big)^TA\Big(I-\sum_k D_kB_k^{-1}A\Big)\\
&=\sum_k AD_kB_k^{-1}A+\sum_k AB_k^{-T}D_kA-\sum_{k,j} AB_k^{-T}D_kAD_jB_j^{-1}A\\
&=\sum_k AB_k^{-T}[B_k^TD_k+D_kB_k-D_kAD_k]B_k^{-1}A\\
&\quad-\sum_{\substack{k,j\\k\neq j}} AB_k^{-T}D_kAD_jB_j^{-1}A\\
&=\sum_k AB_k^{-T}[B_k^TD_k+D_kA+D_kC_k-D_kAD_k]B_k^{-1}A\\
&\quad-\sum_{\substack{k,j\\k\neq j}} AB_k^{-T}D_kAD_jB_j^{-1}A\\
&=\sum_k AB_k^{-T}[B_k^TD_k+D_kC_k+\sum_{\substack{j\\j\neq k}} D_kAD_j]B_k^{-1}A\\
&\quad-\sum_{\substack{k,j\\k\neq j}} AB_k^{-T}D_kAD_jB_j^{-1}A\\
&=\sum_k AB_k^{-T}[B_k^TD_k+D_kC_k]B_k^{-1}A\\
&\quad+\sum_{\substack{k,j\\k\neq j}} AB_k^{-T}D_kAD_j[B_k^{-1}A-B_j^{-1}A]\\
&\equiv \mathbf{S}_1+\mathbf{S}_2.
\end{aligned}
$$

Let $\operatorname{sym}(P)\equiv(P+P^T)/2$ denote the symmetric part of the matrix P. Then

$$\operatorname{sym}(\mathbf{S}_1)=\sum_k \alpha_kAB_k^{-T}\operatorname{sym}(B_k+C_k)B_k^{-1}A,$$

and each of these terms is positive definite. Now

$$
\begin{aligned}
2\operatorname{sym}(\mathbf{S}_2) &= \sum_{\substack{k,j\\k\neq j}} \alpha_k\alpha_j[(AB_k^{-T}-AB_j^{-T})AB_k^{-1}A+AB_k^{-T}A(B_k^{-1}A-B_j^{-1}A)]\\
&= \sum_{\substack{k,j\\k\neq j}} \alpha_k\alpha_j[AB_k^{-T}AB_k^{-1}A-AB_k^{-T}AB_j^{-1}A-AB_j^{-T}AB_k^{-1}A+AB_j^{-T}AB_j^{-1}A]\\
&= \sum_{\substack{k,j\\k\neq j}} \alpha_k\alpha_j[(AB_k^{-T}-AB_j^{-T})A(B_k^{-1}A-B_j^{-1}A)].
\end{aligned}
$$

Thus $\operatorname{sym}(\mathbf{S}_2)$ is positive definite and the result is established.

(c) The infinity norm of a matrix is the maximum absolute row sum, and the absolute row sums of $\mathbf{H}$ are bounded by convex combinations of the absolute row sums of $B_k^{-1}C_k$. Thus $\|\mathbf{H}\|_\infty<1$, and convergence is established. □

This theorem says that if we have a collection of convergent splittings of a matrix, then under certain conditions we can construct a convergent multi-splitting. There is another way to construct convergent multi-splittings. We break the matrix into simple pieces A_k and add diagonal matrices E_k to ensure that each $B_k\equiv A_k+E_k$ is invertible.

DEFINITION. Let A_k, B_k, and D_k be $n \times n$ matrices. Then (A_k, E_k, D_k) is called a *dissolution of* A if

i) $A = \sum_k A_k$.

ii) E_k and D_k are diagonal matrices.

iii) (B_k, C_k, D_k) is a multi-splitting of A, where $B_k = A_k + E_k$ and $C_k = E_k - \sum_{j \neq k} A_j$.

We call (A_k, E_k, D_k) a *convergent dissolution of* A if it is a dissolution for which the multi-splitting (B_k, C_k, D_k) leads to a convergent algorithm.[1]

The next theorem gives explicit conditions on the matrices A, A_k, and E_k such that (A_k, E_k, D_k) will be a convergent dissolution of a matrix A.

THEOREM 2. *Let $A = \sum_k A_k$ be an M-matrix and let the matrices E_k be nonnegative diagonal matrices with diagonal components equal to e_{lk}. Then if the matrices $A_k = (a^k_{lm})$ satisfy*

(a) $0 \leqq -a^k_{lm} \leqq -a_{lm}$, $l \neq m$,

(b) $e_{lk} + a^k_{ll} > -\sum_{m \neq l} a^k_{lm}$,

(c) $e_{lk} + a^k_{ll} \geqq a_{ll}$,

then for all nonnegative diagonal matrices D_k with $\sum_k D_k = I$, (A_k, E_k, D_k) is a convergent dissolution of A.

Proof. Let us examine the elements of C_k. For $l \neq m$,

$$c^k_{lm} = -\sum_{j \neq k} a^j_{lm} = a^k_{lm} - a_{lm} \geqq 0$$

using assumption (a) and the fact that $A = \sum_k A_k$. By (c), $c^k_{ll} = e_{lk} + a^k_{ll} - a_{ll} \geqq 0$. Now $B_k = A_k + E_k$ satisfies $b^k_{lm} = a^k_{lm} \leqq 0$, $l \neq m$, by (a), and $b^k_{ll} = a^k_{ll} + e_{lk} > 0$ by (c). Further, by (b), B_k is a strictly row diagonally dominant matrix. Therefore, B_k is an M-matrix [7] and $B_k^{-1} \geqq 0$. Thus $A = B_k - C_k$ is a weak regular splitting for each k, and, by Theorem 1a, the multi-splitting is convergent. □

THEOREM 3. *Let $A = \sum_k A_k$ be a symmetric positive definite matrix, and let $A_k + E_k$ be nonsingular and $2(A_k + E_k) - A$ be positive definite, $k = 1, 2, \cdots, K$. Then for nonnegative diagonal matrices $D_k = \alpha_k I$, (A_k, E_k, D_k) is a convergent dissolution of A.*

Proof. The conditions in the theorem assure that $B_k = A_k + E_k$ is invertible and

$$B_k + C_k = A_k + 2E_k - \sum_{j \neq k} A_j = 2(A_k + E_k) - A$$

is positive definite. Convergence follows from Theorem 1b. □

3. Examples of multi-splittings. In this section we construct some examples of convergent multi-splittings of matrices. We also discuss the use of multi-splittings on parallel computers. Many other approaches to parallel iterative methods have been developed; see, for example, [2], [4], [8]. We consider the following algorithm for solving $Ax = b$. It is equivalent to Algorithm 1 when $\omega = 1$.

ALGORITHM 2

Choose x_0 arbitrarily; choose a parameter ω.

For $i = 1, 2, \cdots$ until convergence

  Let $\bar{x} = x_i$.

  For $k = 1, 2, \cdots, K$

    Find $D_k y_k$ where y_k satisfies

(3.1) $B_k y_k = C_k \bar{x} + b$.

  Form $x_{i+1} = (1 - \omega) x_i + \omega \sum_k D_k y_k$.

[1] We use the term "dissolution" in the sense of "the breaking up of an assembly or organization" (*Random House Dictionary*, 1980).

As before, D_k is a diagonal matrix and $A = B_k - C_k$.

Parallelism in Algorithm 2 could be exploited in several ways, depending on the precise machine architecture and the choice of the multi-splitting (B_k, C_k, D_k). First, the computations in (3.1) for various k are independent, and could be performed in parallel. (Note that if a main diagonal element of D_k is zero, the corresponding component of y_k need not be computed at all.) Second, the n components of a single vector y_k (or x_{i+1}) could be computed in parallel. Third, the accumulation of the sum of $K+1$ terms which forms a component of x_{i+1} could be formed in $O(\log_2(K))$ time using parallel computation.

Our first two examples illustrate the use of convergent multi-splittings to solve algebraic systems resulting from applying the finite difference and finite element methods to partial differential equations. In both examples, the original matrix is decomposed into a sum of matrices which are considerably "simpler" than the original one and which reflect significant contributions to A from given subsets of nodes. Thus it is natural to use these decompositions of A as the basis for a dissolution as defined in § 2.

Example 1. *Decomposition by blocks of unknowns.* Consider the partial differential equation

$$-u_{xx} - u_{yy} = f \quad \text{on } \Omega, \qquad u = g \quad \text{on } \partial\Omega.$$

Let Ω be a square and use the second order accurate 5-point finite difference method to discretize the equation with m equally spaced interior mesh points in each direction. This gives the algebraic equation $Au = \bar{f}$ where A is an $m^2 \times m^2$ matrix, $\bar{f}$ is a $m^2 \times 1$ column vector whose components reflect f, g, and the dimension m, and $u = (u_{11}, \cdots, u_{1m}, \cdots, u_{m1}, \cdots, u_{mm})^T$. The matrix may be written as

$$A = \begin{bmatrix} B & -I & & & \\ -I & B & -I & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & -I \\ & & & -I & B \end{bmatrix},$$

where

$$B = \begin{bmatrix} 4 & -1 & & & \\ -1 & 4 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & -1 \\ & & & -1 & 4 \end{bmatrix},$$

and is of dimension $m \times m$. This linear system can be solved, for example, by the alternating direction implicit (ADI) iterative method. One version of this method is, first, solve m sets of equations, one set for each row of mesh points, and second, solve another m sets of equations, one set for each column. That is, A is decomposed as a sum of 2 matrices:

$$A = \begin{bmatrix} T & & & \\ & T & & \\ & & \ddots & \\ & & & T \end{bmatrix} + P \begin{bmatrix} T & & & \\ & T & & \\ & & \ddots & \\ & & & T \end{bmatrix} P^T,$$

where

$$T = \begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \cdot & \vdots & \vdots & \\ & & & & -1 \\ & & & -1 & 2 \end{bmatrix}_{m \times m},$$

and P is the permutation matrix which would reorder the vector u as $Pu = (u_{11}, \cdots, u_{m1}, \cdots, u_{1m}, \cdots, u_{mm})^T$. This can further be broken into the sum of $2m$ matrices A_k, each one corresponding to one of the matrices T. We introduce nonnegative diagonal matrices E_k such that the matrices $B_k \equiv A_k + E_k$ are invertible, and construct a set of nonnegative diagonal matrices D_k which sum to I. Since it is natural to let a diagonal component of D_k be zero if the component corresponds to a mesh point or element not in block k, most of the linear systems in Equation (3.1) of Algorithm 2 do not require the computation of a full n-dimensional problem but one whose size is much smaller—dimension m. The matrix E_k can be taken as 0 when A_k is nonzero and as arbitrary positive diagonal elsewhere. The solution of each linear system is independent of the others and can be performed in parallel if sufficient processors are available. Under natural assignments of unknowns to processors, nearby mesh points will be computed in nearby processors, so communication in step (3.1) will be local. Theorem 1a applies to this multisplitting and assures convergence.

Example 2. *Decomposition by finite elements.* Consider the Galerkin formulation of the finite element method applied to the ordinary differential equation

$$-u_{xx} = f, \qquad u(0) = u_0, \quad u(1) = u_1.$$

When linear shape functions are used on an equally spaced mesh of size $1/(m+1)$, this method gives a system of equations $Au = \bar{f}$, where A is the matrix T defined in Example 1 above. This matrix may be "assembled" by using the element matrices. The domain, $[0, 1]$, is a union of $m+1$ elements $[x_i, x_i + \Delta x]$, where $m \geqq 3$. The element matrices have the form

$$\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}.$$

Then A may be written as

$$A = \sum_{k=1}^{m-1} A_k,$$

where

$$A_1 = \begin{bmatrix} 2 & -1 & & & \\ -1 & 1 & & & \\ & & 0 & & \\ & & & \ddots & \\ & & & & 0 \end{bmatrix},$$

$$A_{m-1} = \begin{bmatrix} 0 & & & & \\ & \ddots & & & \\ & & 0 & & \\ & & & 1 & -1 \\ & & & -1 & 2 \end{bmatrix},$$

and A_k, $k \neq 1, m-1$ has the element matrix as a diagonal block starting in row and column k and zeros elsewhere. This splitting of finite element matrices has also been used as the basis of an iterative method by Hayes and Devloo [3]. Let $B_k = A_k + E_k$ where E_k is any diagonal matrix which makes $B_k^{-1} \geqq 0$. (For example, it is sufficient that all diagonal elements of B_k be equal to 2.) Then Theorem 1a ensures that Algorithm 2 will converge. Again, under natural assignment of nodes to processors, high parallelism can be achieved.

We have chosen trivial problems in Examples 1 and 2 to make the descriptions easier, but the methods are equally applicable to irregular meshes in several space dimensions.

We now give an example of a convergent multi-splitting which is not derived from a dissolution.

Example 3. *Decomposition by block iterative methods.* Let A be a sparse M-matrix (The matrices of the first two examples satisfy this hypothesis). Assume that each unknown has been assigned to a processor on a parallel computer. Choose some subset of K unknowns, and direct the corresponding processors to "grow" a block of unknowns from those local to it and in nearby processors in order to identify a principal submatrix of A for which linear systems are easy to solve. Note that an unknown may appear in several blocks, and the idea is to let the blocks grow to some point such that each unknown appears in at least one block and the work among processors is nearly balanced. Then, for $k = 1, 2, \cdots, K$, we have partitioned a permuted version of A as

$$\begin{bmatrix} G_{11}^k & G_{12}^k \\ G_{21}^k & G_{22}^k \end{bmatrix},$$

where G_{11}^k is the principal submatrix grown by the kth unknown. Then let

$$B_k = \begin{bmatrix} G_{11}^k & 0 \\ 0 & G_{22}^k \end{bmatrix}, \qquad C_k = \begin{bmatrix} 0 & G_{12}^k \\ G_{21}^k & 0 \end{bmatrix}, \qquad D_k = \begin{bmatrix} D_{11}^k & 0 \\ 0 & 0 \end{bmatrix},$$

where each diagonal element of $(D_{11}^k)^{-1}$ equals the number of blocks in which the corresponding unknown appears. We have a set of regular splittings of the M-matrix (because each corresponds to a block Jacobi method), and Theorem 1a assures convergence. Note that the blocks corresponding to the second row of B_k and C_k are never used since the corresponding elements of D_k are 0.

Although convergence is assured in each of these examples, it may be too slow in practice. The practical use of these algorithms in the parallel solution of sparse linear systems may be as highly parallel preconditionings of some faster iterative method such as conjugate gradients or block conjugate gradients [5].

4. Numerical examples. In the following examples we apply Algorithm 2 to two problems and study the convergence of the algorithm as the block size, the choice of E_k, and the choice of ω are changed. The second example arises from an elliptic boundary value problem and is more realistic than the first in the size and character of the resulting matrix.

Numerical example 1. Consider the ordinary differential equation

$$-u_{xx} = f(x) = 10, \qquad u(0) = 1 = u(1).$$

Let the interval [0, 1] be divided into 18 equal elements of length $h = \frac{1}{18}$, and consider

the Galerkin formulation of the finite element method with linear shape functions. The resulting algebraic system is

$$\frac{1}{h}(-u_{i+1}+2u_i-u_{i-1})=\frac{h}{6}(f_{i+1}+4f_i+f_{i-1}), \qquad i=1,2,\cdots,17,$$

with $u_0=1=u_{18}$. We construct $K=16$ matrices $A_k^{(K)}$ as in Example 2 of § 3, and further consider examples in which $K=8$, $K=4$, and $K=2$. In these later examples, $A_j^{(K)}$ is formed by grouping together the elements which contributed to $A_{2j-1}^{(2K)}$ and $A_{2j}^{(2K)}$. (The case $K=1$ reduces to one iteration in which a tridiagonal problem is solved for 17 unknowns.) We choose the weighting matrices D_k to have either zero, $\frac{1}{2}$, or 1 as diagonal components:

0 if node i does not belong to finite elements in the kth set,
$\frac{1}{2}$ if node i is on the boundary of the kth set of elements,
1 if node i is in the interior of the kth set of elements.

The first choice of E_k has the form $E_k=(d/h)I$. In the last set of experiments, we used E_k defined by

$$(E_k)_{ii}=\begin{cases}0 & \text{if node } i \text{ is in the interior of block } k,\\ 1/h & \text{otherwise.}\end{cases}$$

This choice means that the diagonals of the iteration matrix $B_k=A_k+E_k$ match those of A for all components in element block k.

Tables 1-4 indicate the number of iterations required to reach convergence, defined when the relative error for each node was less than 10^{-4}. The initial guess was taken to be the vector of all one's. The values of d in Table 2 and ω in Table 3 are near optimal.

TABLE 1
Algorithm performance on Example 1 varying K

K	ω	d	Number of iterations
2	1.00	0.35	72
4	1.00	0.35	74
8	1.00	0.35	75
16	1.00	0.35	∞

TABLE 2
Algorithm performance on Example 1 with $\omega=1$ and near optimal $E_k=(d/h)I$

K	ω	d	Number of iterations
2	1.00	0.05	18
4	1.00	0.20	47
8	1.00	0.35	75
16	1.00	0.70	127

TABLE 3
Algorithm performance on Example 1 with $d = 1$ and near optimal $\omega = 1.32$

K	ω	d	Number of iterations
2	1.32	1.00	138
4	1.32	1.00	137
8	1.32	1.00	136
16	1.32	1.00	135

TABLE 4
Algorithm performance on Example 1 for the second choice of E_k

K	ω	Number of iterations
2	1.00	33
4	1.00	55
8	1.00	96
16	1.00	169

Numerical example 2. Consider the elliptic partial differential equation

$$-(c_1 u_x)_x - (c_2 u_y)_y = g \quad \text{on } \Omega \equiv (0,1)\times(0,1) - [\tfrac{2}{5}, \tfrac{3}{5}]\times[\tfrac{2}{5}, \tfrac{3}{5}],$$
$$u = x^2 + y^2 \quad \text{on } \partial\Omega,$$

where

$$c_1 = 1 + x^2 + y^2,$$
$$c_2 = 1 + e^x + e^y,$$
$$g = -2(2 + 3x^2 + y^2 + e^x + (1+y)e^y).$$

The data have been chosen so that the solution is $u = x^2 + y^2$. This problem is discretized by the second order accurate finite difference method with mesh spacings in both directions equal to $h = 1/(m+1)$ where $m = 9$, $m = 19$ or $m = 29$. We consider a multi-splitting as in Example 1 of § 3. Thus each A_k corresponds to some row or column of mesh points. When $m = 9$, $K = 24$ and the number of unknowns is $N = 9^2 - 3^2 = 72$. When $m = 19$, $K = 48$ and $N = 19^2 - 5^2 = 336$; when $m = 29$, $K = 72$ and

TABLE 5
Algorithm performance on Example 2

N	ω	Number of iterations
72	1.30	17
336	1.30	91
792	1.30	234

$N = 29^2 - 7^2 = 792$. Since each mesh point is involved in exactly two matrices A_k, we take the ith diagonal element of D_k equal to $1/2$ if mesh point i is in block k and 0 otherwise. Let E_k be that matrix which makes the diagonal elements of $B_k = A_k + E_k$ equal to the diagonal elements of A whose rows correspond to nodes in block k. (Numerical experiments showed that this choice led to fewer iterations than a choice of the form $E_k = (d/h^2)I$.) Convergence was defined by the l_2 norm of the discrete error vector being less than h^2. The initial guess was the zero vector. By numerical experiments, $\omega = 1.3$ was determined to be near optimal. Results appear in Table 5.

REFERENCES

[1] A. BERMAN AND R. J. PLEMMONS, *Nonnegative Matrices in the Mathematical Sciences*, Academic Press, New York, 1979.

[2] V. CONRAD AND Y. WALLACH, *Alternating methods for sets of linear equations*, Numer. Math., 32 (1979), pp. 105-108.

[3] L. J. HAYES AND P. DEVLOO, *An overlapping block iterative scheme for finite element methods*, Dept. Aerospace Engineering and Engineering Mechanics Report, University of Texas at Austin, to appear.

[4] R. W. HOCKNEY AND C. R. JESSHOPE, *Parallel Computers: Architecture, Programming, and Algorithms*, Adam and Hilger, Bristol, 1981.

[5] D. P. O'LEARY, *The block conjugate gradient algorithm and related methods*, Linear Algebra Appl., 29 (1980), pp. 293-322.

[6] JAMES M. ORTEGA, *Numerical Analysis, A Second Course*, Academic Press, New York, 1972.

[7] A. M. OSTROWSKI, *Uber die Determinanten mit überwiegender Hauptdiagonale*, Comment. Math. Helv., 10 (1937), pp. 69-96.

[8] F. ROBERT, *Méthodes itératives série parallèle*, C.R. Acad. Sc. Paris, A-271 (1970), pp. 847-850.

[9] P. STEIN, *Some general theorems on iterants*, J. Res. Nat. Bur. Standards, 48 (1952), pp. 82-83.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 4, October 1985

011

HOMOLOGY AS A TOOL IN INTEGER PROGRAMMING*

SAUL STAHL†

Abstract. Several years ago Lovász [J. Combin. Theory (A), 25 (1978), pp. 319-324] pointed out that homotopy theory has very deep applications to graph colorings. These ideas are extended a little further here to show that homology theory, whose groups are easily computable, can be used to obtain bounds on the solutions of certain integer programs. Some graph coloring techniques which are closely related to those of Lovász are also shown to have similar applications.

AMS(MOS) subject classifications. 90C10, 55M20

As part of his startling resolution of the Kneser conjecture, Lovász [1] associated a simplicial complex $N(G)$ with every graph G and proved the following theorem.

THEOREM 1. *If the homotopy groups* $\pi_0(N(G)), \cdots, \pi_k(N(G))$ *are all trivial, then the chromatic number of the graph* G *is at least* $k+3$.

A homological version of this theorem has recently been proved independently by J. W. Walker [4] and by A. H. Wright [5]. This may be restated as follows, where the homology groups are reduced with coefficients in Z_2.

THEOREM 1′. *If the homology groups* $\tilde{H}_0(N(G)), \cdots \tilde{H}_k(N(G))$ *are all trivial, then the chromatic number of the graph* G *is at least* $k+3$.

The main advantage of this version is that the homology groups are computable whereas the homotopy groups are not. Consequently, if G is a graph with at least one edge, then the largest k which satisfies the hypothesis of Theorem 1′ is also effectively computable. We denote this value by $\chi_\lambda(G)$. Thus, Theorem 1′ might be simply restated as $\chi(G) \geqq \chi_\lambda(G)+3$.

It is, of course, widely known that the evaluation of the chromatic number of a graph can be expressed as an integer program. Specifically, if $I_1, \cdots, I_q$ are the independent sets of vertices of the graph G, and $v_1, \cdots, v_p$ are its vertices, then the matrix $M = m(G)$ is the associated $p \times q$ incidence matrix whose columns correspond to the independent sets and whose rows correspond to the vertices of G. The chromatic number of G then equals $\min \{\mathbf{x} \cdot \mathbf{l}_q | M\mathbf{x} \geqq \mathbf{l}_p, \mathbf{x} \in Z_+^q\}$ where Z_+^q denotes the set of all vectors in R^q with nonnegative integer components and $\mathbf{l}_p$ and $\mathbf{l}_q$ denote vectors of all whose components are 1. This formulation indicates that Theorem 1 might also be applicable to other integer programs as well—as is indeed the case.

The graph G can be recovered from $m(G)$ by noting that two vertices are adjacent in G if and only if their corresponding rows in $m(G)$ are orthogonal. This motivates the following definition. Let M be a $p \times q$ matrix all of whose entries are either 0 or 1. We define a graph $G = g(M)$ whose vertices are the rows of M. Two vertices of G are adjacent if and only if they are orthogonal as rows of M.

THEOREM 2. *Let* M *be a* $p \times q$ *matrix with* 0, 1 *entries and distinct columns. If* $G = g(M)$, *then*

$$\min \{\mathbf{x} \cdot \mathbf{l}_q | M\mathbf{x} \geqq \mathbf{l}_p, \mathbf{x} \in Z_+^q\} \geqq \chi_\lambda(G)+3.$$

Proof. Let $M' = m(G)$ be a $p \times q'$ matrix. Then each column of M is also a column of M'. For suppose $\mathbf{c} = (c_1, c_2, \cdots, c_p)$ is a column of M with $c_i = 1$ if and only if $i = i_1, i_2, \cdots, i_k$, and $c_i = 0$ otherwise. By the definition of adjacency in G,

* Received by the editors September 6, 1983, and in revised form June 5, 1984.
† Department of Mathematics, University of Kansas, Lawrence, Kansas 66045.

$\{v_{i_1}, v_{i_1}, \cdots, v_{i_k}\}$ is an independent set of vertices of G and hence $\mathbf{c}$ is also a column of M'. Reorder the columns of M' so that its first q columns agree with the q columns of M. If $\mathbf{x}_0$ is a vector of R_+^q such that $\mathbf{x}_0 \cdot \mathbf{l}_q = \min\{\mathbf{x} \cdot \mathbf{l}_q | M\mathbf{x} \geqq \mathbf{l}_p, \mathbf{x} \in Z_+^q\}$ let $\mathbf{x}_0^*$ be the vector in $R_+^{q'}$ obtained by tagging $(q'-q)$ zero components on to $\mathbf{x}_0$. The following string of inequalities is then easily verified.

$$\begin{aligned}
&\min\{\mathbf{x} \cdot \mathbf{l}_q | M\mathbf{x} \geqq \mathbf{l}_p, \mathbf{x} \in Z_+^q\} \\
&\quad = \mathbf{x}_0 \cdot \mathbf{l}_q = \mathbf{x}_0^* \cdot \mathbf{l}_{q'} \\
&\quad \geqq \min\{\mathbf{x} \cdot \mathbf{l}_{q'} | M'\mathbf{x} \geqq \mathbf{l}_p, \mathbf{x} \in Z_+^{q'}\} = \chi(G) \geqq \chi_\lambda(G) + 3. \qquad \square
\end{aligned}$$

Should the matrix M happen to have duplicate columns, it should be replaced by the matrix $\tilde{M}$ obtained from M by deleting all such duplicates. It is easily verified that by adding the appropriate components of any vector $\mathbf{x}$ satisfying $M\mathbf{x} \geqq \mathbf{l}_p$, one obtains a vector $\tilde{\mathbf{x}}$ satisfying $\tilde{M}\tilde{\mathbf{x}} \geqq \mathbf{l}_p$. since the object function is $\mathbf{x} \cdot \mathbf{l}_p$, an optimizing vector for the integer program determined by $\tilde{M}$ is immediately converted to one for the integer program determined by M.

Given an arbitrary integer program $\min\{\mathbf{x} \cdot \mathbf{c} | M\mathbf{x} \geqq \mathbf{b}, \mathbf{x} \in Z_+^m\}$ there are standard procedures for converting it to one in which all the entries of M, $\mathbf{b}$, and $\mathbf{c}$ are 0 or 1. This is done at the cost of increasing the number of variables. We therefore point out the following procedure which can provide useful information without adding variables. The proof depends on concepts and results from [2], [3]. For any positive integer k, a k-tuple coloring of G assigns to each vertex of G a set of k colors so that adjacent vertices receive disjoint sets. The kth chromatic number $\chi_k(G)$ is the minimum number of colors needed to provide G with a k-tuple coloring. It is known [2], [3] that if G has at least one edge then $\chi_k(G) \geqq 2 + \chi_{k-1}(G)$ and hence $\chi_k(G) \geqq 2k - 2 + \chi(G)$.

THEOREM 3. *Suppose* $\mathbf{b} = (b_1, b_2, \cdots, b_p)$, $k = \min\{b_1, \ldots, b_p\} \geqq 1$. *Let* M *be a* 0, 1 $p \times q$ *matrix some two of whose rows are orthogonal, and having all distinct columns. Then, if* $G = g(M)$,

$$\min\{\mathbf{x} \cdot \mathbf{l}_q | M\mathbf{x} \geqq \mathbf{b}, \mathbf{x} \in Z_+^p\} \geqq 2k + 1 + \chi_\lambda(G).$$

Proof. Let $\mathbf{k}$ be the vector all of whose components are k. Then

$$\begin{aligned}
&\min\{\mathbf{x} \cdot \mathbf{l}_q | M\mathbf{x} \geqq \mathbf{b}, \mathbf{x} \in Z_+^p\} \\
&\quad \geqq \min\{\mathbf{x} \cdot \mathbf{l}_q | M\mathbf{x} \geqq \mathbf{k}, \mathbf{x} \in Z_+^p\} \\
&\quad = \chi_k(G) \geqq 2k - 2 + \chi(G) \geqq 2k - 2 + \chi_\lambda(G) + 3. \qquad \square
\end{aligned}$$

REFERENCES

[1] L. LOVÁSZ, *Kneser's conjecture, chromatic number, and homotopy*, J. Combin. Theory (A), 25 (1978), pp. 319-324.

[2] S. STAHL, *n-tuple colorings and associated graphs*, J. Combin. Theory (B), 20 (1976), pp. 185-203.

[3] ———, *Reductions of n-fold covers*, Proc. Amer. Math. Soc., 72 (1978), pp. 422-424.

[4] J. W. WALKER, *From graphs to ortholattices and equivariant maps*, J. Combin. Theory (B), to appear.

[5] A. W. WRIGHT, *Graph coloring and homology*, submitted for publication.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 4, October 1985

GRAPH THEORY AND FLUID DYNAMICS*

KARL GUSTAFSON† AND ROBERT HARTMAN‡

Abstract. We describe recent applications of network-theoretic graph theory to the analysis of certain discretizations of fluid flow. Also given are a natural extension to elliptic equations of divergence form and a computation of a sparseness matrix previously inaccessible due to lack of a general basis method.

AMS(MOS) subject classifications. 65N30, 68E10, 35Q10, 76D05

1. Introduction. Consider a viscous incompressible liquid in a vessel Ω in Euclidean space. Let the velocity and pressure be given by $\mathbf{u}(x, t)$ and $p(x, t)$, where a scale has been chosen so that the density equals one. Then the *Navier-Stokes equations* describing the motion of the liquid are (in three dimensions):

$$u_t - \gamma \Delta u + \sum_{k=1}^{3} u_k \mathbf{u}_{x_k} = -\text{grad } p + \mathbf{f}, \qquad x \in \Omega, \quad t > 0, \tag{1.1}$$

$$\text{div } \mathbf{u} = 0, \qquad x \in \Omega, \quad t > 0, \tag{1.2}$$

$$\mathbf{u}(x, t) = 0, \qquad x \in \partial\Omega, \quad t > 0, \tag{1.3}$$

$$\mathbf{u}(x, 0) = \mathbf{a}(x) \qquad x \in \Omega, \quad t = 0. \tag{1.4}$$

Equations (1.1) and (1.2) are coupled partial differential equations called the momentum equation and continuity equation. The boundary condition (1.3) is called the noslip condition, (1.4) is the known initial condition, $\mathbf{f}$ represents known external forces, $\gamma = 1/R$ is the viscosity, R is the Reynolds number.

It is generally accepted that for smooth known $\mathbf{f}$ and $\mathbf{a}$ the problem (1.1)-(1.4) is *well-posed* for all $t \geqq 0$. This means that the equations possess a unique solution stable under small changes of data and regular for all time. This has been shown for two dimensions and has been a longstanding open problem, which we will not discuss, in three dimensions. We shall instead look at the construction of approximate solutions following [1], [2], [3], [4].

The French school, of which [2], [3] are a part, over the last ten years has pushed ahead with a two- and three-dimensional finite element method (FEM). This has then been employed in fluids problems, in potential flow methods for airframe design, and elsewhere. They have concentrated on accuracy and flexibility to do arbitrary three-dimensional configurations (e.g., airplane contours). For such irregular regions the FEM has many advantages, especially when one has adequate computational resource times. There appears to have been excellent cooperation between mathematicians, engineers, and programmers, and between the Paris universities, the government, and their aviation industry.

In 1979-81 we looked at [2] and found an important gap in an otherwise superb treatment: finite element bases had been shown to exist, and found in some cases, but generally their construction, in some cases even their dimension, was not known. In

* Received by the editors May 2, 1983, and in revised form May 15, 1984. A preliminary version of this paper was presented at the SIAM Conference on Applied Linear Algebra, Raleigh, North Carolina, April 26-29, 1982. This research was partially supported by the National Science Foundation under grant NSF MCS 80-12220-A2.

† Department of Mathematics, University of Colorado, Boulder, Colorado 80309.

‡ Department of Mathematics, Texas Tech. University, Lubbock, Texas 79409.

[1], [4] we resolved those questions, employing graph theory to provide a unified treatment.

Briefly:

1. Equation (1.2), the divergence-free condition of mass conservation, is seen to be of fundamental importance.

2. Temam [2] analyzes six finite element schemes satisfying discrete forms of equation (1.2) and boundary condition (1.3):

APX1: Finite differences in R^n,

APX2: Finite elements, quadratic polynomials, R^2,

APX2′: APX2 + cubic perturbations, R^2,

APX3: Finite elements, cubic polynomials, R^3,

APX4: Stream-function approximated by quintics, R^2.

APX5: Nonconforming piecewise linear fits, for R^n.

3. The momentum equation (1.1) and initial condition (1.4) are then solved numerically within the approximating subspaces APXi.

In this paper we describe the problem of finding the dimensions and bases of the APXi of Step 2 above. Because their construction and properties are somewhat complicated, occupying for example a good part of [2], we shall use freely the details of [1], [2] when needed. In particular, this paper may be viewed as augmenting and extending [1], with the following two goals:

(a) Describe, in § 2, the next section, the recent applications of graph and network theory to finite difference and finite element discretizations of fluid flow, with emphasis on the analysis of the algebraic and combinatorial questions resulting from those discretizations. Previous analytic treatments of the Navier-Stokes equations generally bypassed such considerations. In numerical solution their resolution is essential. We discuss not only our methods from [1] but also indicate and compare those arrived at recently and independently, by Amit, Hall and Porsching [5], and Hecht [6].

(b) Present, in § 3, the last section, further results:

(i) The application of these methods to equations of divergence form. This possibility was mentioned in Remark 4.3 of [1]. The idea is that, given a partial differential equation of the form $Lu = \text{div}\,(a(x)\,\text{grad}\,u) = 0$ with appropriate boundary conditions, e.g. Dirichlet or Newmann boundary data, one may adapt the graph-theoretic analysis of div $v = 0$ to the case $v = a(x)\,\text{grad}\,u$.

(ii) A calculation of the sparseness matrix for APX2. This responds to a point raised by Temam [2, p. 138]. For the examples run the linear systems found were quite sparse.

2. Graph theory and fluid dynamics. We will restrict attention to the schemes APX1, APX2, APX5 from [2] treated in [1], [4], [6] and to the K − L scheme of [7] treated in [5]. Other FEM schemes may be treated similarly, see [1], and indeed it would appear that such graph-theoretic methods would have important application to two- and three-dimensional finite element methods in general. This point, for finite difference methods, has already been made in [5].

For a two-component vector field $\mathbf{u} = (u_1(x_1, x_2), u_2(x_1, x_2))$ on a given domain Ω, the APX1 scheme approximates div $\mathbf{u}$ by forward differences on a grid superimposed on Ω. By forward we mean a right-up convention.

Example 2.1. APX1 *approximation.* Let $\mathbf{u}(x_1, x_2) = (x_1, -x_2)$ and $\mathbf{u}_h = (u_{1h}(m, n), u_{2h}(m, n))$ be its APX1 step function approximant on a unit mesh ($h = 1$, $k = 1$) square with center (m, n). The given vector field $\mathbf{u}$ is clearly divergence-free and we wish its discrete approximant to also be discretely so. The discrete divergence operator

at a point p is:

$$\operatorname{div}_h \mathbf{u}_h = \sum_{i=1}^{2} (u_i(p+\varepsilon_i) - u_i(p))$$

where the ε_i denote unit vectors so that div_h is the sum of the forward differences. Let

$$u_{1h}(m, n) = \int_{n-1/2}^{n+1/2} u_1(m-\tfrac{1}{2}, x_2)\, dx_2 = m-\tfrac{1}{2},$$

$$u_{2h}(m, n) = \int_{m-1/2}^{m+1/2} u_2(x_1, n-\tfrac{1}{2})\, dx_1 = -n+\tfrac{1}{2}.$$

That is, we define $\mathbf{u}_h$ to be facial averages of $\mathbf{u}$. It is easily seen that $\mathbf{u}_h$ represents a step function approximation to $\mathbf{u}$ from below. Checking the discrete divergence, we note

$$\nabla_{1h} u_{1h}(m, n) = (m+\tfrac{1}{2}) - (m-\tfrac{1}{2}) = 1,$$

$$\nabla_{2h} u_{2h}(m, n) = (-n-\tfrac{1}{2}) - (-n+\tfrac{1}{2}) = -1,$$

so that $\operatorname{div}_h \mathbf{u}_h = 0$.

For APX1 on a grid $X \subset Z^n$ (n-dimensional integers), let $U(X)$ be the vector space of vector functions $U: Z^n \to R^n$ such that $u = 0$ outside of X and such that $\operatorname{div}_h \mathbf{u} = 0$ for every $p \in Z^n$. Interpret $u_i(p)$ as fluid flow through a tube from node $p - \varepsilon_i$ to node p. The divergence-free condition at each node point p requires that flow in = flow out. This corresponds to Kirchhoff's second law for currents subject to no sources.

Let the set of nodes (graph theory: vertices), tubes (graph theory: edges), and σ, where $\sigma(t, p) = -1, +1, 0$ if flow in t is directed out of p, into p, or not at all, be interpreted as a directed graph $(N(X), T(X), \sigma(X))$. Let $U(N, T, \sigma)$ be the vector space of all functions $u: T \to R$ such that

$$\sum_{t \in T} u(t)\sigma(t, p) = 0, \qquad p \in N.$$

It is well known from graph theory that the dimension of the flow space $U(N, T, \sigma)$ is $\bar{T} - \bar{N} + \bar{C}$, where $\bar{T}$ is the number of tubes (edges), $\bar{N}$ the number of nodes (vertices), $\bar{C}$ the number of components of the graph.

Example 2.2. APX1 *dimension.* Let $X = \{(0, 0), (1, 0), (2, 0), (2, 1), (2, 2), (1, 2), (0, 2), (0, 1)\}$. The vertices and directed arcs of the corresponding directed graph are shown in Fig. 1 where "•" denotes a vertex from X and "∘" denotes a virtual vertex added to complete the differencing scheme APX1.

The equations corresponding to the divergence-free condition in the APX1 formulation are

$$-u_1 = 0, \qquad u_7 + u_9 - u_{11} = 0,$$

$$-u_2 = 0, \qquad -u_{10} - u_{12} = 0,$$

$$-u_3 = 0, \qquad u_8 + u_{10} - u_{13} = 0,$$

$$-u_4 = 0, \qquad -u_{14} = 0$$

$$u_1 + u_4 - u_5 - u_7 = 0, \qquad u_{11} + u_{14} - u_{15} = 0,$$

$$u_2 + u_5 - u_6 = 0, \qquad u_{12} + u_{15} - u_{16} = 0,$$

$$u_3 + u_6 - u_8 = 0, \qquad u_{13} + u_{16} = 0.$$

$$-u_9 = 0,$$

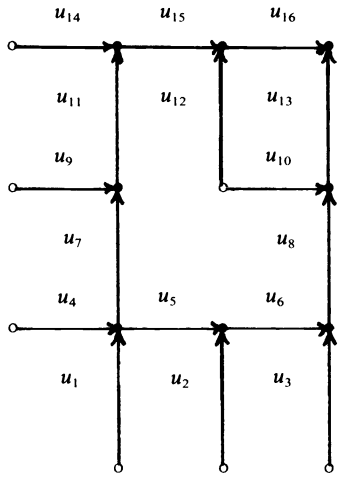


FIG. 1. *A flow network.*

These may be reduced directly to two degrees of freedom. On the other hand, from the graph theory, we see immediately that $\dim = \bar{T} - \bar{N} + \bar{C} = 16 - 15 + 1 = 2$.

Example 2.3. APX1 *basis.* The values of (u_1, u_2) at each node from Fig. 1 are indicated in Fig. 2. Note that the vector field basis vector shown in (a) corresponds to a large cycle, the other in (b) to a small cycle in the graph. We will return to this point later.

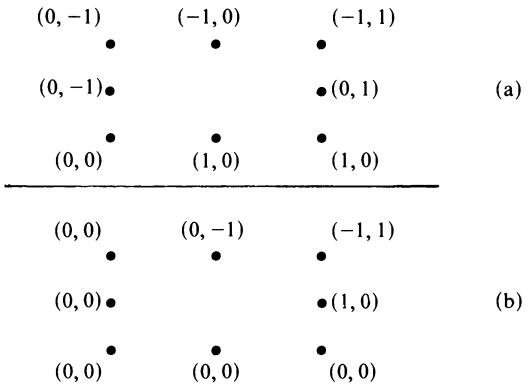


FIG. 2. *The basis vector field.*

APX2, utilizing quadratic vector field fits on triangulations of the given domain Ω, is a genuine FEM. We present some details of it next, in order to bring out the fluid dynamics decompositions used in connection with the graph theoretic concepts to enable the dimensions and basis of the divergence-free FEM subspaces.

First we want to make connection to the Helmholtz Decomposition Theorem: Any smooth vector field $\boldsymbol{F}$ on a smooth domain Ω in R^n may be decomposed as follows: $\boldsymbol{F} = \boldsymbol{F}_1 + \boldsymbol{F}_2 + \boldsymbol{F}_3$ where $\boldsymbol{F}_1$ is both divergence-free and curl-free, $\boldsymbol{F}_2$ is divergence-free but not curl-free, and $\boldsymbol{F}_3$ is curl-free but not divergence-free. In this way one may write more generally (see, e.g. [2]):

$$(L^2(\Omega))^n = H_1 \oplus H_2 \oplus H_3, \tag{2.1}$$

where the H_i are the subspaces corresponding to the F_i described above. The key to

our analysis of APX2 will be to first so decompose it, and then to apply graph theory to enable its dimension and basis construction.

To describe APX2 (for more details see [1], [2]), assume the given domain Ω in the plane is already triangulated. For example, suppose the boundary of Ω is polygonal. There are some conditions for an admissible triangulation (see [2]).

For each triangle τ in the triangulation T of Ω let W denote the space of all vector functions $\boldsymbol{\phi}: \Omega \to R^2$ such that $\boldsymbol{\phi} = (\phi_1, \phi_2)$ where ϕ_1 and ϕ_2 are polynomials of order less than or equal to 2 on each τ in T, and such that $\boldsymbol{\phi} = \mathbf{0}$ on the boundary $\partial\Omega$ of Ω. The latter condition is to enforce a Dirichlet boundary condition in the schemes of [2]. Let V denote those $\boldsymbol{\phi}$ in W which also satisfy the divergence-free condition

$$\int_\tau \operatorname{div} \boldsymbol{\phi}\, dA = 0 \quad \forall \tau \in T$$

and the "conforming basis" condition that $\boldsymbol{\phi} \in C^0(\Omega)$. This (weakly) divergence-free approximating space of quadratic polynomials is now decomposed into three subspaces V_i corresponding to the H_i above. The decomposition is, roughly, performed on the side, i.e., in one lower dimension than that of the original triangulation.

THEOREM [1].

$$V = V^1 \oplus V^2 \oplus V^3$$

and

$$\dim V = 2\,(\textit{the number of interior vertices}) + (\textit{the number of interior midedges}) + (\textit{the number of interior vertices} + \textit{the number of interior components of } R^2 - \Omega) = 3l + \bar{\bar{T}} + m,$$

where l = *number of interior vertices,* $\bar{\bar{T}}$ = *number of interior midedges,* m = *number of interior components of* $R^2 - \Omega$.

The theorem embodies the two steps we mentioned: decomposition, then graph theory. How does the latter come in? The analysis of the subspaces V^1 and V^2 do not really use it and may be described briefly as follows. In terms of barycentric coordinates for a single element τ as depicted in Fig. 3, and let us assume that τ is an interior element and therefore free of any specified boundary conditions, each component of ϕ is given by

$$\phi(x) = \sum_{i=1}^{3} (2\lambda_i^2 - \lambda_i)\phi(A_i) + 4 \sum_{i<j} \lambda_i \lambda_j \phi(A_{ij}).$$

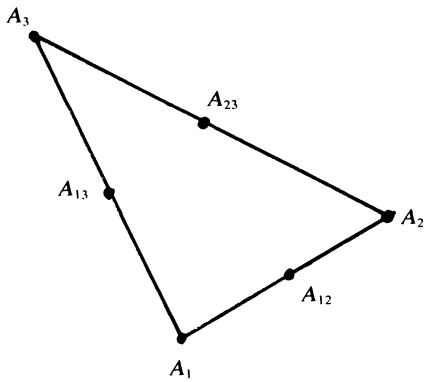


FIG. 3. *An element.*

The condition

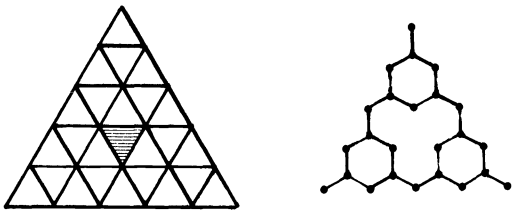
$$0 = \int_{\tau} \operatorname{div} \boldsymbol{\phi}\, dA = \int_{\partial\tau} \boldsymbol{\phi} \cdot \boldsymbol{\nu}\, dl$$

leads to the sufficient condition $\phi(A_i)+\phi(A_j)+4\phi(A_{ij})=0$ on each of the three edges of $\partial\tau$. From this, one may specify ϕ_1^1 at interior vertices, requiring midedge values to be minus one fourth of the sum of the two vertex values. Likewise for the second component, ϕ_2^1, and hence dim V^1 is twice the number of interior vertices, values of 0 being specified at boundary vertices to meet the Dirichlet boundary condition assumed in [2] and here.

The subspace V^1 is thus divergence free on sides of τ. It may be seen to be curl-free also. In a similar way one obtains V^2 by specifying $\boldsymbol{\phi}^2$ in terms of tangential values at midedges. Both $\boldsymbol{\phi}^2$ and $\boldsymbol{\phi}^3$ are taken to be 0 at all vertices in order to not interfere with the $\boldsymbol{\phi}^1$ specification. It turns out that $\boldsymbol{\phi}^2$ is divergence-free but not curl-free on an edge. Getting on to $\boldsymbol{\phi}^3$, again defined in terms of edge curl-freeness but now in the sum sense involving all three edges of τ, leads to the condition

$$|A_1 - A_2| d_{12} + |A_2 - A_3| d_{23} + |A_3 - A_1| d_{31} = 0 \tag{2.2}$$

for otherwise arbitrary values d_{ij}. It can be shown [1] that V^3 is isomorphic to the flow space $U(T,$ overlay edges, arbitrary). That is, we construct an overlay graph on the triangulation T of Ω by considering each $\tau \in T$ as a vertex, from its barycenter we construct an edge to the original midedge and on to the adjacent barycenter. See Fig. 4.



(a) (b)

FIG. 4. *Graph and dual.*

In graph theoretic terms the overlay graph construction is that of the graph inner dual. In this way it is seen that dim $V^3 =$ dim $U(T, E', \sigma') =$ the number of interior vertices + the number of interior components of $R^2 - \Omega = l + m$. Note also that dim V^3 is given by the number of small cycles plus the big cycle in Fig. 4(b). In the triangulation of Fig. 4 one has therefore

$$\dim V = 3 \cdot 3 + 27 + 1 = 37.$$

For the APX5 scheme the first subspace V^1 in the Helmholtz decomposition of the above Theorem vanishes and $V = V^2 \oplus V^3$. For the example triangulation of Fig. 4, dim $V = 31$ for the scheme APX5. Methods for basis construction for APX5 in two dimensions were known, see the discussions in [1], [2], [3]. Parallel to our graph-theoretic approach (cycle bases) for APX5 in any number of dimensions, Hecht [6] analyzed APX5 for three dimensions by using graph-theoretic results for maximal trees. These two approaches are equivalent although they would differ in implementation. Hecht [6] also gives an interesting application to Stokes flow in a cube. By cutting the unit cube into 27 smaller cubes and triangulating each of the smaller cubes into 5

tetrahedra, which gives 135 elements τ with 324 faces and 64 vertices, Hecht [6] obtains dim $V = 514$. Our methods in [1] are in agreement: by the formula of [1, Example 3.4.2] we have

$$\dim V = 25lmn - 6(lm + mn + nl) + 1 = 514 \tag{2.3}$$

by $l = m = n = 3$. Similar agreement between [6] and [1] for dim $V = 1313$ when $l = m = n = 4$ may be observed.

A full discussion of aspects of practical computational results is beyond these pages. See for example the beautiful results for an air-intake configuration in [3, Figs. 37-44], utilizing over 10,000 elements. These were obtained using a variant of APX5 in two dimensions in which each triangular element for pressure approximation is subdivided by midedge-joining into four subelements for velocity approximation. With the permission of F. Thomassett we reproduce in Fig. 5 an example of these complex results used in aerodynamic design.

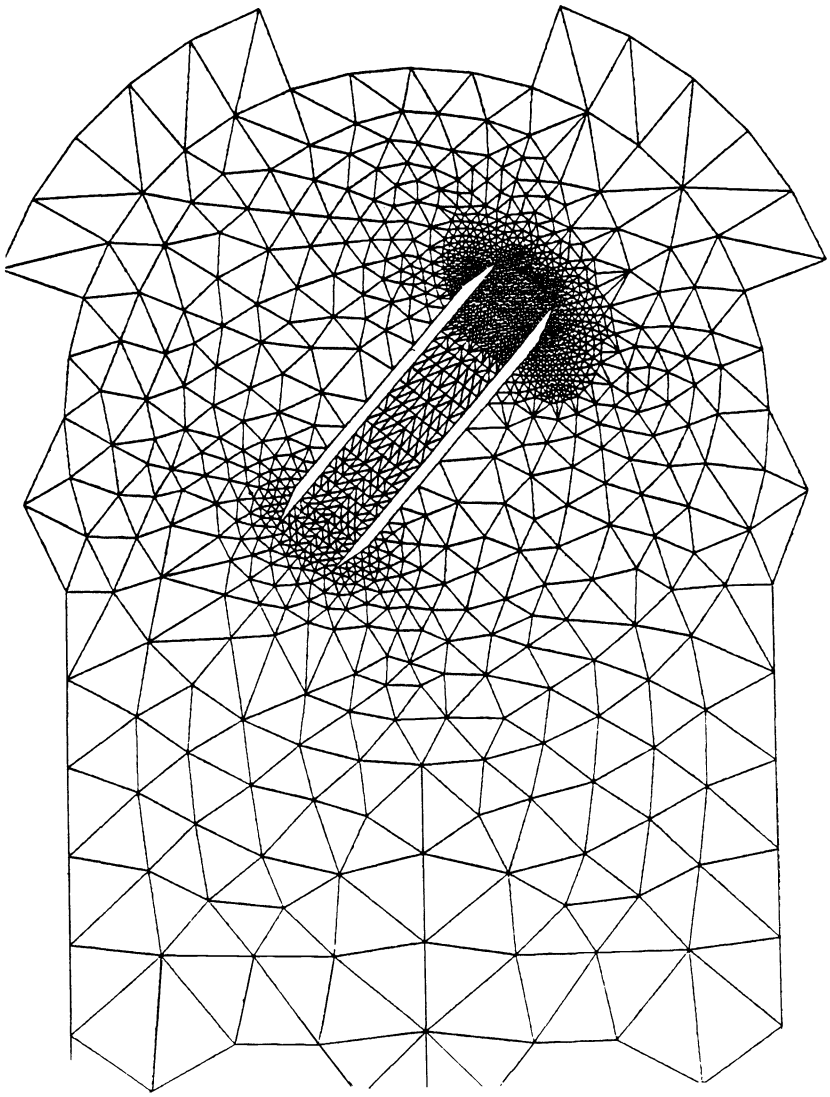


FIG. 5. *Discretized aerodynamic design.* (Taken from [3], used by permission.)

To complete this section, we turn to a third use of graph theory, independent of ours and of the French school, namely, that of Amit, Hall, and Porsching [5]. Their interest was quite different from that of [1] and [6] in that the latter were primarily concerned with the "algebra" of the various finite difference and finite element discretizations of the incompressibility condition (1.2) whereas in [5] a particular finite difference discretization of the momentum equation (1.1) was also under scrutiny, that of Krzhivitski and Ladyzhenskaya [7]. For comparison a MAC (marker and cell) finite difference scheme is also analyzed in [5], employing graph theoretic methods.

Referring the reader to [5] for full details, let us point out a few comparisions here.

Applying APX1 to the grid of [5, Fig. 1], (note that the original nodes are shown by ∘ in [5] in reverse of our convention in [1]) as depicted in Fig. 6 below, one finds from either [1] or [5] the dimension dim $V = 10-9+1 = 2$. Thus the K-L discretization of (1.2) employs the same right-up convention of APX1 and as far as the incompressibility constraint goes, they may be regarded as essentially the same.

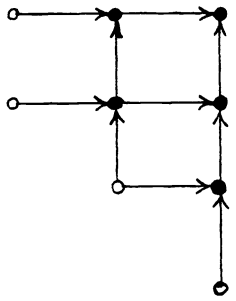


FIG. 6. *A 2-cycle grid.*

Secondly, the dual arguments of [5] go beyond those of [1] in the sense of application to the momentum equation. We will return to this point in § 3 in another context. On the other hand, [5] does not treat finite element methods.

Thirdly, we would like to comment on the rather good qualitative agreement between the two-dimensional cavity flow experiment of [5, see Fig. 9] as calculated in the dual variable formulation, and other cavity flow calculations using other scheme formulations. See for example the compendium of such result in [8, § 8, Driven Cavity Flow]. For the reader's convenience we show in Fig. 7 a result from [9], for Reynolds number 400 as in that of [5], from a MAC scheme. We would like to raise the question of whether the graph-theoretic methods can provide better schemes for the study of secondary and tertiary eddy development.

Finally, let us clarify the connection between the K-L discretization of the momentum equation in particular the nonlinear term, see [5, eq. (5)]

$$\frac{1}{2h}(S_x+I)(u_{ij}^{m-1}\nabla_x V_{ij}^m)+\frac{1}{2h}(S_y+I)(v_{ij}^{m-1}\nabla_y V_{ij}^m)$$

and Temam [2, Scheme 5.1, p. 334], in particular the nonlinear term

$$b_h(u_h^{m-1}, u_h^m, v_h), \qquad v_h \in V_h.$$

Scheme 5.1, although cast in the weak solution formulation, and the K-L scheme are essentially the same, are fully implicit and unconditionally stable. Thus the analysis of [5] may be viewed as reducing by use of graph theory from $O(3N)$ to $O(N)$ the number of equations to be solved in Scheme 5.1 at each time step.

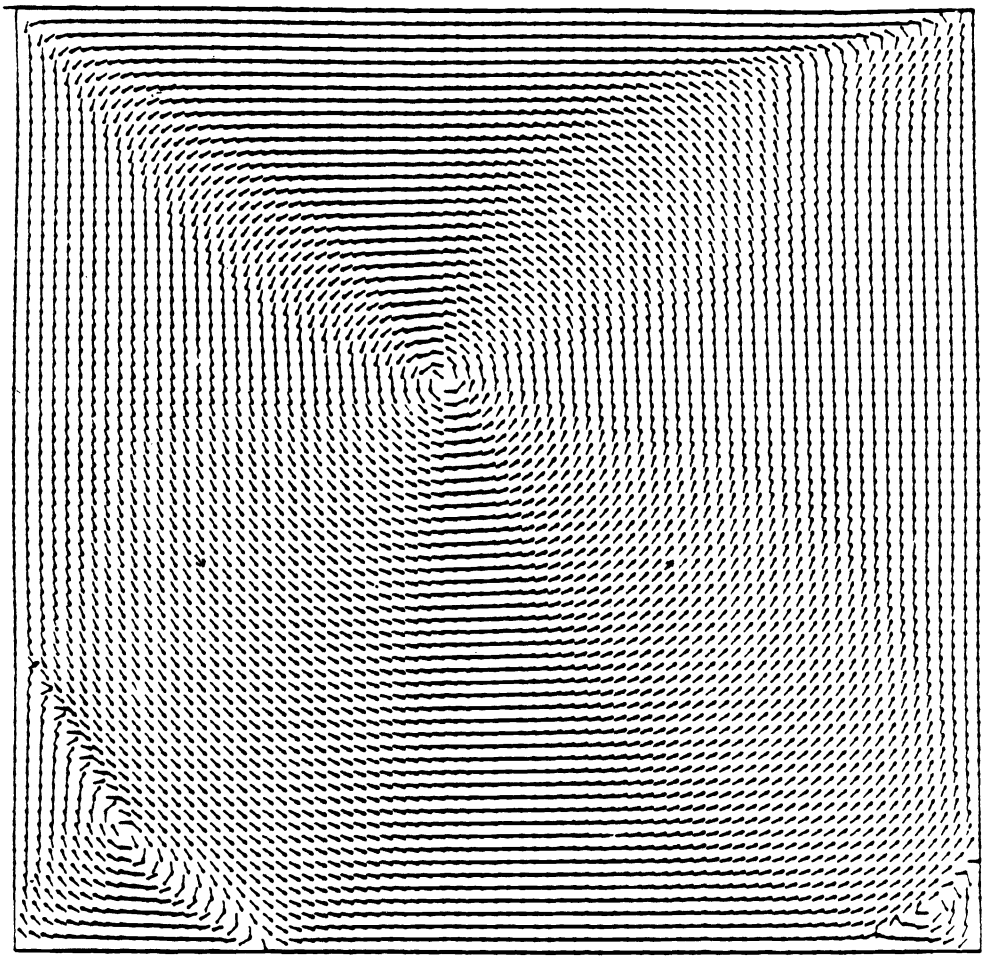


FIG. 7. *Normalized cavity flow graphs.*

3. Further results. In this section we:

(i) extend the analysis of div $v = 0$ to that of div (a grad u) $= 0$,

(ii) exhibit a sparseness matrix for the Scheme APX2.

These developments beyond [1] are to be regarded as preliminary, are included for illustration, and are not meant to be conclusive. However, we would like to mention a few particulars.

The graph-theoretic solutions of the finite difference problems in § 3.1 generally will yield positive definite symmetric sparse linear systems amenable to fast iterative linear methods. On the other hand, the number of equations is relatively large and the cycle base management could be costly. Finally, with all of the extensive recent and current work on fast Poisson solvers, it would take considerable testing to determine relative efficacy of these methods.

The sparseness matrix given in § 3.2 responds to a question of Temam [2, p. 138] and indicates good sparseness qualities for the linear systems resulting from APX2. However, the sparseness could be heavily domain and cycle base dependent.

3.1. Application to equations of divergence form. We restrict attention to Laplace's equation

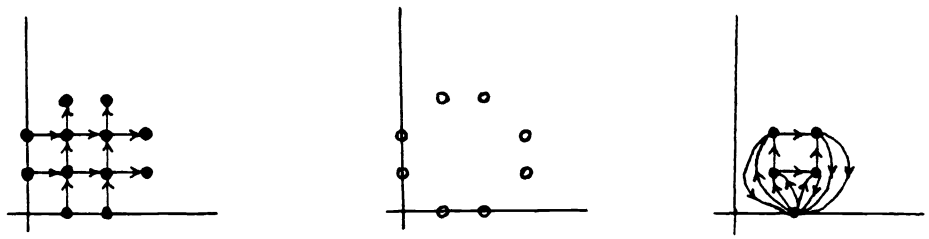
$$\Delta u = \operatorname{div}(\operatorname{grad} u) = 0 \tag{3.1.1}$$

but this approach can be extended to more general uniformly elliptic operators $Lu = \operatorname{div}(a \operatorname{grad} u)$ and with modification to any equation of divergence form. We mentioned this possibility in [1, Additional Remark 4.3]. It also is implicit, as we shall indicate, in the treatment of the K-L scheme as analyzed in [5]. Rather than attempting a general formulation for arbitrary domains and boundary conditions, we shall be content to illustrate the approach with application to a Dirichlet Problem.

Consider the discretized Dirichlet problem on the rectangle $\Omega = [0, 3] \times [0, 3]$:

$$\Delta u = 0 \quad \text{for } p \in X^0, \qquad u = g \quad \text{for } p \in \partial X^0, \tag{3.1.2}$$

where X is the stencil as shown in Fig. 8a, where X^0 denotes its interior (4 points), where g denotes the boundary values shown in Fig. 8(b), and where Δ denotes the standard finite difference Laplacian of centered differences. Let (N, E, σ) denote the directed graph obtained from the stencil X in Figure 8a by drawing linear edges directed up-right between the nodes of X, and let (N', E, σ') be the directed graph shown in Fig. 8(c) obtained from (N, E, σ) by retaining interior nodes and directions but identifying all boundary nodes of Fig. 8b. By this construction and our analysis



(a) (b) (c)

FIG. 8. *Stencils for Dirichlet problem.*

of APX1 in [1] the flow space $U(N', E, \sigma')$ of divergence-free vector fields has cycle basis

$$\begin{aligned}
c_1 &= (1,1)(2,1)(2,2)(1,2)(1,1),\\
c_2 &= (1,0)(1,1)(1,2)(2,2)(2,1)(2,0),\\
c_3 &= (1,0)(1,1)(1,2)(2,2)(2,1)(3,1),\\
c_4 &= (1,0)(1,1)(1,2)(2,2)(3,2),\\
c_5 &= (1,0)(1,1)(1,2)(2,2)(2,3),\\
c_6 &= (1,0)(1,1)(1,2)(1,3),\\
c_7 &= (1,0)(1,1)(1,2)(0,2),\\
c_8 &= (1,0)(1,1)(0,1),
\end{aligned} \tag{3.1.3}$$

where (x, y) denotes the node coordinates of X.

Let U denote the set of mappings $u: X^0 \cup \partial X^0 \to R^1$ for which the discretized equation $\Delta u = 0$ is satisfied for all $p \in X^0$. Let $\phi: U \to U(N', E, \sigma')$ according to the

rule $\phi(u(e)) = u(p) - u(p-\varepsilon_i)$, if e is an arc of X from $p-\varepsilon_i$ to p. Letting $v = \phi(u)$ we then have

$$\sum_E \sigma(e,p)v(e) = -\Delta u(p) = 0 \tag{3.1.4}$$

for all p in X^0, from which it follows that v is indeed in $U(N', E, \sigma')$.

The point here for further, more general treatment is that v is a discretized gradient of u, and as we mentioned earlier, the divergence of v is zero. That is, eq. (3.1.4) is an instance of div $(v) =$ div (grad u) $= 0$.

The vector v may be expressed as a linear combination $v = \sum_{j=1}^{8} b_j c_j$ in terms of the cycle basis of (3.13). Let $\langle w, v\rangle$ denote the inner product $\sum_{e\in E} w(e)v(e)$ on $U(N', E, \sigma')$. Note that if c is a cycle in (N', E, σ') beginning at node p_0 and ending at node p_k in X, then

$$\langle c, v\rangle = \sum_{e\in E} c(e)v(e) = \sum_{j=1}^{k} u(p_j) - u(p_{j-1}) = u(p_k) - u(p_0). \tag{3.1.5}$$

Thus we may solve for v from the matrix equation

$$[\langle c_i, c_j\rangle]b = [\langle c_1, v\rangle, \langle c_2, v\rangle, \cdots, \langle c_8, v\rangle]^T. \tag{3.1.6}$$

The matrix $[\langle c_i, c_j\rangle]$ will always be symmetric positive definite.

For a specific computation, consider the Dirichlet problem with boundary values as shown in Fig. 9. Then $\langle c_1, v\rangle = 0$, $\langle c_2, v\rangle = 5-3 = 2$, $\langle c_3, v\rangle = 3-3 = 0$, $\langle c_4, v\rangle = 1-3 = -2$, $\langle c_5, v\rangle = 3-3 = 0$, $\langle c_6, v\rangle = 1-3 = -2$, $\langle c_7, v\rangle = -1-3 = -4$, $\langle c_8, v\rangle = 1-3 = -2$, and

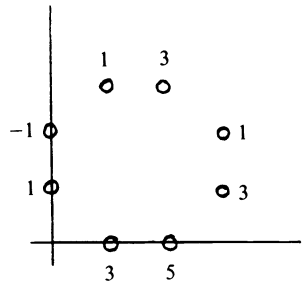


FIG. 9. *Boundary values.*

the cycle coefficients for v may be found from

$$\begin{bmatrix} 4 & 3 & 3 & 2 & 2 & 1 & 0 \\ 3 & 5 & 4 & 3 & 3 & 2 & 1 \\ 3 & 4 & 5 & 3 & 3 & 2 & 1 \\ 2 & 3 & 3 & 4 & 3 & 2 & 1 \\ 2 & 3 & 3 & 3 & 4 & 2 & 1 \\ 1 & 2 & 2 & 2 & 2 & 2 & 1 \\ 1 & 1 & 1 & 1 & 1 & 3 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 & 2 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \\ b_7 \\ b_8 \end{bmatrix} = \begin{bmatrix} 0 \\ 2 \\ 0 \\ -2 \\ 0 \\ -2 \\ -4 \\ -2 \end{bmatrix} \tag{3.1.7}$$

from which

$$b = (-1, 2, 0, -1, 1, 0, -2, -1)^T. \tag{3.1.8}$$

From the definition of v as the gradient of the solution u we then obtain

$$
\begin{aligned}
u(1,1) &= u(1,0) + \sum_{j=2}^{8} b_j = 2, \\
u(1,2) &= u(1,1) + \sum_{j=1}^{7} b_j = 1, \\
u(2,2) &= u(1,2) + \sum_{j=1}^{5} b_j = 2, \\
u(2,1) &= u(2,2) + \sum_{j=1}^{3} b_j = 3.
\end{aligned}
\tag{3.1.9}
$$

Neumann problems may be treated similarly. For the example above, one lets $v = \phi(u)$ so that the problem is formulated as $\Delta u = \operatorname{div} v = 0$. From $v = \sum_{j=1}^{8} b_j c_j$ in terms of the cycle base and the Neumann boundary condition with the form $v \cdot n = g(p_j)$ for Neumann data g at boundary nodes p_j one obtains $b_j = g(p_j)$ for all $j = 2, \cdots, 8$. From the condition that $\langle v, c_1 \rangle = 0$ one then finds $b_1 = -\sum_{j=2}^{8} g(p_j)\langle c_1, c_j \rangle / \langle c_1, c_1 \rangle$. One then returns the solution u from the difference equations which define v as the gradient of u.

Finally, we note that the grid of [5, Fig. 6] collapses to that of [5, Fig. 1], in a manner similar to that of Fig. 8. In both our approach and that of [5] this is necessary to accommodate the central differencing of second order operators.

For a more detailed presentation of the ideas in this subsection, see [10].

3.2. An APX2 sparseness calculation. A question beyond dimension and basis determination is that of the algebraic properties of the resultant system for use in Stokes or Navier-Stokes solvers. In particular, as pointed out by Temam [2, p. 138], one desires a sufficiently sparse matrix.

In Fig. 11, we give the sparseness matrix for the scheme APX2 on the triangulated Ω in Fig. 10. Recall that one is treating a Stokes problem

$$-\Delta u = -\nabla p + f \tag{3.2.1}$$

with zero boundary conditions by means of the variational formulation

$$a(u, v) = (f, v) \tag{3.2.2}$$

for all v that are divergence-free in Ω and vanish on the boundary of Ω. Here $a(u, v)$ is the Dirichlet form $\int_\Omega \nabla u \cdot \nabla v$ which for APX2 elements may be obtained piecewise by summing the contributions over the triangulation.

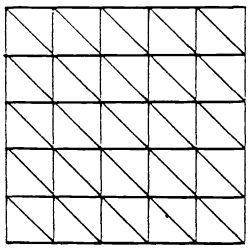


FIG. 10. *Triangulated 50-element domain.*

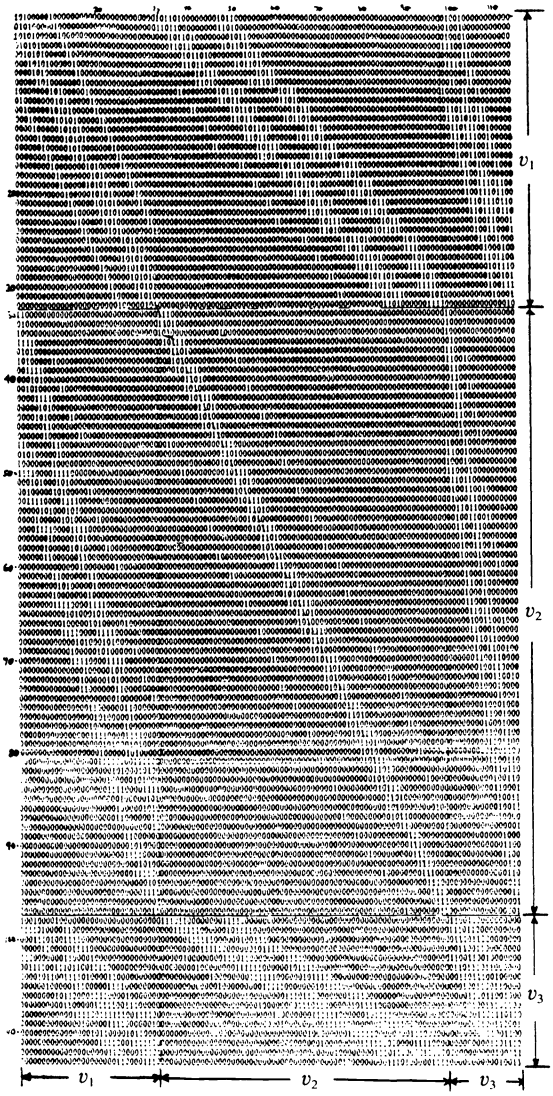


FIG. 11. *Sparseness matrix, with Helmholtz components indicated.*

The domain of Fig. 10 with 50 elements has from the analysis of [1] and our discussion in § 2 an APX2 approximating subspace V of form $V = V^1 \oplus V^2 \oplus V^3$ with dimension

$$\begin{aligned} \dim V &= \dim V^1 + \dim V^2 + \dim V^3 \\ &= (2)(16) + (65) + (16+0) = 113. \end{aligned} \tag{3.2.3}$$

In the sparseness matrix of Fig. 11, a_{ij} represents the Dirichlet form $a(c_i, c_j)$ with $a_{ij} = 1$ meaning a nonzero entry and $a_{ij} = 0$ meaning zero entry. This was computed on a Vax 11-780 and slight nonsymmetries are due to roundoff error. This and other examples show that the linear systems resulting from the APX2 scheme, although of relatively wide bandwidth, have good sparseness properties.

REFERENCES

[1] K. GUSTAFSON AND R. HARTMAN, *Divergence-free bases for finite element schemes in hydrodynamics*, SIAM J. Numer. Anal., 20 (1983), pp. 697-721.

[2] R. TEMAM, *Navier-Stokes Equations: Theory and Numerical Analysis*, 2nd ed., Elsevier-North-Holland, New York, 1979.

[3] F. THOMASSETT, *Implementation of Finite Element Methods for Navier-Stokes Equations*, Springer, New York, 1981.

[4] R. HARTMAN AND K. GUSTAFSON, *On the dimension of a finite difference approximation to divergence-free vectors*, in Quantum Mechanics in Mathematics, Chemistry, and Physics, K. Gustafson and W. Rinehardt, eds., Plenum Press, New York, 1981, pp. 125-131. See also N.A.M.S. 1 (1980), p. 196.

[5] R. AMIT, C. HALL AND T. PORSCHING, *An application of network theory to the solution of implicit Navier-Stokes difference equations*, J. Comp. Phys., 40 (1981), pp. 183-201.

[6] F. HECHT, *Construction d'une base de fonctions P_1 nonconforme à divergence nulle dans R^3*, RAIRO Anal. Numer., (1981), pp. 119-150.

[7] A. KRZHIVITSKI AND O. LADYZHENSKAYA, Proc. Math. Inst. Steklov, 92 (1966), p. 105; in Boundary Value Problems of Mathematical Physics IV, AMS Translation American Mathematical Society, Providence, RI, 1968.

[8] C. TAYLOR, J. JOHNSON AND W. SMITH, eds., *Numerical methods in laminar and turbulent flow*, Proc. III. International Conf., Seattle, August, 1983, Pineridge Press, Swansea, UK, 1981, pp. 553-666.

[9] K. GUSTAFSON AND K. HALASI, *Vortex dynamics of cavity flows*, to appear.

[10] R. L. HARTMAN, Ph.D. dissertation, Univ. Colorado, Boulder, CO, 1981.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 4, October 1985

013

EULERIAN ORIENTATIONS AND CIRCULATIONS*

GUAN MEIGU† AND WILLIAM PULLEYBLANK‡

Abstract. Let $G=(V, E)$ be an (undirected) eulerian graph. An *eulerian orientation* $\vec{G}$ of G is a directed graph obtained by giving each edge of G an orientation in such a way that $\vec{G}$ contains a directed eulerian tour. If each edge $[u, v]$ of G has real costs $c(u, v)$ and $c(v, u)$ associated with it, depending on which way it is oriented, then the cost of an eulerian orientation $\vec{G}$ is the sum of the costs of its arcs.

We show how the problem of finding a minimum cost eulerian orientation can be transformed into a minimum cost circulation problem. We also describe direct algorithms for the minimum cost eulerian orientation problem which can be viewed as specializations of general network flow algorithms for the transformed problem.

The orientation graph $\theta(G)$ has a node for every eulerian orientation, and the nodes corresponding to $\vec{G}$ and $\tilde{G}$ are adjacent in $\theta(G)$ if and only if $\vec{G}$ and $\tilde{G}$ differ only for the edges belonging to a simple cycle C of G. (Necessarily, the corresponding arcs will comprise directed cycles in $\vec{G}$ and $\tilde{G}$.) We show that $\theta(G)$ is either isomorphic to a d-dimensional hypercube for some d or else is hamilton connected (i.e. each pair of nodes is joined by a hamiltonian path of $\theta(G)$).

AMS(MOS) subject classifications. 05C45, 90B10

1. Introduction.

Let $G=(V, E)$ be an eulerian graph. An *orientation* of G is any directed graph $\vec{G}=(V, A)$ where A is obtained by replacing each edge with one of the two possible (directed) arcs joining the incident vertices. We say that an orientation is *eulerian* if it contains a directed Euler tour.

For each edge $[u, v] \in E$, the corresponding arcs (u, v) and (v, u) have real costs $c(u, v)$ and $c(v, u)$, which will generally be different. The *minimum cost eulerian orientation problem* is to find an eulerian orientation $\vec{G}$ of G for which the sum of the arc costs is minimized.

This problem arose when studying heuristics for the so-called *windy postman problem.* In this case we are given an arbitrary (connected) graph G and costs $c(u, v)$ and $c(v, u)$ associated with the possible orientations of each edge $[u, v]$ of G. A *postman tour* is a closed tour in G that traverses each edge at least once. The *cost* of the tour is the sum of the costs of the orientations used of the edges in the tour. Then the problem is to find a minimum cost postman tour.

This problem differs from the standard postman problem only in that the cost of traversing an edge depends on the direction of traversal. Guan [4] showed that the windy postman problem is NP-complete and proposed the following heuristic: first solve an undirected Chinese postman problem in order to determine a minimum cost subset of the edges to duplicate so as to obtain an eulerian graph, then solve the resulting minimum cost eulerian orientation problem.

In the next section we describe the relationship between the minimum cost eulerian orientation problem and a certain circulation problem. In § 3 we describe two algorithms for solving minimum cost eulerian orientation problems which can be viewed as specializations of a primal simplex algorithm and the out-of-kilter algorithm for this problem.

We define $\theta(G)$, the *eulerian orientation* graph of G, as follows: The node set of $\theta(G)$ consists of all the eulerian orientations of G. Two nodes of $\theta(G)$, i.e. two eulerian

* Received by the editors August 23, 1983, and in revised form May 31, 1984.

† Department of Combinatorics and Optimization, University of Waterloo, Waterloo, Ontario, Canada N2L 3G1. On leave from Shandong Teachers' University, China.

‡ Department of Combinatorics and Optimization, University of Waterloo, Waterloo, Ontario, Canada N2L 3G1. Research supported in part by the Natural Sciences and Engineering Research Council of Canada.

orientations $\vec{G}$ and $\tilde{G}$, are adjacent in $\theta(G)$ if and only if one can be obtained from the other by reversing the orientation of the arcs of a simple directed cycle. (Note that if we reverse the arcs of a directed cycle in any eulerian orientation, we obtain another eulerian orientation.) In § 4 we show that $\theta(G)$ is either isomorphic to a d-dimensional hypercube for some d or else it is hamilton connected i.e. every pair of distinct nodes is the pair of end nodes of a hamiltonian path. Thus, in particular, unless G has only two different eulerian orientations, i.e. unless G is a simple cycle, every edge of $\theta(G)$ will belong to a hamilton cycle.

Results of this sort appear frequently in the literature. (See [6] for a survey.) They establish classes of graphs which are relatively sparse and yet hamiltonian. They also show that it is possible to generate all members of a set of combinatorial objects once by starting with any object in the set and then applying a simple transition rule. In our case, we can start with any eulerian orientation $\vec{G}$ and produce each other eulerian orientation exactly once by a sequence of directed cycle reversals.

Frank [2] (see also [1]) considered a related problem. A *k-strong orientation* of $G=(V, E)$ is an orientation for which each $\emptyset \neq X \subsetneq V$ has at least k arcs directed in and k directed out. He showed that the problem of finding a minimum cost k-strong orientation could be solved as a special case of the more general problem of maximizing a linear function subject to a set of constraints given by a crossing family of sets and a submodular function. He also noted that some additional restrictions could be added to the orientations, e.g. fixing the indegree of every node. Clearly an orientation is eulerian if and only if it is a 1-strong orientation for which every node has indegree equal to one-half the degree in G. Our reductions of § 2 are similar to those of Frank and our algorithms of § 3 provide streamlined direct algorithms as alternatives to Frank's general approach.

In [3] Frank showed that each k-strong orientation of an undirected graph can be obtained from any other k-strong orientation by sequentially reversing the orientations of arcs belonging to directed paths and circuits, in such a way that all graphs produced are k-strong orientations. In other words, if $\vec{\theta}^k(G)$ is the graph whose nodes are the k-strong orientations of G and for which two nodes are adjacent if and only if the corresponding orientations differ only on the arcset of a directed path or cycle, then $\vec{\theta}^k(G)$ is connected. This is weaker than our result for $\theta(G)$ in § 4. We do not know whether or not it is possible to obtain hamiltonicity results for $\vec{\theta}^k(G)$.

2. Orientations and circulations. Throughout this section, $G=(V, E)$ is an eulerian graph.

For any directed graph $\vec{G}=(V, A)$, for any $v \in V$, we let $\delta^+(v)$ be the set of all arcs (u, v) and let $\delta^-(v)$ be the set of all arcs (v, u). Then it is well known that

(2.1) an orientation $\vec{G}$ of G is an eulerian orientation if and only if $|\delta^+(v)|=|\delta^-(v)|$ for all $v \in V$.

Let $\vec{G}=(V, \vec{A})$ be a fixed eulerian orientation of G. We construct a circulation problem on $\vec{G}$ by letting each arc j have a capacity of 1. Thus a *feasible circulation* is a vector $(x_j: j \in \vec{A})$ satisfying

$$\begin{aligned} &0 \leqq x_j \leqq 1 \quad \text{for all } j \in \vec{A}, \\ &\sum (x_j: j \in \delta^+(v)) - \sum (x_j: j \in \delta^-(v)) = 0 \quad \text{for all } v \in V. \end{aligned} \tag{2.2}$$

It is a fundamental result of network flow theory that if we let $C(\vec{G})$, the *circulation*

polytope, be the set of all solutions to (2.2), then

(2.3) $\hat{x}$ is a vertex of $C(\vec{G})$ if and only if $\hat{x}$ is a feasible circulation and $\hat{x}_j \in \{0, 1\}$ for all $j \in A$.

LEMMA 2.1. *Let $\tilde{G} = (V, \tilde{A})$ be any orientation of G. Then $\tilde{G}$ is an eulerian orientation if and only if the vector $\tilde{x} = (\tilde{x}_{(u,v)}: (u, v) \in \vec{A})$ defined by*

$$\tilde{x}_{(u,v)} = \begin{cases} 0 & \text{if } (u, v) \in \tilde{A}, \\ 1 & \text{if } (u, v) \notin \tilde{A} \end{cases}$$

is a vertex of $C(\vec{G})$, i.e., satisfies (2.2).

Proof. This follows easily from (2.1) and the observation that $\tilde{G}$ is an eulerian orientation if and only if for each $v \in V$ the orientations $\tilde{G}$ and $\vec{G}$ differ in the same number of arcs of $\delta^+(v)$ as of $\delta^-(v)$. These arcs j for which the orientations differ are precisely those for which $\tilde{x}_j = 1$. □

Recall that for each $[u, v] \in E$, $c(u, v)$ and $c(v, u)$ are the costs of the two possible orientations of $[u, v]$. For each arc $(u, v) \in \vec{A}$ we define $d(u, v) = c(v, u) - c(u, v)$, and for any orientation $\tilde{G} = (V, \tilde{A})$ of G we let $c(\tilde{G}) = \sum (c(u, v): (u, v) \in \tilde{A})$.

LEMMA 2.2. *Let $\tilde{G} = (V, \tilde{A})$ be an arbitrary eulerian orientation and let $\tilde{x} = (\tilde{x}_j: j \in \vec{A})$ be the corresponding circulation in $\vec{G}$ defined in Lemma* 2.1. *Then*

$$c(\tilde{G}) - c(\vec{G}) = \sum (\tilde{x}(u, v) d(u, v): (u, v) \in \vec{A}).$$

Proof. Immediate. □

Therefore finding a minimum cost orientation reduces to the problem of finding a minimum cost circulation. In the next section we show how this leads to the development of an efficient direct algorithm for the minimum cost orientation problem.

3. Finding minimum cost orientations. Let $\vec{G}$ be any orientation of G and let C be any directed cycle in $\vec{G}$. We define $c(C)$ to be the sum of the $c(u, v)$ for the arcs of C and let $\bar{C}$ be the directed cycle obtained from C by reversing the orientation of all its arcs. The following provides a useful optimality criterion.

THEOREM 3.1. *An eulerian orientation $\tilde{G} = (V, \tilde{A})$ of G is of minimum cost if and only if, for each directed cycle C of $\tilde{G}$, $c(C) \leqq c(\bar{C})$.*

Proof. The necessity is immediate, since reversing the orientation of all arcs of a directed cycle of an eulerian orientation yields another eulerian orientation. In order to prove the sufficiency, let $\tilde{x}$ be the corresponding circulation in $\vec{G}$ as defined in Lemma 2.1. It is a standard result of network flow theory that the arcs j of $\vec{G}$ for which $\tilde{x}_j = 1$ decompose into a set $\{C_1, C_2, \cdots, C_k\}$ of arc disjoint directed cycles in $\vec{G}$. Therefore $c(\tilde{G}) - c(\vec{G}) = \sum_{i=1}^{k} c(C_i) - \sum_{i=1}^{k} c(\bar{C}_i) = \sum_{i=1}^{k} (c(C_i) - c(\bar{C}_i))$. Therefore, if $c(\tilde{G}) < c(\vec{G})$ there must be at least one C_i such that $c(C_i) < c(\bar{C}_i)$. Since $\bar{C}_i$ is a directed cycle in $\tilde{G}$, the sufficiency follows. □

We can obtain a very simple algorithm for obtaining an eulerian orientation of minimum cost using this theorem.

ALGORITHM 1.

Step 1. Construct any eulerian orientation $\vec{G}$ of G.

Step 2. If every directed cycle C in $\vec{G}$ satisfies $c(C) \leqq c(\bar{C})$ then $\vec{G}$ is the optimum orientation; halt. Else let C be a directed cycle for which $c(C) > c(\bar{C})$ and go to Step 3.

Step 3. Reverse the orientation of all arcs of C and go to Step 2.

We can perform the test for a cycle C satisfying $c(C) > c(\bar{C})$ as follows. Again, define $d(u, v) = c(v, u) - c(u, v)$ for all arcs (u, v) of G. Then such a C exists if and only if $\vec{G}$ contains a directed cycle for which the sum of the $d(u, v)$ for its arcs is negative. This can be checked using a standard shortest path algorithm.

Each time we perform Step 3, the cost of the orientation strictly decreases. Therefore this algorithm is finite, since it can never repeat an eulerian orientation and the number of orientations is finite. However it may have to perform Step 3 a great many times before an optimum solution is found. We now describe a slightly more complicated algorithm which does, however, have an overall polynomial bound.

The correctness of our second algorithm is based upon the fact that the following two transformations of the costs do not affect which eulerian orientation will have minimum cost.

Transformation 1. *For some edge* $[u, v] \in E$, *add a constant value* θ *to both of* $c(u, v)$ *and* $c(v, u)$, *i.e. let*

$$c(v, u) := c(v, u) + \theta,$$

$$c(u, v) := c(u, v) + \theta.$$

Transformation 2. *Suppose* $X \subset V$ *and* $\bar{X} = V \setminus X$ *are both nonempty. For some constant* δ, *for all edges* $[u, v]$ *with* $u \in X$, $v \in \bar{X}$ *let*

$$c(u, v) := c(u, v) + \delta,$$

$$c(v, u) := c(v, u) - \delta.$$

Note that in fact Transformation 2 does not even change the value of the optimum solution.

By applying Transformation 1 we can ensure that c has the following properties:

(3.1) $$c(u, v) \geqq 0 \quad \text{for all } (u, v),$$

(3.2) for each $[u, v] \in E$, one of $c(u, v)$ and $c(v, u)$ has value zero.

We call a cost function *regular* if (3.1) and (3.2) hold.

Henceforth we assume that we are dealing with a regular cost function c. The algorithm starts with an arbitrary eulerian orientation $\vec{G} = (V, \vec{A})$. If $c(\vec{G}) = 0$ then (since $c \geqq 0$) $\vec{G}$ is of minimum cost. Otherwise, there exists an arc $(s, t) \in \vec{A}$ for which $c(s, t) > 0$. Since c is regular,

(3.3) $$c(t, s) = 0.$$

Define $A^* = \{(u, v) \in \vec{A}: c(v, u) = 0\}$. Necessarily any arc (u, v) of $\vec{A}$ for which $c(u, v) > 0$ will belong to A^*, but there may be others. Let X be the set of all nodes v of G for which there exists a directed path $\pi(v)$ in $\vec{G}$ from t to v all of whose arcs are in A^*.

If $s \in X$ then $\pi(s)$ together with the arc (s, t) forms a directed cycle in $\vec{G}$. If we reverse the orientation of the arcs of C we obtain a new eulerian orientation $\vec{G}'$ which (by (3.3)) contains at least one more arc (u, v) for which $c(u, v) = 0$ than does $\vec{G}$. If $c(\vec{G}') \neq 0$ then we replace $\vec{G}$ with $\vec{G}'$ and repeat the process.

If $s \notin X$ then

(3.4) $$t \in X, \qquad s \in V \setminus X,$$

(3.5) for all $(u, v) \in \vec{A}$ such that $u \in X$, $v \in V \setminus X$ we have $c(v, u) > 0$ and hence $c(u, v) = 0$.

Let $\delta = .5 \min \{c(v, u): (u, v) \in \vec{A},\ u \in X,\ v \in V \setminus X\}$ and let $(\bar{u}, \bar{v})$ be an arc for which this minimum is attained. Apply Transformation 2, let c' be the resulting costs. Then apply Transformation 1 for each $[u, v] \in E$ with $u \in X$, $v \in V \setminus X$ so as to obtain, once again, a regular cost function c''. Then

(3.6) for any $(u, v) \in \vec{A}$ such that $c(u, v) = 0$ we have $c''(u, v) = 0$.

For if $u \in X$, $v \in V \setminus X$ then $c(u, v) = 0$ (by (3.5)), $c'(u, v) = \delta$, $c'(v, u) = c(v, u) - \delta \geqq \delta$ and so $c''(u, v) = 0$. If $u \in V \setminus X$ and $v \in X$ then $c'(u, v) = -\delta$, $c'(v, u) = c(v, u) + \delta$ so $c''(u, v) = 0$. In all other cases $c''(u, v) = c'(u, v) = c(u, v)$.

Moreover, for any $(u, v) \in \vec{A}$ such that $u \in X$, $v \in V \setminus X$ we have $c''(v, u) = c(v, u) - 2\delta$ so, in addition,

(3.7) $$c''(\bar{v}, \bar{u}) = 0.$$

Finally, note that for $(u, v) \in \vec{A}$ with $u \in V \setminus X$, $v \in X$ such that $c(u, v) > 0$ we have $c''(u, v) < c(u, v)$. This applies, in particular, to the arc (s, t). If $c''(s, t) = 0$ then we have increased by at least one the number of arcs (u, v) of $\vec{A}$ having zero cost. Replace c by c'' and repeat the process for a new arc (s, t) with $c(s, t) > 0$, if such an arc exists. If $c''(s, t) > 0$, and hence $c''(t, s) = 0$, we can replace c by c'' and (3.3) still holds. Both the sets A^* and X will be strictly larger than before, since $(\bar{u}, \bar{v}) \in A^*$, $\bar{v} \in X$, and we now repeat the process.

We can summarize the algorithm as follows:

Step 0 [Initialization]. Let $\vec{G} = (V, \vec{A})$ be any eulerian orientation. Apply Transformation 1 so as to make c regular.

Step 1 [Optimality test]. If $c(u, v) = 0$ for all $(u, v) \in \vec{A}$ then $\vec{G}$ is optimal; halt. If not, choose $(s, t) \in \vec{A}$ such that $c(s, t) > 0$. Let t have the label "$\varnothing$", let $X = \{t\}$ and let $S = \varnothing$.

Step 2 [Labelling]. If $S = X$ then go to Step 4. Otherwise choose $u \in X \setminus S$. For every $v \in V \setminus X$ such that $(u, v) \in \vec{A}$, $c(v, u) = 0$ let $X := X \cup \{v\}$ and give v the label "u". Let $S := S \cup \{u\}$. If $s \in X$ then go to Step 3. Otherwise, return to Step 2.

Step 3 [Cycle reversal]. By using the labels on the nodes we can trace the directed path $\pi(s)$ in $\vec{G}$ from t to s such that $c(v, u) = 0$ for all arcs (u, v) of this path. Reverse the orientation of all arcs of $\pi(s)$ and of (s, t). Go to Step 1.

Step 4 [Cost transformation]. Compute $\delta = .5 \min \{c(v, u): u \in X,\ v \in V \setminus X, (u, v) \in \vec{A}\}$, let $(\bar{u}, \bar{v})$ be an arc giving the minimum. Apply Transformation 2 then apply Transformation 1 for each $[u, v] \in E$ such that $u \in X$, $v \in V \setminus X$ to obtain a regular cost function c''. If $c''(s, t) = 0$ then replace c with c'' and go to Step 1. If $c''(s, t) > 0$ then for each $(u, v) \in \vec{A}$ such that $u \in X$, $v \in V \setminus X$ and $c''(v, u) = 0$, add v to X and give v the label "u". Then go to Step 2. (Since $c''(\bar{v}, \bar{u}) = 0$ we will then have $\bar{v} \in X \setminus S$.)

The correctness of the algorithm is evident. Note that the set S is used as we construct the set X to hold the "scanned" nodes. We obtain a bound on the complexity as follows: Suppose that $\vec{G}$ has $m(\vec{G})$ arcs (u, v) with $c(u, v)$ positive (after regularizing c). We only go to Step 1 after decreasing the number of positive cost arcs of $\vec{G}$ by at least one, and we never increase this number. Hence Step 1 is performed at most $m(\vec{G}) + 1$ times. Each time we go to Step 2 we increase the size of X by 1. Hence this step can be performed at most $|V|$ times, for each execution of Step 1.

The total time spent in Step 1 will be no greater than the time required to scan the arcs of $\vec{G}$ once, i.e. $O(|E|)$. Similarly, using a depth-first search technique, Step 0 can be performed in time $O(|E|)$. If we maintain a stack or queue of nodes in $X \setminus S$, then the time for each execution of Step 2 is constant. The time for Step 3 is $O(|V|)$ and for Step 4 is $O(|E|)$. Therefore the total time for the algorithm is $O(m(\vec{G}) \cdot |V| \cdot |E|) = O(|V| \cdot |E|^2)$. Thus this algorithm is polynomial, as desired.

In fact this algorithm can be viewed as a translation of the out-of-kilter algorithm (see Lawler [5]) specialized to the equivalent circulation problem described in § 2.

4. Hamiltonicity of orientation graphs. A graph $G = (V, E)$ is *hamilton connected* if, for every distinct $u, v \in V$, there exists a hamilton path in G whose end nodes are u, v. There are two trivial hamilton connected graphs: K_1 and K_2—the complete graphs on one and two nodes respectively.

A *hypercube* can be defined inductively as follows: K_1 is a hypercube of dimension 0. For $d \geqq 1$, a d-dimensional hypercube is obtained by taking two disjoint copies of a $(d-1)$-dimensional hypercube and then joining all corresponding pairs of nodes. Hypercubes are bipartite and are "almost" hamilton connected in the sense that if u, v are distinct nodes belonging to the opposite parts of a hypercube, then there exists a hamilton path joining u and v. Hence, for $d \geqq 2$, every edge of a d-dimensional hypercube belongs to a hamilton cycle.

In this section we show that for any eulerian graph G, either $\theta(G)$ is hypercube or else $\theta(G)$ is hamilton connected. The basic result we use is a theorem of Naddef and Pulleyblank [7] (see also [6]) concerning 0-1 polytopes. The *graph* (or skeleton) $G(P)$ of a polytope P is defined as follows: The nodes of $G(P)$ are the vertices of P and two nodes of $G(P)$ are adjacent if and only if the corresponding vertices of P are adjacent on P, i.e. belong to a 1-dimensional face of P. (See [6].) Recall that distinct vertices u and v of P are adjacent if and only if, for every λ satisfying $0 < \lambda < 1$, the unique way that the point $\lambda u + (1-\lambda)v$ can be expressed as a convex combination of distinct vertices of P is as $\lambda u + (1-\lambda)v$.

THEOREM 4.1 (Naddef and Pulleyblank [7]). *Let P be a polytope all of whose vertices have 0-1 valued coordinates. Then $G(P)$ is a hypercube or is hamilton connected.*

We obtain results about $\theta(G)$ from Theorem 4.1 by again exploiting the relationship to circulations described in § 2. Recall that $C(\vec{G})$, the circulation polytope, was defined to be the solution set to the linear system (2.2). As we noted in (2.3), $C(\vec{G})$ is also the convex hull of the 0-1 valued circulation of $\vec{G}$.

We say that a vector y indexed by the arcs of $\vec{G}$ is a *cycle vector* if there exists an undirected simple cycle C in $\vec{G}$ such that

i) $y_j = 0$ for all arcs j not in C;

ii) if we choose the appropriate one of the two possible directions of travel around C, then $y_j > 0$ for all arcs j traversed in the forward direction and $y_j < 0$ for all arcs j traversed in the reverse direction.

In our present situation we are primarily interested in cycle vectors y having $y_j \in \{0, 1, -1\}$ for all arcs j. In this case, if we have a circulation x in $\vec{G}$ (taking all arc capacities to be one) and a cycle vector y such that $y_j = 1$ implies $x_j = 1$ and $y_j = -1$ implies $x_j = 0$, then $x - y$ will also be a circulation. Conversely, if $y_j = 1$ implies $x_j = 0$ and $y_j = -1$ implies $x_j = 1$, then $x + y$ will be a circulation.

LEMMA 4.2. *Let $\bar{x}$ and $\tilde{x}$ be distinct vertices of $C(\vec{G})$. Then $\bar{x}$ and $\tilde{x}$ are adjacent on $C(\vec{G})$ if and only if $\bar{x} - \tilde{x}$ is a cycle vector of $\vec{G}$.*

Proof. Suppose $\bar{x} - \tilde{x}$ is a cycle vector and for λ satisfying $0 < \lambda < 1$ let $x(\lambda) = \lambda\bar{x} + (1-\lambda)\tilde{x}$. Suppose there exists a finite set X of distinct vertices of $C(\vec{G})$ and

nonnegative reals $(\alpha_x: x \in X)$ such that

$$\sum(\alpha_x: x \in X) = 1,$$
$$\sum(\alpha_x x: x \in X) = x(\lambda).$$

Let C be the undirected cycle containing those edges j for which $\bar{x}_j - \tilde{x}_j \neq 0$. For every arc j not in C, we have $\bar{x}_j = \tilde{x}_j = x(\lambda)_j$, therefore we must also have $x_j = x(\lambda)_j$ for all $x \in X$. Therefore all vectors in X are identical, except for those components indexed by arcs of C. But then it is easy to check that the only way we can assign values to the components corresponding to arcs of C and obtain a 0-1 valued circulation is to have these components set to agree with either $\bar{x}$ or $\tilde{x}$. That is, they must all have value 1 for arcs oriented one direction and 0 for those oriented the other. Therefore $X = \{\bar{x}, \tilde{x}\}$ and so $\alpha_{\bar{x}} = \lambda$ and $\alpha_{\tilde{x}} = (1-\lambda)$ and so $\bar{x}$ and $\tilde{x}$ are adjacent.

Conversely, suppose that $\hat{x} = \bar{x} - \tilde{x}$ is not a cycle vector. Since $\bar{x} \neq \tilde{x}$ we have $\hat{x} \neq 0$ and since $\hat{x}$ is the difference of circulations we necessarily have

$$\sum(\hat{x}_j: j \in \delta^+(v)) - \sum(\hat{x}_j: j \in \delta^-(v)) = 0 \quad \text{for all } v \in V. \tag{4.1}$$

Choose any edge j for which $\hat{x}_j \neq 0$, let u be one end of j and construct an undirected path by performing the following process until some node is reached for a second time: Choose either an edge k having the same type of end (a head or a tail) incident with u as does j and for which $\hat{x}_k = -\hat{x}_j$ or else an edge k having the opposite type of end incident with u as does j and for which $\hat{x}_k = \hat{x}_j$. (This is always possible by (4.1) and because we stop as soon as the path is no longer simple.) Let $j := k$ and let u become the other end of k.

Because G is finite, the above procedure will terminate with a path containing a cycle C, and if we let x' be the vector equal to $\hat{x}$ on C, and zero elsewhere, then $x' \neq \hat{x}$ is a cycle vector. Therefore $\bar{x} - x'$ and $\tilde{x} + x'$ are both integer circulations distinct from $\bar{x}$ and $\tilde{x}$, and hence vertices of $C(G)$, and $.5\bar{x} + .5\tilde{x} = .5(\bar{x} - x') + .5(\tilde{x} + x')$ so $\bar{x}$ and $\tilde{x}$ are not adjacent on $C(G)$. □

It follows from Lemma 4.2 and Theorem 4.1 that the graph of $C(\vec{G})$ is either a hypercube or else hamilton connected. For present purposes however we are more interested in the following relationship.

LEMMA 4.3. *Let $\vec{G}$ be a fixed eulerian orientation of G. Eulerian orientations $\tilde{G}$ and $\bar{G}$ are adjacent in $\theta(G)$ if and only if the corresponding circulations $\tilde{x}$ and $\bar{x}$ (with respect to $\vec{G}$) are adjacent on $C(\vec{G})$.*

Proof. Orientations $\tilde{G}$ and $\bar{G}$ are adjacent in $\theta(G)$ if and only if they differ only in that a single directed cycle $\hat{C}$ of $\tilde{G}$ has the orientations of all its arcs reversed in $\bar{G}$. This holds if and only if $\tilde{x}$ and $\bar{x}$ differ only in components corresponding to arcs of $\hat{C}$ and the difference $\hat{x} = \tilde{x} - \bar{x}$ satisfies $\hat{x}_j = -1$ for all arcs j of $\hat{C}$ having the same orientation in $\tilde{G}$ as in $\vec{G}$ and $\hat{x}_j = +1$ for all arcs j of $\hat{C}$ having the opposite orientation. This holds if and only if, where we let C be the (undirected) cycle of $\vec{G}$ corresponding to $\hat{C}$, when we traverse C in the direction given by $\hat{C}$ we have $x_j < 0$ if the orientation of j agrees with the direction of traversal and $x_j > 0$ if the orientation disagrees. This holds if and only if $\bar{x} - \tilde{x}$ is a cycle vector which by Lemma 4.2 holds if and only if $\bar{x}$ and $\tilde{x}$ are adjacent. □

Now we can prove our final result.

THEOREM 4.4. *Let G be an eulerian graph. Then its orientation graph $\theta(G)$ is either a hypercube or else is hamilton connected.*

Proof. It follows from (2.3) and Lemma 4.3 that for any eulerian orientation $\vec{G}$ of G, the graph of $C(\vec{G})$ is isomorphic to $\theta(G)$. The result then follows from Theorem 4.1. □

In fact it is easy to see that $\theta(G)$ is nonbipartite if and only if G contains two distinct simple cycles C_1 and C_2 which have an edge in common. Consequently the only eulerian graphs G for which $\theta(G)$ is a hypercube are those consisting of d edge disjoint cycles (which may share cutnodes). In this case, $\theta(G)$ is a d-dimensional hypercube. In all other cases $\theta(G)$ is hamilton connected.

REFERENCES

[1] A. FRANK, *On the orientation of graphs*, J. Combin. Theory Ser. B, 28 (3) (1980), pp. 251-261.

[2] ———, *An algorithm for submodular functions on graphs*, Ann. Discrete Math., 16 (1982), pp. 97-120.

[3] ———, *A note on k-strongly connected orientations of an undirected graph*, Discrete Math., 39 (1982), pp. 103-104.

[4] M. GUAN, *On the windy postman problem*, CORR Report 83-6, Dept. Combinatorics and Optimization, Univ. Waterloo, Waterloo, Ontario, Canada.

[5] E. L. LAWLER, *Combinatorial Optimization: Networks and Matroids*, Holt, Rinehart and Winston, New York, 1976.

[6] D. NADDEF AND W. R. PULLEYBLANK, *Hamiltonicity and combinatorial polyhedra*, J. Combin. Theory Ser. B, 31 (3) (1981), pp. 297-312.

[7] ———, *Hamiltonicity in* 0-1 *polyhedra*, Research Report CORR 82-46, Dept. Combinatorics and Optimization, Univ. Waterloo, 1982, J. Combin. Theory Ser. B, to appear.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 4, October 1985

014

PROFILE SCHEDULING OF OPPOSING FORESTS AND LEVEL ORDERS*

DANNY DOLEV† AND MANFRED K. WARMUTH‡

Abstract. The question of existence of a schedule of a given length for n unit length tasks on m identical processors subject to precedence constraints is known to be NP-complete [Ullman, J. Comput. System Sci., 10 (1976), pp. 384–393]. For a *fixed value* of m we present polynomial algorithms to find an optimal schedule for two families of precedence graphs: level orders and opposing forests. In the case of opposing forest our algorithm is a considerable improvement over the algorithm presented in [Garey et al., SIAM J. Alg. Disc. Meth., 4 (1983), pp. 72–93].

1. Introduction. The goal of deterministic scheduling is to obtain efficient algorithms under the assumption that all the information about the tasks to be scheduled is known in advance [Co76], [GL79]. One of the fundamental problems in deterministic scheduling is to schedule a collection of n partially ordered, unit length tasks on a number of identical processors. As in [GJ83], [DW84a], [DW84b] we allow the number of identical processors to vary with time. This is described by a sequence of natural numbers, called a *profile* specifying how many processors are available at each unit of time (time slot). The *breadth* m, of a profile is an upper bound on the number of processors available at any time. A profile is *straight* if the number of available processors is the same at any time.

A *schedule* for a given profile is a partitioning of all the tasks into a sequence of sets which does not violate the precedence constraints and the number of tasks in each set does not exceed the number of available processors specified by the profile for the corresponding time slot.

Various aspects of scheduling theory have been extensively studied in recent years [GL79] and many scheduling problems are known to be NP-complete [GJ79]. The first NP-completeness result on scheduling with precedence constraints was published by Ullman [Ul75]. He showed that the existence of a schedule of a given length on a straight profile for a collection of unit length tasks subjected to precedence constraints is NP-complete in case where the breadth of the profile is a variable of the problem, that is, the breadth of the profile is not bounded by a constant. This problem remains NP-complete even for precedence graphs of special forms [GJ83], [Ma81], [Wa81].

Polynomial algorithms have been developed only for a few special cases of scheduling unit length tasks with precedence constraints. The first polynomial algorithm was developed by Hu [Hu61]. It produces an optimal schedule for a straight profile of arbitrary breadth if the precedence graph is either an inforest or an outforest. Hu's algorithm produces a schedule according to the *Highest Level First* (HLF) strategy, meaning tasks of higher level are chosen over tasks of lower level and among tasks of the same level ties are broken arbitrarily. Restricted versions of HLF provide optimal schedules if the precedence graph is an interval order [PY79], [Ga81], or if the number of available processors is two [FK71], [CG72], [Ga82].

The major scheduling problem remaining open is whether the scheduling of an arbitrary graph is NP-complete or polynomial for fixed number ($m \geqq 3$) of processors. In this paper we address two special cases of the above open problem. We utilize the

* Received by the editors August 2, 1982, and in revised form May 31, 1984.

† IBM Research Laboratory, San Jose, California 95193. Current address: Institute of Mathematics and Computer Science, Hebrew University, Jerusalem, Israel.

‡ Computer Science Department, University of California, Santa Cruz, California 95064.

results presented in [Wa81], [DW84a] to obtain polynomial algorithms for two families of precedence constraints (precedence graphs): level orders and opposing forests. A graph is a level order if each connected component is partitioned into some k levels $L_0, \cdots, L_{k-1}$ such that for every two tasks $x \in L_i$ and $y \in L_j$, where $i > j$, x precedes y. We present an algorithm for finding optimal schedules for this class that requires time and space $O(n^{m-1})$. An opposing forest [GJ83] is a graph composed of intrees and outtrees only. It is a generalization of the cases solvable by Hu's [Hu61] algorithm. Garey, et al., [GJ83] presented a polynomial algorithm for finding an optimal schedule in the case of opposing forest and straight profile of fixed breadth $m \geqq 3$. Their algorithm costs $O(n^{m^2+2m-5} \log n)$ time and $O(n)$ space. The algorithm we presented for this case is bounded by $O(n^{2m-2} \log n)$ time and $O(n^{m-1})$ space. For the special case $m = 3$ there exist a linear algorithms to find an optimal schedule [DW84a], [GJ83].

Our polynomial algorithms are based on the reduction theorem, which is proved in § 3. The reduction theorem is another form of the elite theorem [DW84a]. It reduces the number of components we have to consider at each step of the algorithm to at most $m-1$ (the highest ones) and therefore enables us to obtain efficient algorithms.

Notice that if the breadth of the profile is a variable of the problem rather than fixed, then scheduling a level order or an opposing forest becomes NP-complete [GJ83], [Ma81], [Wa81]. Thus our algorithms are expected to have a high complexity (exponential in the breadth m). A similar case was published in [DW84b]. It was shown that scheduling a precedence graph of bounded height on a profile of fixed breadth is polynomial. For profiles of arbitrary breadth the problem is again NP-complete [LR78], even if there is an arbitrary number of processors in only one time slot and one processor in all other slots [Wa81], [DW84a].

In § 2 we present the main notions used in the rest of the paper. Section 3 contains the reduction theorem. In §§ 4 and 5 we present the polynomial algorithm for level orders and opposing forests, respectively.

2. Basic definitions and properties.

2.1. Graph definitions. A (*precedence*) *graph* G is a directed acyclic graph given as a tuple (V, E), where V is the set of n *vertices* (or *tasks*) and E the set of *edges* of G. A (*directed*) *path* π of length r in a precedence graph $G = (V, E)$ is a sequence of vertices $x_0, \cdots, x_r$, such that the edge (x_i, x_{i+1}), for $0 \leqq i \leqq r-1$, is in E. A precedence graph G specifies the precedence constraints between the vertices (tasks) of G. We assume that if a task x has to be executed before a task y, then there exists a (directed) path of positive length from x to y in G, that is, x is a *predecessor* of y, and y is a *successor* of x. In the case where the longest path from a vertex x to a vertex y is the edge (x, y), x is an *immediate predecessor* of y and y is an *immediate successor* of x. Vertices x and y are *incomparable*, if x is neither a predecessor nor a successor of y. A set of vertices is incomparable if for any two vertices x and y of the set, x and y are incomparable, that is, there is no path between any two distinct vertices of the set.

By $h(G)$ we mean the *height* of G, which is the length of the longest path in G. For a vertex $x \in G$ (i.e., $x \in V$) we denote by $h(x)$ the length of the longest path that starts at x. A vertex with no successors has zero height. Vertices with identical height are said to be at the same *level*. Observe that all vertices of the same level are incomparable.

The graph G' is a (*closed*) *subgraph* of G if every vertex of G' has the same successors in G' as it has in G. A vertex of G, is *initial* if it has no predecessors. Note that an initial vertex of G is not necessarily of maximum height in G. A set of t *highest initial vertices* of G is a subset of initial vertices containing the t highest ones. Ties

are resolved arbitrarily. If there are less than t initial vertices then the set consists of all of them.

Let R be a set of initial vertices of G; then $G-R$ is the closed subgraph of G obtained by removing all the vertices of R from G. Given two graphs $G=(V, E)$ and $G'=(V', E')$, then $G \cup G'$ denotes the graph $(V \cup V', E \cup E')$. The graph $G=(V, E)$ is composed of $\{G_1, \cdots, G_r\}$ if these closed subgraphs (called *components* of G) are a decomposition of G into its connected components, that is each closed subgraph is a connected graph and there are no edges between vertices of different components; therefore, $G=\bigcup_i G_i$.

An *inforest* (respectively *outforest*) is a graph in which each vertex has at most one immediate successor (respectively one immediate predecessor). Notice that outforest is composed of components, each of which has exactly one initial vertex and it consists of this vertex and all its successors. A component of an outforest is called *outtree* and similarly a component of an inforest is called *intree*.

In a *level order* graph each component has the following form: Every vertex of level i precedes all vertices of the component from all the levels below i. Note that all vertices of the same component of a level order that are at the same level are isomorphic. Thus, we can assume that such a component is given as a tuple specifying how many vertices are in each level of the component.

2.2. Profile definitions. We partition the time scale into *time slots* of length one. The time interval $[i-1, i)$ for $i \geqq 1$ is the ith time slot. A *profile* is a sequence of positive integers specifying the number of identical processors that are available in each time slot. We shall interpret profile $M=(m_1, \cdots, m_d)$, where d is its length, to mean that for each slot i in $[0, d)$ there are m_i processors available.

The *breadth* of profile M is the upper bound on the number of processors that are available at any time slot of M. The profile of Table 2.1 has breadth 4. Throughout the paper we denote the breadth of the given profile with the letter m. We call a profile M *straight* if $m_i = m$, for all $1 \leqq i \leqq d$.

TABLE 2.1
A schedule for G fitting the profile $M=(2, 4, 2, 1, 1)$.

slot	1	2	3	4	5
P_1	4	1	3	6	
P_2	5	2	9		
P_3		7			
P_4		8			
m_i	2	4	2	1	1

2.3. Schedule definition. A *schedule* S for a precedence graph G is a sequence of sets $(S)_1, \cdots, (S)_k$ such that:

(i) the sets $(S)_i$, for $1 \leqq i \leqq k$, partition the vertices of G;

(ii) if $x \in (S)_i$ and $y \in (S)_j$, for $1 \leqq i \leqq j \leqq k$, then there is no path from y to x.

The *length* of a schedule $\lambda(S)$ is the index of the last nonempty set in the sequence. A minimal length schedule is called *optimal*. The schedule S *fits* the profile M if the length of S is not greater than the length of the profile and the cardinality of $(S)_i$ is not greater than m_i. The set of tasks $(S)_i$ get executed in the ith time slot, that is $|(S)_i|$ of the m_i processors of slot i each execute a task of $(S)_i$ during the time interval $[i-1, i)$. Note that all the tasks have unit length, which corresponds to the length of a time slot. An example is given in Fig. 2.1 and Table 2.1. The ith slot of S, $1 \leqq i \leqq \lambda(S)$,

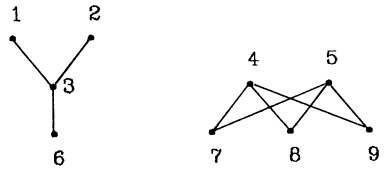


FIG. 2.1. *A precedence graph G.*

has $m_i - |(S)_i|$ *idle periods* meaning that there are this many processors idle during time slot i of S.

Given a precedence graph G and profile M, the initial problem is to determine if a schedule S exists for G and M. If a feasible schedule does exist, then we look for the shortest schedule S for G that fits M. In the first issue we allow the possibility that there does not exist a schedule for G that fits M. In the second we assume that there exists a feasible schedule and we are only interested in an optimal schedule.

A schedule S is an HLF-*schedule* for G and M if $(S)_i$, $1 \leqq i \leqq \lambda(S)$, is a set of m_i highest initial tasks of the closed subgraph of G induced by all tasks scheduled in slot i of S or later. HLF-schedules have the following property. Assume task x is scheduled in slot i and y is scheduled in slot j. If $h(x) > h(y)$, then either $i \leqq j$ or there is a predecessor of x in the jth slot. We say that HLF *produces an optimal schedule* if any HLF-schedule is optimal; that is, if an optimal schedule can be constructed by choosing higher initial tasks before lower ones and choosing arbitrarily among initial tasks of the same height. Note that the schedule of Table 2.1 is not a HLF-schedule; moreover, no HLF-schedule is optimal for G (Fig. 2.1) and the profile of Table 2.1.

2.4. The median. The following definition relates the number of components of a graph and the heights of the components with m; where m is the breadth of the profile.

DEFINITION. The *median* of precedence graph G with respect to a given m, denoted by $\mu(G)$, is one plus the height of some mth highest component of G. If the graph has less than m components, then the median is 0.

For example, if the precedence graph is the one in Fig. 2.2 and the breadth of the given profile is three, then the median is three because three is one plus the height of the third highest component. For the graph described by Fig. 2.1 the median is 0 with respect to $m = 3$.

We use the median to split the precedence graph G into two subgraphs. Let $G = H(G) \cup L(G)$, where the *high-graph* $H(G)$ contains all components of G that are strictly higher than the median; the *low-graph* $L(G)$ is the remaining subgraph of G. Note that $H(G)$ has at most $m-1$ components. Fig. 2.2 presents such a splitting of a precedence graph. We sometimes write $\mu(G, m)$, $H(G, m)$ and $L(G, m)$ to denote the median, the high-graph and the low-graph, respectively, for a specific m.

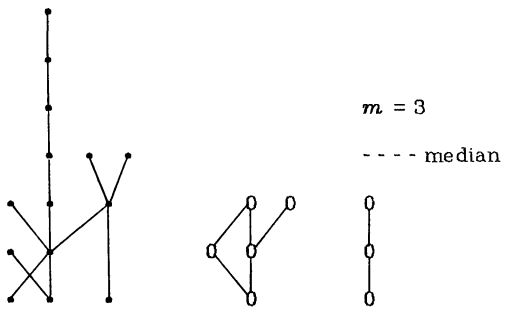


FIG. 2.2. *The decomposition of a graph G into $H(G)$ and $L(G)$; • denote vertices of $H(G)$ and 0 vertices of $L(G)$.*

The following properties of the median are used in the current paper.

PROPERTIES OF THE MEDIAN.

M1: There are at most $m-1$ components of G having height at least $\mu(G)$.

M2: If $\mu(G)>0$, then there are at least m components of G having height at least $\mu(G)-1$.

M3: If G has at most $m-1$ components of height at least h, then $\mu(G)\leqq h$.

M4: If G has at least m components of height at least $h-1$, then $\mu(G)\geqq h$.

The above properties follow directly from the definition of the median. Further properties of the median were given in [DW84a].

3. Reduction theorem. In this section we present our main result, the reduction theorem. We also prove several related theorems that are needed in later sections. The reduction theorem is a consequence of the MERGE Algorithm. The following lemma implies the correctness of the MERGE Algorithm. A component of a graph G is called *principal* if its height is at least $h(G)-1$.

LEMMA 3.1. *Let G be a graph and let G' be a subgraph of G obtained by removing a set of q highest initial tasks from G. Then G' contains at least as many principal components as the original graph G, unless $h(G')=0$.*

Proof. If the lemma holds for $q=1$, then it clearly holds for arbitrary q. Let x be a highest vertex of G, I be the principal component of G that contains x and $G'=G-\{x\}$. Assume $h(G')>0$. To show that G' contains at least as many principal components as G observe that $h(I)=h(G)>0$ and therefore $I-\{x\}$ contains a principal component of G'. Furthermore all principal components of G other than I are also principal components of G', because $h(G')\leqq h(G)$. We conclude that the number of principal components does not decrease when x is removed. □

The following algorithm shows how one can "merge" a schedule for a collection of subgraphs with a collection of subgraphs of lower height to get a schedule for the combined graph.

ALGORITHM 3.1. (the MERGE Algorithm)

Input: A graph $L=\cup_{i=1}^{r} L_i$, such that $h(L_i)\geqq h(L)-1$;
a graph $H=\cup_{i=1}^{q} H_i$, such that $h(H_i)>h(L)$, and $q+r\geqq m$;
a schedule S for H and M with p idle periods, where M is a profile of breadth m.

Output: A schedule S' for $H\cup L$ and M such that S' is not longer than the schedule S in the case where $p\geqq|L|$; and otherwise, S' is longer than S but has idle periods only in its last slot.

1. $k:=0$
   $S':=S$
2. **While** $h(L)>0$ **do**
   2.1. $k:=k+1$
   2.2. **While** $(S')_k$ is not full and not all initial vertices of H are scheduled in $(S')_k$ **do**
      2.2.1. Transfer an initial vertex of H from a slot after k to $(S')_k$.
   2.3. Fill $(S')_k$ with $m_k-|(S')_k|$ highest initial vertices of L.
   2.4. Remove the vertices of $(S')_k$ from L, H and its subgraphs H_i.
   2.5. **While** there is a subgraph H_i of H, such that $h(H_i)=h(L)$ **do**
      2.5.1. Transfer the graph H_i from H to L.
         $q:=q-1$; $r:=r+1$
      2.5.2. Remove the vertices of H_i from S'.

3. **While** L is nonempty **do**
   3.1. $k := k+1$
   3.2. **While** $(S')_k$ is not full or L is not empty **do**
      3.2.1. Add a vertex of L to slot k of S' and remove it from L.

A high level description of the MERGE *algorithm*. The aim of the algorithm is to "merge" the schedule S for H and M with the vertices of L producing a schedule S' for $H \cup L$ and M. The length of S' depends on the relationship between p, the number of idle periods in S and the number of vertices in L. If $p > |L|$, then there is enough "space" in S for all vertices of L and the resulting schedule S' is at most as long as S. Otherwise, S does not have enough idle periods and S' is longer than S. In this case, S' only has idle periods in its last slot.

At Step 1 of the algorithm we initialize S' with the schedule S for H and M. During Steps 2 and 3 the vertices of L are added into in S'. While doing so we sometimes reschedule vertices of H in S' (see Steps 2.2.1 and 2.5.2).

If $h(L) = 0$, then "merging" is easy (see Step 3). In this case, L is a set of single vertices. The algorithm consecutively fills the slots of S' with vertices of L until L is empty.

If $h(L) > 0$, then "merging" is slightly more involved (see Step 2). The variable q will be the number of subgraphs H_i that are left in H. All of these graphs will have height bigger than $h(L)$. If some of them drop down to height $h(L)$ during Step 2.4 then these subgraphs are transferred from H to L at Step 2.5. The variable r has the following meaning. During the algorithm it will be assumed that L has at least r principal components. The sum of q and r is at least m throughout the loop 2. This assures that there will be at least m initial vertices in $H \cup L$, at least q in H and at least r in L. We transfer components from H to L to avoid that some subgraphs H_i of H get completely scheduled and the sum of q and r drops below m.

Correctness of the MERGE *algorithm*: In the new schedule S' the precedence constraints specified by G are not violated, because we iteratively add vertices to S' (Steps 2.2.1, 2.3 and 3.2) that are initial in the unscheduled portion of $H \cup L$. Loop 2 has the following invariant: L has at least r principal components and H has q subgraphs H_i of height bigger than $h(L)$ and $q + r \geqq m$.

Note that by the definition of H, L, q and r the loop invariant trivially holds after Step 1. We want to show that if the loop invariant holds before Step 2.1 and $h(L)$ is bigger than zero then it holds after Step 2.5, or $h(L)$ equals zero.

At Step 2.4 only initial vertices are removed from H, H_i and L. Therefore, their height can drop at most by one. This assures that after Step 2.4 the graph H contains q subgraphs H_i of height at least $h(L)$. By Lemma 3.2 we know that after Step 2.4 either the graph L contains at least r principal components or $h(L) = 0$. Note that at Step 2.3 $(S')_k$ was filled with highest initial vertices of L. At Step 2.5 all subgraphs H_i of H that dropped down to height $h(L)$ are transferred from H to L. The height $h(L)$ and the sum $q + r$ does not change during Step 2.5. Furthermore, if before Step 2.5.1 L has at least r principal components then L has also at least r principal components after Step 2.5.1, since each H_i that is transferred contains at least one component of height $h(L)$. This completes the proof of the invariant of Loop 2.

The following claim completes the proof of correctness. It shows that if $p \geqq |L|$ then $\lambda(S') \leqq \lambda(S)$, and if $p < |L|$ then $\lambda(S') > \lambda(S)$ and S' has idle periods only in its last slot.

CLAIM 3.1.

(i) *After Step* 3 *the schedule* $(S')_1, \cdots, (S')_{k-1}$ *does not have any idle periods.*

(ii) *After Step* 3 *either* $k = \lambda(S')$ *or* $\lambda(S') \leqq \lambda(S)$.

(iii) *If* $\lambda(S') > \lambda(S)$ *then* S' *can have idle periods only in its last slot.*

(iv) $p \geqq |L|$ *if and only if* $\lambda(S') \leqq \lambda(S)$.

Proof of (i). By the loop invariant we know that the current slot is filled up in Steps 2.2 and 2.3. Thus, the schedule $(S')_1, \cdots, (S')_k$ does not have any idle periods when Step 3 is reached. At Step 3 all the slots, except may be the last one, are completely filled. This completes the proof of (i).

Proof of (ii). At Step 1 the schedule S' is initialized with S, and therefore $\lambda(S') = \lambda(S)$. During Steps 2 and 3 the algorithm never adds any vertices to any slot of S' with a higher index than the current slot k. On the other hand, in Steps 2.2.1 and 2.5.2 there are vertices removed out of slots with higher indices than the current slot k. This implies that after Step 3, $k = \lambda(S')$ or $\lambda(S') \leqq \lambda(S)$.

Proof of (iii). If $\lambda(S') > \lambda(S)$ then (ii) implies that $k = \lambda(S')$ after Step 3. Applying (i) we get that S' can have idle periods only in its last slot.

Proof of (iv). Assume $\lambda(S') > \lambda(S)$; then by (iii) we know that S' can have idle periods only in its last slot. In particular, there are no idle periods in slots 1 through $\lambda(S)$ of S', which implies that $\sum_{i=1}^{\lambda(S)} m_i < |H| + |L|$. Since p can be expressed as $(\sum_{i=1}^{\lambda(S)} m_i) - |H|$, it follows that $p < |L|$.

To prove the opposite direction of (iv) assume that $p < |L|$. Expressing p as $\sum_{i=1}^{\lambda(S)} m_i - |H|$ implies that $\sum_{i=1}^{\lambda(S)} m_i < |H| + |L|$. Since S' is a schedule for $H \cup L$, we have $|H| + |L| \leqq \sum_{i=1}^{\lambda(S')} m_i$. Combining both inequalities we get $\sum_{i=1}^{\lambda(S)} m_i < \sum_{i=1}^{\lambda(S')} m_i$, which implies $\lambda(S) < \lambda(S')$.

Herewith we completed the proof of the claim and the proof of correctness of the MERGE algorithm. □

The MERGE algorithm is linear even if G is not transitively reduced [AH74].

LEMMA 3.2 [Wa81]. *The* MERGE *algorithm can be implemented in time and space* $O(n+e)$, *where* n *is the number of vertices and* e *the number of edges in* $H \cup L$.

Proof. We only give a general idea of the implementation of the MERGE algorithm. A complete description appears in [Wa81]. We keep track of the set of current initial vertices of H and L; call these sets I_H and I_L, respectively. Whenever we remove vertices from these sets we add the vertices that become initial to the list.

In Step 2.2.1 we can choose any vertex of I_H that is not already in $(S')_k$. On the other hand, vertices of I_L should be scheduled according to their height (Step 2.3). Thus we need a data structure that will enable us to retrieve vertices from I_L efficiently. We represent I_L as an array of lists, where the entry $I_L(h)$ points to a linked list of all the initial vertices of height h (in arbitrary order); see [DW84a], [Wa81] for details. As shown in the proof of correctness there are always enough initial vertices in $I_L(h(L))$ and $I_L(h(L)-1)$ to fill $(S')_k$ in Step 2.3. Thus it is enough to pick vertices of the last and second to last nonempty list of I_L. This is the main reason for the fact that the MERGE algorithm can be implemented in $O(n+e)$ time. We do not have to do a complicated search to find highest vertices in Step 2.3.

For Step 2.5 we need to keep track of the heights of the subgraphs H_i. This is easy since during each iteration of the loop the height of a subgraph H_i can drop at most by one. To be able to transfer components easily we need to keep track of the vertices of each H_i and keep pointers from each vertex of G to all its occurrences in the data structures. This completes the summary of the proof. □

The reduction theorem is an immediate consequence of the following theorem, in which we apply the MERGE algorithm 3.1.

THEOREM 3.1. *Let* G *be a graph and* M *be a profile of breadth* m. *Given a schedule* S *for the high-graph of* G *and* M *that has* p *idle periods, then with the* MERGE *algorithm*

one can find a schedule S' for the whole graph G and M in time and space $O(n+e)$ that has the following form:

(i) *if $p \geqq |L(G)|$ then S' is at most as long as S*;

(ii) *if $p < |L(G)|$ then S' is longer than S and has idle periods only in its last slot.*

Proof. We run the MERGE algorithm on the following input parameters:

H is the high-graph and L the low-graph of G;

q is the number of components of $H(G)$ and $H_1, \cdots, H_q$ are the components of $H(G)$;

$r = m - q$ and $L_1, \cdots, L_{r-1}$ are some $r-1$ principal components of $L(G)$;

L_r is the remaining subgraph of $L(G)$ after removing $L_1, \cdots, L_{r-1}$.

Note that $h(H_i) > h(L)$, for $1 \leqq i \leqq q$, since H consists of all components of G that have height higher than the median of G, and L consists of all components which are at most as high as the median. By property M1 of the median we know that $H(G)$ has less than m components, and therefore $q < m$. Note that $H(G)$ might be empty and $q = 0$. Property M2 says that G has at least m components of height $\mu(G, m) - 1$. This implies that $L_1, \cdots, L_r$ exist and that $h(L_i) \geqq h(L(G)) - 1$, for $1 \leqq i \leqq r$. Note that $h(L(G)) \leqq \mu(G)$. It is easy to see that the input parameters can be found in time $O(n+e)$. Using Lemma 3.2 the proof is completed. □

We are now ready to present the main result of this section, the reduction theorem. It shows that finding an optimal schedule for G and M reduces to finding an optimal schedule for $H(G)$ and M.

THEOREM 3.2 (the reduction theorem). *Let G be a graph and M be a profile of breadth m. Then given an optimal schedule for the high-graph of G and M, the* MERGE *algorithm finds an optimal schedule for the whole graph G and M in time and space $O(n+e)$.*

Proof. Let S be the given optimal schedule for $H(G)$ and M. In Theorem 3.1 we showed that with the MERGE algorithm one can find a schedule S' for G and M in time and space $O(n+e)$ which has the following form:

(i) if $p \geqq |L(G)|$, then $\lambda(S') \leqq \lambda(S)$;

(ii) $p < |L(G)|$, then $\lambda(S') > \lambda(S)$ and S' has idle periods only in its last slot.

We want to show that the optimality of S for $H(G)$ and M implies the optimality of S' for G and M. Every schedule that has only idle periods in its last slot is optimal. Therefore, if $p < |L(G)|$ then S' is optimal. In the case $p \geqq |L(G)|$, S' is at most as long as S. An optimal schedule for G and M has to be at least as long as an optimal schedule for $H(G)$ and M, since $H(G)$ is a closed subgraph of G. Thus in the case $p \geqq |L(G)|$ we get $\lambda(S') = \lambda(S)$ and the optimality of S implies the optimality of S'. □

The following corollary of the reduction theorem implies that in the case where $H(G)$ is empty finding an optimal schedule is linear.

COROLLARY 3.1. *If $H(G)$ is empty then* HLF *is optimal for G and M and an* HLF *schedule can be found in time and space $O(n+e)$.*

Proof. The "empty schedule" is an optimal schedule for the empty graph $H(G)$ and M. The MERGE algorithm (applied as in Theorem 3.1) produces an arbitrary HLF schedule. Such a schedule has idle periods only in its last slot and is therefore optimal. □

The fact that HLF is optimal in the case where $H(G)$ is empty is also implied by the elite theorem of [DW84a].

The following theorem shows that the length of an optimal schedule is determined by the high-graph and the cardinality of the low-graph. The structure of the low-graph is not important.

THEOREM 3.3. *Let G and I be graphs such that $H(G, m) = H(I, m)$ and $|L(G, m)| =$*

$|L(I, m)|$. *Let M be a profile of breadth m. Then the optimal schedules for G and M and for I and M have the same length.*

Proof. Let S be some optimal schedule for $H(G) = H(I)$ and M. Let p be the number of idle periods in the schedule S. Note that all optimal schedules for $H(G) = H(I)$ and M have the same number of idle periods, since they have the same length and since they contain the same vertices.

In Theorem 3.1 we showed that with the MERGE algorithm one can find a schedule S' for G and M whose length only depends on the relationship between p and $|L(G)|$. In the same way we can find a schedule $\bar{S}$ for I and M by "merging" S with $L(I)$. Since S is a schedule for $H(G) = H(I)$ and since $|L(G)| = |L(I)|$ we have that $\lambda(S') = \lambda(\bar{S})$. In the reduction theorem we showed that both S' and $\bar{S}$ are optimal for G and I, respectively. This completes the proof of the theorem. □

In the following theorem we show which subsets of the set of initial vertices of a graph start an optimal schedule for this graph. Iterating this theorem we can find an optimal schedule for the whole graph. The elite theorem of [DW84a] is a stronger version of this theorem.

THEOREM 3.4. *Let G be a graph, M be a profile of breadth m and I be the set of initial vertices of $H(G, m)$. If there exists a schedule for G and M then:*

Case $|I| > m_1$. *There exists a set, R of m_1 vertices of I which starts some schedule for $H(G)$ and M, and for any such set R there exists a schedule for G and M starting with R.*

Case $|I| \leqq m_1$. *For any set T of $m_1 - |I|$ highest initial vertices of $L(G)$ there exists a schedule for G and M starting with $I \cup T$.*

Proof. We first show that if there exists a schedule for G and M, then there exists a schedule for $H(G)$ and M that has $\min(m_1, |I|)$ vertices in its first slot. Let S be a schedule for G and M. By removing the vertices of $L(G)$ from S we get a schedule $\bar{S}$ for $H(G)$ that fits M. Now, if the first slot of $\bar{S}$ has idle periods and not all vertices of I are scheduled in the first slot of $\bar{S}$, then we can move vertices of I from higher slots to the first slot of $\bar{S}$. We keep on doing this until either the first slot becomes filled up or all the vertices of I are scheduled in the first slot. The resulting schedule has the form we are looking for. It has $\min(m_1, |I|)$ vertices in its first slot.

Case $|I| > m_1$. Let S be a schedule for $H(G)$ and M starting with a set R of m_1 vertices of I. As shown above such a schedule always exists. We now "merge" S with $L(G)$ as done in Theorem 3.1. The schedule S' for G and M constructed by the MERGE algorithm starts also with R, since Steps 2.2 and 2.3 are redundant for $k = 1$. To see that the schedule S' for G fits the profile M we observe that there exists a schedule for G and M and therefore there is enough "space" in the profile M.

Case $|I| \leqq m_1$. Let S be a schedule for $H(G)$ and M starting with the set I. We showed already that such a schedule exists. Let T be a set of $m_1 - |I|$ highest initial vertices of $L(G)$. We again "merge" S with $L(G)$ as in Theorem 3.1. The MERGE algorithm constructs a schedule S' for G and M that starts with I and a set of $m_1 - |I|$ highest initial vertices of $L(G) = L$. Note that Step 2.2 is redundant for $k = 1$, since $(S)_1$ contains all initial vertices of $H(G)$. Assume the set T is chosen at Step 2.3 as a set of $m_1 - |I|$ highest initial vertices of $L(G)$. Then S' starts with $I \cup T$, which we wanted to show. □

4. Level orders. In this section, we present a polynomial algorithm for finding an optimal schedule in the case where the graph is a level order of q components (where q is a positive constant) and a profile of unbounded breadth m. Our algorithm runs in time $O(m^q n^q)$ and uses space $O(n^q)$.

By property M1 of the median we know that $H(G, m)$ has less than m components. Therefore, combining the $O(m^q n^q)$ algorithm with the reduction theorem 3.2 we get the following result: An optimal schedule for a graph G, such that $H(G, m)$ is a level order, and a profile of constant breadth $m \geqq 3$ can be found in time and space $O(n^{m-1})$.

Level orders are a proper subclass of the class of series parallel digraphs [TL79] which in turn is a proper subclass of the class of totally interacting digraphs [Go76]. In [Go76] it was shown that HLF produces an optimal schedule if the graph is totally interacting and the profile is straight and of breadth two. It is an easy exercise to see that this result holds for nonstraight profiles of breadth two. Note that HLF does not produce an optimal schedule if $m = 2$ and the graph is arbitrary. In this case, restricted forms of HLF produce an optimal schedule [CG72], [Ga80a].

HLF also produces an optimal schedule for a single level order component (in linear time). By applying the reduction theorem we obtain an $O(n+e)$ time bound for any graph whose high-graph consists of at most one level order component. On the other hand, neither HLF nor restricted HLF produce an optimal schedule even if the whole graph is a level order of two components and the profile is straight and of breadth three. In Fig. 4.1, we give an example to show this. An optimal schedule S for this graph and the straight profile of breadth three is: $\{11, 10', 9'\}$, $\{10, 8', 7'\}$, $\{9, 6', 5'\}$, $\{8, 7, 4'\}$, $\{6, 5, 3'\}$, $\{4, 3, 2'\}$, $\{2, 1, 1'\}$. Note that this schedule has no idle periods, while any HLF schedule will have idle periods in its second slot.

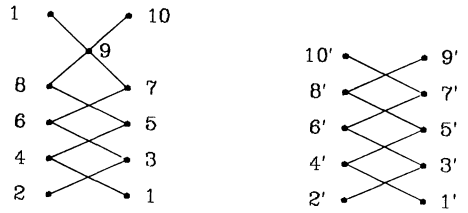


FIG. 4.1. HLF *is not optimal for three processors.*

To describe some special properties of level orders we use the following definitions: Given two graphs $G = (V, E)$ and $G' = (V', E')$. Then G is (*transitively*) *isomorphic* to G' if and only if there exists a bijective function $f: V \to V'$, such that for all vertices x and y of V, we have the following: x precedes y in G if and only if $f(x)$ precedes $f(y)$ in G'. Note that the fact that f is bijective implies that $|V| = |V'|$. Two vertices x and y of the same graph G are isomorphic to each other if and only if x maps into y in an isomorphism of G onto itself. Many closed subgraphs of a level order are isomorphic. This is the main reason why scheduling a constant number of level order components is polynomial. There will be only a polynomial number of possible closed subgraphs that we need to handle in the algorithm.

LEMMA 4.1. *Let G be a level order with one component. Then all closed subgraphs of G that have the same number of vertices are transitively isomorphic.*

Proof. We want to show that all closed subgraphs of G with k vertices ($k \leqq n$) are isomorphic. Let h be the maximum height such that G has less than k vertices of height smaller than h. Let n_k be the number of vertices in G of height smaller than h. It is easy to see that every closed subgraph of G with k vertices is of height h, it

contains all vertices of G of height smaller than h, and $k-n_h$ initial vertices of height h. This completes the proof since within each component of a level order graph all vertices of the same level are isomorphic. □

We now apply Lemma 4.1 to level orders with q components.

LEMMA 4.2. *Let G be a level order with components $H_1, \cdots, H_q$. If I and J are two closed subgraphs of G containing the same number of vertices from each component, then I is transitively isomorphic to J.*

Proof. Let $I_1, \cdots, I_q$ be the subgraphs of I that contain all vertices of I from components $H_1, \cdots, H_q$, respectively. Define $J_1, \cdots, J_q$ similarly. Since I and J are closed subgraphs of G, we conclude that I_r and J_r are closed subgraphs of H_r, for $1 \leqq r \leqq q$. The graphs I and J contain the same amount of vertices from each H_r, that is, $|I_r| = |J_r|$. By Lemma 4.1 we conclude that I_r is isomorphic to J_r. This implies that I is isomorphic to J. □

DEFINITION 4.1. Let G be a level order with components $H_1, \cdots, H_q$ and I be a closed subgraph of G. Then I is *represented by the tuple* $\langle n_1, \cdots, n_q \rangle$ if I contains n_r vertices of H_r, $1 \leqq r \leqq q$.

In the following corollary we rewrite Lemma 4.2 using this definition.

COROLLARY 4.1. *Let G be a level order with the components $H_1, \cdots, H_q$. If two subgraphs of G are represented by the same tuple, then they are isomorphic. All closed subgraphs of G correspond to $O(n^q)$ distinct tuples.*

Proof. The first part of the corollary follows directly from Lemma 4.2 and Definition 4.1.

Every closed subgraph of G can be represented by some tuple $\langle n_1, \cdots, n_q \rangle$. By Definition 4.1 we know that $0 \leqq n_r \leqq |H_r|$, for $1 \leqq r \leqq q$. Since $|H_r| \leqq n$, all closed subgraphs of G can be represented by at most $(n+1)^q$ tuples. Clearly, $(n+1)^q \leqq (q+1)n^q$; since q is constant, this implies that $(n+1)^q = O(n^q)$. □

The above corollary describes the key property of level orders that guarantees the polynomial algorithms. A level order with q components contains at most $O(n^q)$ equivalence classes of closed subgraphs. During the scheduling algorithm, we will keep track of all of these closed subgraphs via dynamic programming.

The following length function is used recursively in the polynomial algorithms we present later.

DEFINITION 4.2. Let G be a graph. Denote by $\lambda(G, m)$ the length of an optimal schedule for G fitting the straight profile of breadth m.

LEMMA 4.3. *The length function λ can be calculated by the following recursive formula*:

$$\lambda(\phi, m) = 0;$$

$$\lambda(G, m) = 1 + \min(\{\lambda(G-R, m) \mid R \text{ is a set of initial vertices of } G,\ 1 \leqq |R| \leqq m\}). \tag{4.1}$$

Proof. The proof is clear from the following fact. Let R be any set of initial vertices of G such that $1 \leqq |R| \leqq m$. Then $\lambda(G-R, m) = \lambda(G, m) - 1$ if and only if there exists an optimal schedule for G fitting a straight profile of breadth m with R in its first slot. □

We will now apply the recursive formula (4.1) to evaluate λ for all closed subgraphs of a level order.

LEMMA 4.4. *Let G be a level order with a constant number q of components. Then $\lambda(I, m)$ can be evaluated for all closed subgraphs I of G in time $O(m^q n^q)$ and space $O(n^q)$.*

Proof. The following algorithm evaluates $\lambda(G)$ via dynamic programming, within time $O(m^q n^q)$.

ALGORITHM 4.1.
1. $\lambda(\phi) = 0$
2. **for** $k := 1$ **to** n **do**
   2.1. **for** all closed subgraphs I of G with k vertices **do**
      2.1.1. $\lambda(I) := 1 + \min(\{\lambda(I - R) |\ R$ is a set of initial vertices of I and $1 \leqq |I| \leqq m\})$.

The correctness follows from Lemma 4.3. To obtain the time bound we need to show more explicitly how we gather the information during the execution of the algorithm. Let $H_1, \cdots, H_q$ be the components of a level order graph. For every component H_r, denote by TOP (p, r) the number of initial vertices in the closed subgraph of H_r that contains p vertices. By Lemma 4.1 all such closed subgraphs are isomorphic. Furthermore, a level order component is completely specified by a sequence of natural numbers specifying how many vertices are in each level. Therefore, TOP (p, r), for every $0 \leqq p \leqq |H_r|$ and $1 \leqq r \leqq q$, can be created in linear time.

By Corollary 4.1, all closed subgraphs of G can be represented by $O(n^q)$ equivalence classes. Each equivalence class is determined by a vector $n = (n_1, n_2, \cdots, n_q)$ in $|H_1| \times |H_2| \times \cdots \times |H_q|$, where n_i is the number of vertices from component H_i. For every such a vector $\bar{n}$ denote by $\mathcal{R}(\bar{n}, m)$ the set of all vectors $\bar{n}'$ obtained from $\bar{n}$ by removing for every i, $n_i - n_i'$ initial vertices from the closed subgraph of H_i (represented by n_i), such that $1 \leqq \sum_{r=1}^{q} (n_r - n_r') \leqq m$ and $n_i - n_i' \leqq$ TOP (n_i, i).

Note that $n_r - n_r' \leqq m$, which implies that $|\mathcal{R}(\bar{n}, m)| = O((m+1)^q)$. Clearly $(m+1)^q \leqq (q+1)m^q$; since q is a constant, we have $|\mathcal{R}(\bar{n}, m)| = O(m^q)$.

Algorithm 4.1 can be rewritten as follows:

ALGORITHM 4.1′.
1′. $\lambda(\phi) = 0$
2′. **for** $k := 1$ **to** n **do**
   2.1′. **for** all $\bar{n} \in |H_1| \times |H_2| \times \cdots \times |H_q|$, such that $\sum_{r=1}^{q} n_r = k$ **do**
      2.1.1′. $\lambda(\bar{n}) := 1 + \min(\{\lambda(\bar{n}') | \bar{n}' \in \mathcal{R}(\bar{n}, m)\})$.

At Step 2.1′, we partition all $O(n^q)$ tuples according to the number of vertices they contain. This can be done in time $O(n^q)$. To implement Step 2.1.1′, we make use of the data structure for TOP (p, r). Since there are $O(m^q)$ choices for $\bar{n}'$, we get the time bound $O(m^q n^q)$, which completes the proof of the time bound. The algorithm needs $O(n^q)$ space to represent all equivalence classes. The array TOP and all remaining data structures require only $O(n)$ space. □

Having analyzed the function λ for all closed subgraphs of a level order G with q components, we can retrieve an optimal schedule for G and the straight profile of breadth m.

THEOREM 4.1. *Let G be a level order with a constant amount $q \geqq 2$ of components. An optimal schedule for G and the straight profile of breadth m can be found in time $O(m^q n^q)$ and space $O(n^q)$.*

Proof. Lemma 4.4 implies that we can prepare the values $\lambda(I, m)$ for all closed subgraphs I of G in time $O(m^q n^q)$ and space $O(n^q)$. We use these values to find an optimal schedule for G.

ALGORITHM 4.2.
1. $k := 0$
2. **while** G is nonempty **do**
   2.1. $k := k + 1$

2.2. Let R be a subset of initial vertices of G such that $1 \leqq |R| \leqq m$ and $\lambda(G-R) = \lambda(G) - 1$

2.3. $S_k := R$

2.4. $G := G - R$.

If we use in Step 2.2 the vector representation for the closed subgraphs then it is easy to see that the total number of different R we scan in Step 2.2 is bounded by $O(m^q)$. Clearly every other step is bounded by $O(q)$. This implies that the total time complexity for both algorithms (Algorithm 4.1 and Algorithm 4.2) is $O(m^q n^q)$. □

We extend Theorem 4.1 to nonstraight profiles of breadth m. For that we need to refine the definition of the function λ.

DEFINITION 4.3. Let G be a graph and $M = (m_1, \cdots, m_d)$ be a profile of breadth m. Then

$$\lambda(G, M) := \min(\{k | \text{there exist a schedule for } G \text{ and } (m_{d-k+1}, \cdots, m_d)\}).$$

Note that k is the length of the profile $(m_{d-k+1}, \cdots, m_d)$. Note also that if m is straight, then Definition 4.3 degenerates to Definition 4.2.

LEMMA 4.5. *The function λ can be expressed recursively as follows*:

$$\lambda(\phi, M) = 0;$$

$$\lambda(G, M) = 1 + \min(\{\lambda(G-R, M) | R \text{ is a set of initial vertices of } G \text{ and } 1 \leqq |R| \leqq m_{d-\lambda(G-R,M)}\}). \tag{4.2}$$

Proof. The proof follows directly from the following observations:

(i) $\lambda(G, M)$ is undefined if and only if there exists no schedule for G fitting M.

(ii) If $\lambda(G, M)$ is defined, then there exists a set R such that $1 \leqq |R| \leqq m_{d-\lambda(G,M)+1}$, and $\lambda(G-R, M) = \lambda(G, M) - 1$.

(iii) If $\lambda(G, M)$ is defined, then for any set R, such that $|R| \leqq m_{d-\lambda(G,M)+1}$ and $\lambda(G-R, M) = \lambda(G, M) - 1$, there exists a schedule for G fitting the profile $(m_{d-\lambda(G,M)+1}, \cdots, m_d)$ which starts with R. □

As in Lemma 4.4 we evaluate the function λ for all closed subgraphs of the level order G achieving the same time bound as for straight profiles.

LEMMA 4.6. *Let G be a level order with a constant amount q of components, and let M be a profile of breadth m. Then $\lambda(I, M)$ can be evaluated for all closed subgraphs I of G in time $O(m^q n^q)$ and space $O(n^q)$.*

Proof. As with Lemma 4.4 and Algorithm 4.1, the only change is in Step 2.1. We replace the recursive formula for straight profiles (4.1) by the recursive formula for arbitrary profiles (4.2). Since $m_{d-\lambda(G-R,M)+1} \leqq m$, we get the same time bound. □

Knowing the function λ we are ready to retrieve a schedule in a similar fashion as in Theorem 4.1 and Algorithm 4.2.

THEOREM 4.2. *Let G be a level order with a constant amount of components, and let M be a profile of breadth m. Then a schedule for G and $M' = (m_{d-\lambda G,M)+1}, \cdots, m_d)$ can be found in time $O(m^q n^q)$ and space $O(n^q)$.*

Proof. Applying Lemma 4.6 we find $\lambda(I, M)$ for all closed subgraphs I of G. To retrieve a schedule for G and M' we use Algorithm 4.2 of Theorem 4.1. Let b be $\lambda(G, M)$, where G is the original graph and M the profile. Since M is not necessarily straight, we change the bound of $|R|$ from m to m_{d-b+k}. □

The *reversed graph* G^R of a graph G is a graph obtained by reversing all the edges in G. For a profile $M = (m_1, m_2, \cdots, m_d)$ we define the *reversed profile* M^R to be $M^R = (m_d, m_{d-1}, \cdots, m_1)$. The reversed schedule S^R is defined accordingly.

In the subsequent corollary we apply Theorem 4.2 on the reversed graph G^R and reversed profile M^R to find an optimal schedule for G and M.

COROLLARY 4.3. *Let G, q and M be defined as in Theorem 4.2. An optimal schedule for G and M has length $\lambda(G^R, M^R)$ and can be found in time $O(m^q n^q)$ and space $O(n^q)$.*

Proof. Rewriting Definition 4.3 for the case where the arguments of λ are the graph G^R and the profile $M^R = (m_d, m_{d-1}, \cdots, m_1)$ we get the following:

$$\lambda(G^R, M^R) = \min(\{k | \text{there exists a schedule for } G^R \text{ and } (m_k, m_{k-1}, \cdots, m_1)\}).$$

Reversing the graph G^R and the profile $(m_k, m_{k-1}, \cdots, m_1)$ the above formula can be rewritten as:

$$\lambda(G^R, M^R) = \min(\{k | \text{there exist a schedule for } G \text{ and } (m_1, \cdots, m_k)\}).$$

We conclude that $\lambda(G^R, M^R)$ is the length of an optimal schedule for G and M.

To prove the second part of the corollary we observe that when G is a level order G^R is also. Applying Theorem 4.2 to G^R and M^R we get a schedule S for G^R and M'^R in time $O(m'^q n^q)$, where M'^R is the profile $(m_{\lambda(G^R, M^R)}, \cdots, m_1)$. Since $\lambda(G^R, M^R)$ is the length of an optimal schedule for G and M, we conclude that S^R is an optimal schedule for G and M. □

Note that $\lambda(G, M)$ is not the length of an optimal schedule for G and M. It is also not the length of an optimal schedule for G and M' (defined as in Theorem 4.2). We could have defined $\lambda(G, M)$ as the length of an optimal schedule for G and M replacing Definition 4.3. Then the recursive formula corresponding to (4.2) would be:

$$\begin{aligned} &\lambda(\phi, M) = 0 \\ &\lambda(G, M) = 1 + \min(\{k | \lambda(H, M) = k, \\ &\qquad \text{where } H \text{ is a subgraph of } G \text{ obtained by removing} \\ &\qquad \text{at least one and no more than } m_{k+1} \text{ terminal vertices}\}), \end{aligned} \tag{4.3}$$

where a *terminal* vertex is a vertex with no successors.

The scheduling algorithms described in the paper obtain optimal schedules by iteratively removing sets of initial vertices from the remaining graph and scheduling them in the first, second, $\cdots$ time slot (for instance, see Algorithm 4.2). Formula (4.3) corresponds to doing the scheduling process "backwards": Iteratively remove sets of terminal vertices from the remaining graph and schedule them in the last, second to last, $\cdots$ time slot. We choose the standard way of scheduling—that is, to iteratively remove sets of initial vertices—even though scheduling "backwards" would make Corollary 4.3 unnecessary.

Combining Corollary 4.1 with Theorem 3.2 we prove the final result of this section.

COROLLARY 4.4. *Let G be a graph such that $H(G, m)$ is a level order and M a profile of constant breadth $m \geq 3$. Then an optimal schedule for G and M can be found in $O(n^{m-1})$ time and space.*

Proof. Let q be the number of components of $H(G, m)$. If $q = 0$, then by Corollary 3.1 an optimal schedule for G and M can be found in time and space $O(n+e) \sim O(n^{m-1})$, since $m \geq 3$. If $q \geq 1$, then Corollary 4.3 shows that an optimal schedule for $H(G, m)$ and M can be found in time $O(m^q n^q)$ and space $O(n^q)$. Property M1 of the median implies that $q < m-1$; therefore $O(m^q n^q) = O(m^{m-1} n^{m-1})$, which equals $O(n^{m-1})$, since m is constant.

So far we have shown that an optimal schedule for $H(G)$ and M can be found in time and space $O(n^{m-1})$. By the reduction theorem, we conclude that an optimal schedule for the whole graph G and M can be found within the same time and space bounds. □

5. Inforests and outforests. In this section we give polynomial algorithms for finding an optimal schedule if the precedence graph is an inforest, an outforest or an opposing forest and the profile has constant breadth $m \geqq 3$. For inforest we present an $O(n^{m-1})$ algorithm, for outforest an $O(n^{m-1} \log n)$ algorithm and an $O(n^{2m-2} \log n)$ algorithm for opposing forest. All three algorithms require $O(n^{m-1})$ space. The algorithm for opposing forest assumes that the profile is straight, whereas the algorithms for inforest and outforest work for arbitrary profile.

A profile M is called *nondecreasing* (respectively, nonincreasing) if $m_i \leqq m_{i+1}$ (respectively, $m_i \geqq m_{i+1}$), for $1 \leqq i \leqq d$. The algorithms of [GJ83] for obtaining optimal schedules for inforests, outforests and opposing forests are less time efficient than ours: the algorithm for opposing forests requires $O(n^{m^2+2m-5} \log n)$ time, and the algorithms for outforests and inforests assume nondecreasing and nonincreasing profiles, respectively, and require $O(n^{m^2+m-6} \log n)$ time.

As in the case of scheduling level orders, deciding whether a feasible schedule exists for each of the three types of forests becomes NP-complete if the breadth of the profile is a variable of the problem instance [GJ83], [Ma81], [Wa81]. The corresponding problems stay NP-complete even in the following restricted cases: nondecreasing profile and outforest graph, nonincreasing profile and inforest and straight profile and opposing forest [GJ83], [Ma81], [Wa81]. In other related cases HLF produces an optimal schedule even if the breadth of the profile is unbounded: straight and inforest [Hu61] or outforest [Br81], [DW84a], nonincreasing profile and outforest [DW84a] and nondecreasing profile and inforest [DW84a]. Note that forests are special cases of series parallel digraphs [LT79] and therefore HLF produces an optimal schedule for forests and profiles of breadth 2 [Go76].

In this section we first present an algorithm for scheduling an outforest on a profile of $O(1)$ breadth. To do this we observe that there are at most $O(n^{m-1})$ choices for the high-graph of a closed subgraph of an outforest. This fact is used to define an equivalence relation on the set of all closed subgraphs of an outforest. The equivalence relation partitions this set into a polynomial amount of equivalence classes. Two subgraphs of the same equivalence class have the same high-graph and the same number of vertices in their low-graph. We then define a length function on the equivalence classes similar to the previous section. All closed subgraphs of one equivalence class have the same length. This length is related to the length of the optimal schedules of the subgraphs of the equivalence class. As in the previous section, the length function is evaluated via dynamic programming. We then use the length function in an algorithm which finds an optimal schedule for an outforest and a profile of constant breadth. This algorithm is similar to the MERGE Algorithm 3.1. To get an optimal schedule for an inforest we apply the outforest algorithm to the reversed profile and the reversed inforest, which is an outforest. Our algorithm for opposing forest is obtained by combining the inforest algorithm with a result of [GJ83].

Let T be a subset of the vertices of G, then CLOSE (T) is the closed subgraph induced by T, that is, the subgraph which contains the vertices of T and all the successors.

The following theorem implies that we have to keep track of $O(n^{m-1})$ high-graphs while scheduling an outforest. This result will imply the $O(n^{m-1})$ time and space bound for scheduling an outforest on a profile of constant breadth m.

THEOREM 5.1. *The high-graph of a closed subgraph of an outforest G and breadth m contains less than m initial vertices. All closed subgraphs of G have $O(n^{m-1})$ different high-graphs.*

Proof. Since G is an outforest, every closed subgraph J of G is, as is the high-graph of J. By property M1 of the median we know that $H(J, m)$ consists of less than m

components, which in this case are outtrees. Each outtree corresponds to exactly one initial vertex and therefore $H(J, m)$ corresponds to a set of less than m initial vertices that are incomparable with each other. On the other hand, each set of up to $m-1$ incomparable vertices of G induces an outforest which is the high-graph of some closed subgraph of G. Since m is constant, there are $O(n^{m-1})$ choices for a set of up to $m-1$ vertices. This completes the proof of the theorem. □

DEFINITION 5.1. Let J and K be two closed subgraphs of a graph G. Then the subgraphs J and K are *equivalent*, $J \equiv K$, if and only if they have the same high-graph, and the number of vertices of J and K that do not have any predecessors above the corresponding medians is the same. That is,

$$J \equiv K \text{ iff } (H(J) = H(K) \text{ and } |L(J)| = |L(K)|).$$

Theorem 3.3 and Theorem 5.1 are the motivation for the definition of the equivalence relation. From Theorem 3.3 we know that if two closed subgraphs are equivalent then the length of their optimal schedules for a given profile of breadth m is the same. Theorem 5.1 implies that if m is a constant, then there is a polynomial number of different equivalence classes.

Let J be a closed subgraph of a graph G. Then INIT (J) denotes the set of all initial vertices of J. Note that the closed subgraph J is completely determined by INIT (J) in the sense that J consists of INIT (J) plus all successors of the vertices of this set, that is $J = \text{CLOSE (INIT } (J))$. The equivalence class to which J belongs is completely specified by $H(J)$ (or INIT $(H(J))$), and $|L(J)|$. We denote the equivalence class of J as the tuple $[\text{INIT } (H(J)), |L(J)|]$. Applying this notation we get the following: A closed subgraph K of G is in the equivalence class $[I, w]$ iff INIT $(H(K)) = I$ and $|L(K)| = w$.

The length function we use in this section is a function of an equivalence class instead of a graph as in § 4.

DEFINITION 5.2. Let G be an outforest, let $[I, w]$ be an equivalence class of G, and M be a profile of breadth m. Then the length $\lambda(I, w, M)$ is the minimum k for which there exists a schedule for the members of $[I, w]$ fitting the profile $(m_{d-k+1}, \cdots, m_d)$.

The function λ is well defined because of Theorem 3.3. This theorem implies that for any two subgraphs K and J, such that $K \equiv J$, there exists a schedule for J and $M' = (m_{d-k+1}, \cdots, m_d)$ if and only if there exists one for K and M'. Notice that the length $\lambda(I, w, M)$ is undefined if and only if there exist no schedules for the closed subgraphs of $[I, w]$ that fit M.

In the following lemma we show how to calculate the value of $\lambda(I, w)$ from $\lambda(I, 0)$.

LEMMA 5.1. *Let* $[I, 0]$ *be an equivalence class,* $M = (m_1, \cdots, m_d)$ *be a profile, and* p *be the number of idle periods in a schedule for* CLOSE (I) *and* $M' = (m_{d-\lambda(I,0)+1}, \cdots, m_d)$. *Then*

$$\lambda(I, w, M) = \begin{cases} \textit{undefined} & \textit{if } \lambda(I, 0, M) \textit{ is undefined,} \\ \lambda(I, 0, M) & \textit{if } \lambda(I, 0, M) \textit{ is defined and } p \geq w, \\ \bar{\lambda} & \textit{if } \lambda(I, 0, M) \textit{ is defined and } p < w, \end{cases}$$

where

$$\bar{\lambda} = \min\left(\left\{k \mid k \geq 1 \textit{ and } \sum_{i=d-k+1}^{d} m_i \geq |\text{CLOSE } (I)| + w\right\}\right).$$

Proof. Case $\lambda(I, 0, M)$ *undefined.* Since the closed subgraphs of $[I, w]$ have at

least as many vertices as the closed subgraphs of $[I, 0]$, the value of $\lambda(I, w)$ is also undefined; and if $\lambda(I, 0)$ is defined, then $\lambda(I, w) \geqq \lambda(I, 0)$.

For the case where $\lambda(I, 0)$ is defined, let S be a schedule for CLOSE (I) and M'. This schedule S has p idle periods and

$$p = \left(\sum_{i=d-\lambda(I,0)+1}^{d} m_i \right) - |\text{CLOSE}(I)|.$$

Case $p \geqq w$. By Theorems 3.1 and 3.3, we know that for any graph J of $[I, w]$ there exists a schedule S' for J and M'. Note that $J \in [I, w]$ if and only if INIT $(H(J)) = I$ and $|L(J)| = w$ and S' has $p - w$ idle periods. We conclude that if $p \geqq w$, then $\lambda(I, w) = \lambda(I, 0)$.

Case $p < w$. Clearly $\lambda(I, w) \geqq \bar{\lambda}$ since the subgraphs of $[I, w]$ contain $|\text{CLOSE}(I)| + w$ vertices. If $\bar{\lambda}$ is undefined, then there is not enough "space" in the profile M to schedule a graph of $[I, w]$ and therefore $\lambda(I, w)$ is undefined also.

For the case where $\bar{\lambda}$ is defined we want to show that for each member of $[I, w]$ there exists a schedule that fits $\bar{M} = (m_{d-\bar{\lambda}+1}, \cdots, m_d)$. Since $\bar{\lambda} > \lambda(I, 0)$, the schedule S for CLOSE (I) and M' can be embedded into the profile $\bar{M}$. Therefore, there exists a schedule for CLOSE (I) and $\bar{M}$, and such a schedule has more than w idle periods, since $\sum_{i=d-\bar{\lambda}+1}^{d} m_i \geqq |\text{CLOSE}(I)| + w$. By applying Theorems 3.1 and 3.3 again, we conclude that there exists a schedule for any member of $[I, w]$ and $\bar{M}$; therefore, $\lambda(I, w) = \bar{\lambda}$, if $p < w$. □

We now want to show that the calculation of $\lambda(I, w)$ from $\lambda(I, 0)$ as described in the previous lemma can be implemented efficiently.

LEMMA 5.2. *Let G be an outforest and M be a profile of constant breadth $m \geqq 3$. Given the appropriate data structures, which can be created in time and space $O(n^{m-1})$, then for any equivalence class $[I, w]$ of G, $\lambda(I, w, M)$ can be calculated from $\lambda(I, 0, M)$ in constant time.*

Proof. The following data structures can be created in time and space $O(n^{m-1})$ and allow us to calculate $\lambda(I, w, M)$ in constant time from $\lambda(I, 0, M)$.

Data structure $A = (U, N, L)$. Let G be an outforest and M be a variable profile of breadth m.

(i) For every $x \in G$, $U[x]$ is one plus the number of successors of x in G.

(ii) For every set T of up to $m-1$ incomparable vertices of G, $U[T] = \sum_{y \in T} U[y]$. Note that if T is a set of incomparable vertices then $U[T] = |\text{CLOSE}(T)|$.

(iii) For every k, $1 \leqq k \leqq d$, $N[k]$ is the total number of available processors in the subprofile $(m_{d-k+1}, \cdots, m_d)$. That is, $N[k] = \sum_{i=d-k+1}^{d} m_i$. Note that k is the length of the subprofile.

(iv) For every r, $1 \leqq r \leqq N[d]$, $L[r]$ is the length of the shortest profile $(m_{d-k+1}, \cdots, m_d)$ having a total amount of r available processors. Therefore, $L[r] = \min(\{k | N[k] \geqq r\})$.

The properties of data structure A which we need for proving Lemma 5.2 are:

A1. Given the value of $\lambda(I, 0)$ then the value of $\lambda(I, w)$ can be calculated in constant time.

A2. The data structure A can be created in time and space $O(n^{m-1})$.

Proof of Property A1. By Theorem 5.1 and Definition 5.1 we know that $|I| \leqq m - 1$, since G is an outforest. In Lemma 5.1 a formula was given to calculate $\lambda(I, w)$ from $\lambda(I, 0)$. Using the arrays U, N and L we can rewrite this formula in the following way:

$$\lambda(I, w) = \begin{cases} \lambda(I, 0) & \text{if } p \geqq w, \\ L[U[I] + w] & \text{if } p < w. \end{cases}$$

Furthermore, the number of idle periods p in a schedule for CLOSE (I) and profile $(m_{d-\lambda(I,0)+1}, \cdots, m_d)$ can be expressed as: $p = N[\lambda(I,0)] - U[I]$. Therefore, if data structure A is given, and $\lambda(I,0)$ and I is known, then $\lambda(I,w)$ can be calculated in constant time.

Proof of Property A2. (i) Determining the number of successors of each vertex of the outforest G can be done in one traversal of the outforest. Thus the array U can be evaluated for all vertices of G in time $O(n)$.

(ii) There are $O(n^{m-1})$ choices for a set T of up to $m-1$ vertices. For a given T the value $U[T]$ can be found in constant time, since m is constant. Therefore, U can be evaluated for all sets T in time $O(n^{m-1})$.

(iii) and (iv) The matrices N and L can easily be created in time $O(n)$.

This completes the proof of the properties of data structure A and therefore also the proof of Lemma 5.2. □

As in the previous section we give a recursive formula for the function λ. While scheduling a graph we repeatedly remove sets of initial vertices from the graph. The notation $A \xrightarrow{R} B$ denotes that B is obtained from A by removing R, which is a set of initial vertices.

Using the above notation we can give a recursive formula for $\lambda(I,0)$:

$$\lambda(I,0) = 1 + \min(\{\lambda(I',w') | ([I,0]) \xrightarrow{R} ([I',w']) \wedge 1 \leqq |R| \leqq m_{d-\lambda(I',w')}\}). \tag{5.1}$$

The notation $([I,0]) \xrightarrow{R} ([I',w'])$ means the following: Let J be a graph of $[I,0]$, then by removing R, which is a subset of I, from J, we obtain a subgraph J', where $J' \in [I',w']$.

The correctness of the above formula is obvious. We make all possible choices to remove sets of initial vertices and we recurse on the remaining graph. This formula is used to evaluate $\lambda(I,0)$ for all sets I of up to $m-1$ incomparable vertices of G via dynamic programming.

LEMMA 5.3. *Let G be an outforest and M be a profile of constant breadth m. Then the function λ can be evaluated for all equivalence classes $[I,0]$ of G in time and space $O(n^{m-1})$.*

Proof. The following algorithm evaluates λ for all $[I,0]$ of G in time $O(n^{m-1})$.

ALGORITHM 5.1.

1. $\lambda(\phi,0) := 0$;
2. **for** $i = 1$ **to** n **do**
   2.1. **for** all sets I of up to $m-1$ incomparable vertices of G, such that $|\text{CLOSE}(I)| = i$ **do**
   2.1.1. $\lambda(I,0) := 1 + \min(\{\lambda(I',w') | ([I,0]) \xrightarrow{R} ([I',w'])$ and $1 \leqq |R| \leqq m_{d-\lambda(I',w')}\})$.

Proof of correctness. The correctness follows from (5.1). Notice also that at Step 2.1.1 $|\text{CLOSE}(I')| < |\text{CLOSE}(I)|$ since $|R| \geqq 1$. As shown in Lemma 5.1 the value of $\lambda(I',q')$ is determined by $\lambda(I,0)$ and w.

Proof of the bounds. By Theorem 5.1 and Definition 5.1 we know that for any equivalence class $[I,w]$ of an outforest G, $|I| \leqq m-1$ and I is a set of incomparable vertices. Note that $|I|$ is constant when m is constant. There are $O(n^{m-1})$ different sets of up to $m-1$ vertices of G. Claim 5.1 below implies that with an appropriate data structure, we can determine in constant time whether the vertices of a given set of cardinality up to $m-1$ are incomparable or not. Therefore, all sets of up to $m-1$ incomparable vertices of G can be found in time and space $O(n^{m-1})$. We then create

data structure A in time and space $O(n^{m-1})$ and bucket sort all sets T of up to $m-1$ incomparable vertices of G according to $U[T]$. Thus, Claim 5.1 and Lemma 5.2 imply that Steps 2 and 2.1 can be implemented in time $O(n^{m-1})$.

CLAIM 5.1. *Given the preorder number and the number of successors for every vertex of G, then for any set T of up to $m-1$ vertices of G it can be determined in constant time whether T is a set of incomparable vertices or not.*

Proof of the claim. Let $p(x)$ denote the preorder number and $n(x)$ the number of successors of the vertex x of G. Now for any two vertices x and y of G, x precedes y if and only if $p(x) \leqq p(y) \leqq p(x)+n(x)$ (see [AH74] for details). To decide in constant time whether some set T of up to $m-1$ vertices of G is incomparable or not we use the following fact: T is a set of incomparable vertices if and only if for every x and y of T, x does not precede y. Since T has a constant size, this can be done in constant time, which completes the proof of the claim. □

Since the preorder number and the number of successors of every vertex can be found in $O(n)$ time the claim implies that Steps 2 and 2.1 can be implemented in time $O(n^{m-1})$. By Theorem 5.1 there are $O(n^{m-1})$ sets of up to $m-1$ incomparable vertices of G. Thus Step 2.1.1 gets executed $O(n^{m-1})$ times and to get an overall $O(n^{m-1})$ time bound we need to show that Step 2.1.1 can be implemented in constant time.

In Lemma 5.2 we showed that $\lambda(I', w')$ can be calculated in constant time given data structure A and $\lambda(I', 0)$. At Step 2.1.1 the value of $\lambda(I', 0)$ has been calculated already since $|\text{CLOSE}(I')| < |\text{CLOSE}(I)|$.

The set R is a subset of the set I and $|I| < m$. Since m is constant, there is only a constant amount of choices for R. Thus to prove that Step 2.1.1 is constant we have left to show that given a set R then $[I', w']$ can be determined in constant time. This is achieved by the following data structure.

Data structure B. The outforest G is represented by its adjacency lists [AH74], in which the immediate successors of every vertex of x are given in a linked list sorted according to decreasing height.

Properties of data structure B.

B1. Let J be a closed subgraph of G, let R be a subset of $\text{INIT}(H(J))$, and let $(\text{INIT}(H(J)),\ \mu(J),\ |L(J)|) \xrightarrow{R} (I', \mu', w')$. Then (I', μ', w') can be obtained from $(\text{INIT}(H(J)),\ \mu(J),\ |L(J)|)$ in constant time.

B2. Data structure B can be created in time $O(n)$.

Proof of Property B1. For every vertex $x \in R$, let T_x be a set of m highest immediate successors of x. If x has less than m immediate successors then let T_x be all immediate successors of x. Define T to be the following set of vertices: $T := \{\text{INIT}(H(J)) - R\} \cup (\cup_{x \in R}(T_x))$.

Obviously T can be found in $O(m^2)$ time, since the immediate successors of $x \in R$ are given in decreasing height. $|\text{INIT}(H(J))| < m$ and $|T| < m^2$. Note that $O(m^2) = O(1)$, because m is constant. Since G is an outforest every vertex of T corresponds to an outtree in J', which is the subgraph of J obtained by removing the set R from J. All the roots of height at least as high as the mth highest component of J' are contained in T. Therefore, $\mu(J') = \mu'$ is one plus the height of an mth highest vertex of T. If $|T| < m$ then μ' is set to zero. Furthermore, $I' = \text{INIT}(J')$ is a subset of T, i.e., I' is the set of all vertices of T of height bigger than μ'. Since the size of T is constant, μ' and I' can be determined in constant time. Finally w' is computed as follows: $w' = |L(J')| = U[I] + |L(J)| - U[I'] - |R|$. This completes the proof of Property B1 of data structure B.

Proof of Property B2. All vertices of G can be bucket sorted according to their height in linear time. Create the adjacency lists of G as follows: starting at the highest

vertices and continue according to decreasing height, insert each vertex to the end of the adjacency list of the immediate predecessor of it (in constant time). Thus data structure B can be constructed in time $O(n)$.

To complete the proof of the time bound we still have to show that in Step 2.1.1 $[I', w']$ can be determined in constant time when I and R are given. This follows from Property B1 of data structure B. Note that at Step 2.1.1, $J = \text{CLOSE}(I) = H(J)$, $\mu(J) = 0$ and $|L(J)| = 0$. □

We are now ready to present the main result of this section.

THEOREM 5.2. *Let G be an outforest and $M = (m_1, \cdots, m_d)$ be a profile of constant breadth m. Then it can be determined in time and space $O(n^{m-1})$ whether there exists a schedule for G and M. If such a schedule exists, then we can find a schedule for G fitting the profile $(m_{d-\lambda(I,w)+1}, \cdots, m_d)$, where $G \in [I, w]$, in time $O(n^{m-1})$.*

Proof. To determine whether there exists a schedule for G and M, we apply Lemma 5.3 and evaluate λ for all equivalence classes $[I, 0]$ of G (Algorithm 5.1). This can be done in time and space $O(n^{m-1})$. Given the value of $\lambda(I, 0)$, it is easy to calculate $\lambda(I, w)$ (see Lemma 5.3). Note that $\lambda(I, w)$ is defined if and only if there exists a schedule for G and M (see Definition 5.2). Thus we showed that one can decide in time and space $O(n^{m-1})$ whether there exists a schedule for G and M.

If such a schedule exists then the following algorithm finds a schedule for G fitting the profile $M' = (m_{d-\lambda(I,w)+1}, \cdots, m_d)$, such that $G \in [I, w]$, in time and space $O(n^{m-1})$.

ALGORITHM 5.2.

1. $I_H := \text{INIT}(H(G))$ $I_L := \text{INIT}(L(G))$ $\mu := \mu(G)$;
   $\lambda := \lambda(I_H, |L(G)|)$
2. **for** $k := d - \lambda + 1$ **to** d **do**
   2.1. $j := k - d + \lambda$
   if $|I_H| > m_k$
   2.1.1. **then** Find R such that $([I_H, 0]) \xrightarrow{R} ([I', w'])$,
   $|R| = m_k$, and $\lambda(I', w') \leqq d - k$
   $(S)_j := R$
   2.1.2. **else** Find a set T of $m_k - |I_H|$ highest vertices of I_L
   $(S)_j := I_H \cup T$
   2.2. Determine I'_H, I'_L, μ' such that $(I_H, I_L, \mu) \xrightarrow{(S)_j} (I'_H, I'_L, \mu')$
   $I_H := I'_H$ $I_L := I'_L$ $\mu := \mu'$.

Proof of correctness. Assume we are before Step 2.1 and the Loop 2 has been executed already several times, that is, vertices of the original graph G have been put into the slots $1, 2, \cdots, k - d + \lambda - 1$ of S. Let G be the remaining graph at this point, that is, the closed subgraph of the original graph that has not been scheduled yet. Then applying the notation of the algorithm we have: $I_H = I(H(G))$, $I_L = I(L(G))$ and $\mu = \mu(G)$.

It is easy to see that both at Step 2.1.1 and 2.1.2 the set $(S)_j$ is a subset of the initial vertices of G. Thus in the constructed schedule S the precedence constraints specified by the graph G are not violated.

The correctness of Algorithm 5.2 is shown by proving the following loop invariant: There exists a schedule for G and $(m_k, \cdots, m_d)$.

At Step 1 we set λ to $\lambda(I_H, |L(G)|)$ and we know that this value is defined. The definition of the function λ (Definition 5.2) implies that there exists a schedule for G and $(m_{d-\lambda+1}, \cdots, m_d)$ after Step 1. Therefore, the invariant holds for $k = d - \lambda + 1$ before the first execution of Loop 2.

We now want to prove the following: If there exists a schedule for G and $(m_k, \cdots, m_d)$ before Step 2.1, then there exists a schedule for G' and $(m_{k+1}, \cdots, m_d)$ after Step 2.2, such that $(G) \xrightarrow{(S)_j} (G')$. The proof of the above implication follows from Theorem 3.4.

Case $|I_H| > m_1$. By Theorem 3.4 there exists a set R of m_k vertices of I_H that starts a schedule for $H(G)$ and $(m_k, \cdots, m_d)$. Define I' and w' such that $([I_H, 0]) \xrightarrow{R} ([I', w'])$. Since there exists a schedule for $H(G)$ and $(m_k, \cdots, m_d)$ starting with R we have $\lambda([I', w']) \leqq d - k$. So far we have shown that the set R as defined in Step 2.1.1 exists.

On the other hand, any set R that is defined as in Step 2.1.1 starts a schedule for $H(G)$ and $(m_k, \cdots, m_d)$. This is implied by the fact that $\lambda(I', w') \leqq d - k$. Define $\bar{H}$ such that $(H(G)) \xrightarrow{R} (\bar{H})$, then $\bar{H} \in [I', w']$ and there exists a schedule for $\bar{H}$ and $(m_{k+1}, \cdots, m_d)$, since $\lambda(I', w') \leqq d - k$. Note that $(m_{k+1}, \cdots, m_d)$ has length $d - k$. By Theorem 3.4 we conclude that R as defined in Step 2.1.1 starts a schedule for G and $(m_k, \cdots, m_d)$, since it starts one for $H(G)$ and $(m_k, \cdots, m_d)$. This implies that there exists a schedule for G' and $(m_{k+1}, \cdots, m_d)$, since $(S)_j = R$ and $(G) \xrightarrow{R} (G')$.

Case $|I_H| \leqq m_k$. Then by Theorem 3.4 we know that for any set T of $m_k - |I_H|$ highest vertices of I_L, there exists a schedule for G and $(m_k, \cdots, m_d)$ starting with $I_H \cup T$. This implies that there exists a schedule for G' and $(m_{k+1}, \cdots, m_d)$. Note that at Step 2.1.2 $(S)_j = I_H \cup T$ and $(G) \xrightarrow{(S)_j} (G')$. This completes the proof of the loop invariant and the proof of correctness of Algorithm 5.2.

Proof of time bound. First, we create the data structures A and B in time and space $O(n^{m-1})$. Represent I_H as a doubly linked list. Implement I_L as an array of linked lists, where the linked list $I_L(h)$ contains all vertices of I_L of the height h.

Step 1. Evaluate the function λ for all equivalence classes $[I, 0]$ of G. By Lemma 5.3 this can be done in time and space $O(n^{m-1})$. Create all the above data structures, and evaluate μ and λ in the same time bound.

Step 2. We want to show that the loop can be implemented in time $O(n)$.

Case $|I_H| > m_k$. *Step* 2.1.1. Since $|I_H| < m$ and m is constant, there is only a constant amount of subsets R of I_H such that $|R| = m_k$. For each set R we can determine in constant time whether $\lambda(I', w') \leqq d - k$. Note that we know $\lambda(I', 0)$ and therefore by Lemma 5.2, $\lambda(I', w')$ can be determined in constant time. We conclude that Step 2.1.1 can be implemented in constant time.

Step 2.2. In the case $|I_H| > m_k$. Step 2.2 can be easily implemented in overall time $O(n)$. By Property B1 of data structure B, I'_H and μ' can be determined in constant time. Note that $(S)_j \subseteq I_H$. To determine I'_L we look at all immediate successors of the vertices of $(S)_j$. If such an immediate successor has height at most μ', then we add it to the appropriate list of I_L in constant time. Since each vertex gets added exactly once to the array of list I_L, this costs overall time $O(n)$.

Case $|I_H| \leqq m_k$. *Step* 2.2.1. By property M2 of the median we know that G has at least m components of height at least $\mu - 1$. Exactly $|I_H|$ of these components have height bigger than μ and therefore, G has at least $m - |I_H| \geqq m_k - |I_H|$ components of height μ and $\mu - 1$. Thus I_L has at least $m_k - |I_H|$ vertices in the lists $I_L(\mu)$ and $I_L(\mu - 1)$, and the set T of Step 2.1.2 can be found in constant time.

Step 2.2. In the case $|I_H| \leqq m_k$ we do Step 2.2 in two steps:

(i) $(I_H, I_L, \mu) \xrightarrow{T} (I_H, \bar{I}_L, \bar{\mu})$,

(ii) $(I_H, \bar{I}_L, \bar{\mu}) \xrightarrow{I_H} (I'_H, I'_L, \mu')$.

That is, we first remove the set T and determine $\bar{I}_L$ and $\bar{\mu}$, and then we remove the set I_H and determine I'_H, I'_L, μ'. Note that $(S)_j = I_H \cup T$. The reason why we can do Step 2.2 in two steps is that $L(G)$ and $H(G)$ are disjoint.

To show that Step (i) can be done in overall time $O(n)$, we observe that T can be removed from I_L in constant time. Note that T is a set of $m_k - |I_H|$ highest vertices of lists $I_L(\mu)$ and $I_L(\mu-1)$. To find $\bar{I}_L$ we insert all immediate successors of the vertices of T into the appropriate list of I_L. Since each vertex gets added at most once this can be done in overall time $O(n)$. To determine $\bar{\mu}$ we observe that $\bar{\mu} = \mu - 1$ if T contains all vertices of $I_L(\mu)$ and $\bar{I}_L(\mu-1)$; otherwise $\bar{\mu} = \mu$. Note that if $\bar{\mu} = \mu - 1$ then T contains all vertices of $I_L(\mu)$. Thus $H(G)$ and therefore I_H does not change when T is removed from G.

We showed already that Step 2.2(ii) can be done in overall time $O(n)$ (see implementation of Step 2.2 in the case where $|I_H| > m_k$).

This completes the proof of the time bound of Algorithm 5.2. Note that the expensive part was to evaluate the function λ in time $O(n^{m-1})$ retrieving a schedule is linear. This also completes the proof of Theorem 5.2. □

We now apply Theorem 5.2 to find an optimal schedule for an outforest.

COROLLARY 5.1. *Let G be an outforest and M be a profile of constant breadth m. Then an optimal schedule for G and M can be found in time $O(n^{m-1} \log m)$ and $O(n^{m-1})$ space.*

Proof. We do a binary search to determine

$$\min(\{d' | d' \leqq d \text{ and there exists a schedule for } G \text{ and } (m_1, \cdots, m_{d'})\}).$$

For every $d' \leqq d$ we can, by Theorem 5.2, decide in time and space $O(n^{m-1})$ whether there exists a schedule for G and $(m_1, \cdots, m'_d)$. Since we can assume that $d \leqq n$, we have to do this $O(\log n)$ times during the binary search. This completes the proof of the $O(n^{m-1} \log n)$ time bound. □

COROLLARY 5.2. *Let G be an inforest and M be a profile of constant breadth m. Then an optimal schedule for G and M can be found in time and space $O(n^{m-1})$.*

Proof. Since G is an inforest, G^R is an outforest. We apply Theorem 5.2 to the outforest G^R and M^R and find a schedule for G^R and $(m_{\lambda(I,w)}, \cdots, m_1)$ in time and space $O(n^{m-1})$, where $[I, w]$ is the equivalence class of G^R of which G^R is an element of. Interpreting the definition of the function λ (see Definition 5.2) we see that $\lambda(I, w)$ is the length of an optimal schedule for G and M. We used the same trick to prove Corollary 4.1. □

To prove our time bound for an opposing forest we use the following result of [GJ83].

THEOREM 5.3. *A schedule for an opposing forest fitting a straight profile of breadth m and length d can be found in time $O(d^{m-1} t(n, m, d))$ and space $O(s(m, n, d) + m)$, where $t(n', m', d')$ and $s(n', m', d')$ are the time and space, respectively, that it takes to find a schedule for an inforest with n' vertices and a nondecreasing profile of constant breadth m' and length d'.*

Proof. Corollary 2.2.1 of [GJ83]. □

We now combine Corollary 5.2 with Theorem 5.3:

THEOREM 5.4. *Let G be an opposing forest and M a straight profile of constant breadth m. Then a schedule S for G fitting M can be found in time $O(n^{2m-2})$ and space $O(n^{m-1})$.*

Proof. Let $t(n, m, d)$ and $s(n, m, d)$ be defined as in Theorem 5.3. By Corollary 5.2 we know that $t(n, m, d) = O(n^{m-1})$ and $s(n, m, d) = O(n^{m-1})$. Applying Theorem 5.3 we follow that it takes time $O(d^{m-1} n^{m-1})$ and space $O(n^{m-1})$ to find a schedule

for G fitting M. We can easily assume that $d < n$. Otherwise it is trivial to find a schedule for G fitting M. Using the fact that $d < n$, we get the $O(n^{2m-2})$ time bound. □

Note that Corollary 5.2 gives a more general result than we need to prove the above theorem. The $O(n^{m-1})$ time bound is for arbitrary profiles of constant breadth m and not only for nondecreasing profiles. Furthermore, in Corollary 5.2 we showed that one can find an optimal schedule in time and space $O(n^{m-1})$ and not just any schedule that fits the profile.

REFERENCES

[AH74] A. V. AHO, J. E. HOPCROFT AND J. D. ULLMAN, *The Design and Analysis of Computer Algorithms*, Addison-Wesley, Reading, MA, 1974.

[Br81] J. BRUNO, *Deterministic and stochastic scheduling problems with treelike precedence constraints*, NATO Conference, Durham, England, July 1981.

[CG72] E. G. COFFMAN, JR. AND R. L. GRAHAM, *Optimal scheduling for two-processors systems*, Acta Inform., 1 (1972), pp. 200-213.

[Co76] E. G. COFFMAN, JR., ed., *Computer and Job Shop Scheduling Theory*, John Wiley, New York, 1976.

[DW84a] D. DOLEV AND M. K. WARMUTH, *Scheduling flat graphs*, Reseach Report 84-04, Hebrew Univ., Jerusalem, March 1984.

[DW84b] ———, *Scheduling precedence graphs of bounded height*, J. Algorithms 5 (1984), pp. 48-59.

[FK71] M. FUJII, T. KASAMI AND K. NINOMIYA, *Optimal sequencing of two equivalent processors*, SIAM J. Appl. Math., 17 (1969), pp. 784-789; *Erratum*, 20 (1971), p. 141.

[Ga82] H. N. GABOW, *An almost linear algorithm for two processor scheduling*, J. ACM, 29 (1982), pp. 766-780.

[Ga81] ———, *A linear-time recognition algorithm for interval dags*, Inform. Proc. Lett., 12 (1981), pp. 20-22.

[GJ79] M. R. GAREY AND D. S. JOHNSON, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman, San Francisco, 1979.

[GJ83] M. R. GAREY, D. S. JOHNSON, R. E. TARJAN AND M. YANNAKAKIS, *Scheduling opposing forests*, this Journal, 4 (1983), pp. 72-93.

[GL79] R. L. GRAHAM, E. L. LAWLER, J. K. LENSTRA AND A. H. G. RINNOOY KAN, *Optimization and approximation in deterministic sequencing and scheduling: A survey*, Ann. Discr. Math., 5 (1979), pp. 287-326.

[Go76] D. K. GOYAL, *Scheduling series parallel structured tasks on multiprocessor computing systems*, Technical Report CS-76-034, Dept. of Computer Science, Washington State Univ., Pullman, September, 1976.

[Hu61] N. C. HU, *Parallal sequencing and assembly line problems*, Oper. Res., 9 (1961), pp. 841-848.

[LR76] J. K. LENSTRA AND A. H. G. RINNOOY KAN, *Complexity of scheduling under precedence constraints*, Oper. Res., 26 (1976), pp. 22-25.

[LT79] E. L. LAWLER, R. E. TARJAN AND J. VALDES, *The recognition of series parallel digraphs*, Proc. 11th Annual Symposium on Theory of Computing, Atlanta, GA, April 30-May 2, 1979, pp. 1-12.

[Ma81] E. W. MAYR, *Well structured parallel programs are not easier to schedule*, Technical Report STAN-CS-81-880, Dept. of Computer Science, Stanford Univ., Stanford, CA, September, 1981.

[PY79] C. H. PAPADIMITRIOU AND M. YANNAKAKIS, *Scheduling interval-ordered tasks*, SIAM J. Comput., 10 (1979), pp. 405-409.

[Ul75] J. D. ULLMAN, *NP-complete scheduling problems*, J. Comput. System Sci., 10 (1976), pp. 384-393.

[Wa81] M. K. WARMUTH, *Scheduling on profiles of constant breadth*, Ph.D. Thesis, Dept. Computer Science, Univ. Colorado, Boulder, August 1981.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 4, October 1985

015

CONVEX SETS AND NONDESTRUCTIVE ASSAY*

YAKOV BEN-HAIM†

Abstract. In the nondestructive assay of sparsely dispersed identical particles, one strives to resolve the number of particles contained in the sample. A perfect assay system can distinguish any spatial configuration of k particles from any spatial configuration of n particles, where $k \neq n$. Let F be the set of all physically realizable measurements of a single particle. It is shown that if F has a nonempty interior, then perfect assay capability is unachievable. When F is a convex set, sufficient conditions are developed which allow efficient analytical evaluation of the degree of resolving power of the assay system. Some results for nonconvex F are also presented. When completely unambiguous resolution is either unnecessary or impossible, one characterizes the resolution of the system by the probability of not distinguishing k from n particles. It is shown that the probability of not distinguishing k from n particles provides an upper bound for the probability of not distinguishing other pairs of numbers of particles.

Key words. convex sets, nondestructive assay, probabilistic nondestructive assay, spatially random materials

AMS(MOS) subject classifications. 52A20, 52A40

1. Motivation. Considerable effort has been devoted to developing the technical capability to measure the number and size of particulate species randomly embedded in the matrix of a container when direct access to the contents of the container is impossible [1]–[5]. Such nondestructive assay presents challenging problems even when it is known that all the particles are identical. An ideal assay system can distinguish any spatial configuration of n particles from any spatial configuration of k particles, when $n \neq k$. In practice many constraints prevent the realization of an ideal assay system. A rigorous basis for achieving the best possible assay system design is lacking, and the design of assay systems has been largely by trial and error. In this paper we shall develop an analytical tool of practical utility in the design of nondestructive assay systems.

Consider a small particle embedded in the contents of a container. Located outside the container we have a set of m measuring devices which are sensitive to the position of the particle. Let the readings obtained from these sensors comprise the m components of a real vector f, called the vector response function. This vector varies as the position of the particle is changed. Let the *response set* F be the set of all possible values which f may obtain.

Consider n different particle positions within a sample container, and let f^i represent the vector response obtained from a sample containing only a single particle, which is located at position i. For systems with sufficiently low number density of small particles, the vector response obtained from a sample with one particle at each of the n positions is $f^1+\cdots+f^n$. Thus the capability of an assay system to resolve n from k particles may be stated as

$$\sum_{i=1}^{k} f^i \neq \sum_{i=1}^{n} g^i \quad \text{for all } f^i,\ g^i \text{ in } F. \tag{1}$$

For an assay system capable of complete resolving power, (1) holds for all $n > k \geqq 0$. Partial resolving power is specified by stating the values of n and k for which (1)

* Received by the editors September 21, 1982, and in revised form July 2, 1984. This work was supported in part by the Technion VPR Fund—Lawrence Deutsch Fund.

† Department of Nuclear Engineering, Technion-Israel Institute of Technology Haifa, Israel.

holds. For instance, one may wish to resolve a single particle from any greater number of particles. Thus (1) must hold for $k=1$, $n>1$. Whatever is the desired degree of resolving power, the system must be designed so that the response function f satisfies (1) for the appropriate values of n and k. Typically, (1) must be satisfied for a large number of values of n and k. If we wish to employ this specification as a practical guideline for the design of nondestructive assay systems, it is desirable to reduce this large group of equations to an equivalent but smaller group. In § 2 we develop such "reduction theorems". We begin our considerations by requiring F to be a convex set. Some results for nonconvex F are also presented.

The results of § 2 are deterministic in the sense that when (1) is satisfied for certain n and k, then "any" spatial distribution of n particles is distinguishable from "any" spatial distribution of k particles. It is sometimes useful or necessary to relax this requirement, and not require distinguishability of very unlikely spatial distributions of n or k particles. The assay system is then designed to assure a specified upper limit for the probability of occurrence of indistinguishable spatial distributions of n or k particles. Section 3 is devoted to the development of probabilistic reduction theorems. Section 4 briefly discusses the utility of the theorems presented.

2. Deterministic reduction theorems.

2.1. Convex sets. Equation (1) is an unwieldy characterization of the design requirements for the set F because it comprises a large or infinite number of relations. We shall aim at reducing this characterization to one composed of a small number of relations. We begin by recording some basic results which will be useful later on.

DEFINITION 1. Let F be a subset of the m-dimensional Euclidean space E^m. For any positive integer n, we define F_n as

$$F_n = \left\{ g: g = \sum_{i=1}^{n} f^i, f^i \in F \right\}.$$

We see from this definition that (1) can be expressed as

$$F_k \cap F_n = \varnothing \tag{1a}$$

where F represents the set of all possible vector responses from a single particle and $\varnothing$ is the null set. Thus (1a) states that any spatial distribution of k particles can be distinguished from any spatial distribution of n particles.

DEFINITION 2. Given the subset F of E^m, for any real number a, we define aF as the set formed by multiplying each element of F by a. That is,

$$aF = \{g: g = af, f \in F\}.$$

If F is convex then so is F_n, and $F_n = nF$. From this we see that, when F is convex, the condition for distinguishability of n from k particles becomes

$$kF \cap nF = \varnothing. \tag{1b}$$

LEMMA 1. *For any subset F of E^m and any nonzero numbers a and b,*

(i) $aF \cap bF = \varnothing$

if and only if

(ii) $F \cap \frac{b}{a} F = \varnothing.$

Proof. If (i) is not true, then there are elements f and g of F such that

$$af = bg$$

which implies that

$$f = \frac{b}{a} g.$$

Hence (ii) is not true. Thus (ii) implies (i). This argument can be reversed to prove the converse. Q.E.D.

The following definition is fundamental.

DEFINITION 3. Let G be a nonempty subset of E^m. The *expansion of* G, if it exists, is denoted $e(G)$ and is defined by

$$e(G) = \sup \{z \in E^1 : G \cap zG \neq \varnothing\}.$$

When the expansion of G exists and when there are elements f and g of G such that

$$g = e(G)f$$

then G is said to be *self-expanded.*

It is evident that an unbounded set may have a finite expansion and may be self-expanded. For example, consider the following set in the plane:

$$G = \{(x, y) : y = 1 - x, x \in E^1\}.$$

Clearly G is self-expanded and $e(G) = 1$.

While the above set is closed, it can be seen that closure is not a sufficient condition for self-expansion. As an example, consider the following two curves in polar coordinates (r, t) in the plane:

$$r_1(t) = \frac{1}{t}, \qquad r_2(t) = \frac{2 - 4t/\pi}{t},$$

for $0 < t \leqq \pi/4$. Define the set

$$G = \{(r, t) : r_1(t) \leqq r \leqq r_2(t), 0 < t \leqq \pi/4\}$$

G is a closed set. Also

$$\frac{r_2(t)}{r_1(t)} = 2 - \frac{4t}{\pi} \xrightarrow{t \to 0} 2,$$

which implies that $e(G) = 2$, and yet G is not self-expanded.

The property of self-expansion is exploited in some of the proofs to follow. It is thus important to establish that a sufficiently broad class of sets are self-expanded. We are interested in response sets, which will almost invariably be both closed and bounded. The next lemma shows that such subsets of E^m are self-expanded, if the expansion exists at all.

It is important to note that expansion is not invariant to translation, and will even cease to exist if the translation causes the set to include the origin. Thus while the expansion of a set is related to its width and to its shape in general (translation-invariant properties), the expansion is distinct from these properties. This has a clear physical meaning when studying the expansion of a response set. The absolute magnitude of the response is as important as the range of responses, in determining the expansion of the set and the resolution-capability of the assay system.

LEMMA 2. *Let G be a nonempty compact subset of E^m. If the expansion of G exists, then G is self-expanded.*

Proof. We require some basic facts from metric topology [8]. Let $f(g)$ represent the distance from the origin to any point g in E^m. Then $f(g)$ is real and continuous in E^m. Also, every restriction of a continuous function to a subset D of its domain is continuous on D. Thus the restriction of f to G is continuous on G. Also, every continuous function on a compact domain achieves a minimum value and a maximum value on its domain.

Since $e(G)$ exists, G does not contain the origin. For every element g in G, define $R(g)$ as the ray from the origin through g. Also define the set

$$I(g) = R(g) \cap G$$

which is nonempty for any g in G. Since $R(g)$ is closed and G is compact, $I(g)$ is the closed subset of a compact set and thus compact.

Now, since $f(g)$ is continuous on $I(g)$ for any g in G and since $I(g)$ is compact, $f(g)$ achieves a minimum and a maximum value on $I(g)$. Define

$$u(g) = \frac{\text{maximum}_{h \in I(g)} f(h)}{\text{minimum}_{h \in I(g)} f(h)}.$$

This ratio exists for any g in G since G does not contain the origin.

Let z denote the expansion of G. It is evident that

$$z = \sup \{u(g) : g \in G\}.$$

Thus, for any $x > 0$ there is a g in G such that

$$0 \leqq z - u(g) < x.$$

Now consider the sequence of points in G

$$S = \{g_1, g_2, \cdots\}$$

satisfying

$$0 \leqq z - u(g_n) < \frac{1}{n}, \qquad n = 1, 2, 3, \cdots.$$

Since G is compact it is a bounded subset of E^m, and thus contained in some cubic m-cell (hypercube) [8]. Thus G can be covered by a finite number of m-cells of arbitrarily small size. Hence S contains an infinite subsequence which converges to a point $\tilde{g}$ for which

$$z = u(\tilde{g}).$$

Since G is closed, $\tilde{g}$ is an element of G. As noted before, f achieves a minimum value and a maximum value on $I(\tilde{g})$. Let these extreme values occur at points h_1 and h_2, respectively. Thus

$$f(h_2) = zf(h_1)$$

which shows that G is self-expanded. Q.E.D.

LEMMA 3. *If F is a nonempty compact convex subset of E^m, then for $a > 1$,*

(i) $F \cap aF = \varnothing$

if and only if the expansion of F exists and

(ii) $e(F) < a$.

Proof. Suppose that (i) does not hold. Then there are elements f and g of F such that

$$f = ag$$

which implies that either the expansion does not exist or, if it exists, that

$$e(F) \geqq a$$

which contradicts (ii). Thus if the expansion exists and if (ii) holds, then (i) must also be true. Now suppose that (ii) does not hold. Recall that, by Lemma 2, F is self-expanded if the expansion exists. Thus if the expansion exists but is not less than a, or if the expansion does not exist, there must be a number $b \geqq a$ and there must be elements f and g of F such that

$$f = bg.$$

Since F is convex,

$$h = \frac{a-1}{b-1} f + \frac{b-a}{b-1} g \in F.$$

Combining the previous two relations yields

$$h = ag.$$

Hence (i) is not true. Thus if (i) holds, (ii) must also be true. Q.E.D.

LEMMA 4. *If F is a nonempty compact convex subset of E^m, then for $a > 1$,*

(i) $F \cap aF = \varnothing$

implies

(ii) $F \cap bF = \varnothing$ *for all* $b \geqq a$.

Proof. Assume that (i) is true. Then Lemma 3 implies that the expansion of F exists and

$$e(F) < a.$$

Hence $b > e(F)$, so Lemma 3 implies (ii). Q.E.D.

Now we are ready to prove our basic deterministic reduction theorem for convex sets.

THEOREM 1. *If F is a nonempty compact convex subset of E^m, then for positive real numbers n, k, r and s,*

(i) $nF \cap (n+k)F = \varnothing$

implies

(ii) $rF \cap sF = \varnothing$ *for any* $s/r \geqq (n+k)/n$.

In other words when n, k, r and s are integers this theorem states that, if n particles are always distinguishable from $n+k$ particles then r particles are always distinguishable from s particles for

$$\frac{s}{r} \geqq \frac{n+k}{n}.$$

Proof. By Lemma 1, statement (i) implies

$$F \cap \frac{n+k}{n} F = \varnothing$$

which by Lemma 4 implies

$$F \cap \frac{s}{r} F = \varnothing \quad \text{for any } \frac{s}{r} \geqq \frac{n+k}{n},$$

which by Lemma 1 implies (ii). Q.E.D.

This theorem can be given a geometrical interpretation, to which we have already alluded by introducing the concept of expansion. Referring to Definition 2, we see that, for $a > 1$, aF can be viewed as a magnified image of F. Likewise $anF \cap a(n+k)F$ is a magnified image of $nF \cap (n+k)F$. It is evident that the image is a null set if and only if the object is null, regardless of whether or not F is convex. Now suppose that F is convex and disjoint from a certain magnification aF. This means that aF is "projected" entirely beyond the set F. Hence greater magnification, say bF for $b > a$, will also be disjoint from F. A few simple examples will show that this is not necessarily true if F is not convex.

Let F be a union of two closed intervals on the real line:

$$F = \{[2, 3], [7, 8]\}.$$

Inspection shows that $F \cap 2F = \varnothing$ and yet $F \cap 3F \neq \varnothing$. In this example F is not a connected set. However, the following example demonstrates that connectedness is not sufficient to obtain the results of Theorem 1. Let F be the curve in E^3 defined in cylindrical coordinates by the following parametric representation:

$$r = -\sin \pi t, \qquad \theta = \pi t, \qquad z = t,$$

for $1 \leqq t \leqq 2$. It can be seen that, for $a > 1$,

$$F \cap aF = \varnothing \quad \text{for } a \neq 2$$

while

$$F \cap aF \neq \varnothing \quad \text{for } a = 2.$$

As a final example we note that the convexity of F is not necessary for the results of Theorem 1. For example, consider the set in E^2 comprised of two tangent circles and their interiors, whose centers lie on a ray from the origin. Theorem 1 is true for this set even though the set is not convex.

Theorem 1 is the basic "reduction theorem" for convex sets, since it establishes a single condition on F which is sufficient for an infinite number of different equations of the form of (1)-(1b) to hold. We shall now demonstrate that for a certain class of not necessarily convex sets, there are always an infinite number of relations as in (1a) which do not hold. The following definition of the interior of a set will facilitate the next theorem.

DEFINITION 4. The interior of the subset F of E^m is the set in (F) whose elements f have the property: given any g in E^m there is a number $x(g) > 0$ such that $f + tg \in \text{in}\,(F)$ if $|t| < x$.

THEOREM 2. *If F is a subset of E^m with a nonempty interior, then there is a positive integer $\tilde{n}$ such that*

$$F_{nq} \cap F_{nq+p} \neq \varnothing \quad \textit{for all integers } q \geqq 1,\ 0 \leqq p \leqq q,\ n \geqq \tilde{n}.$$

Proof. Choose an element f^i from $\text{in}(F)$ and let f^j be any other element of F. By the definition of the interior there is a number $x > 0$ such that

$$f^i + \frac{p}{qn} f^j = f^k \in F, \quad \frac{1}{n} < x, \quad q \geqq 1, \quad 0 \leqq p \leqq q.$$

Hence

$$nqf^i + pf^j = nqf^k$$

and the theorem is proven for $\tilde{n} > 1/x$. Q.E.D.

2.2. Nonconvex sets. In this section we shall develop two reduction theorems for nonconvex sets. For any set F, let ch (F) represent the convex hull of F.

THEOREM 3. *Let F be a nonempty subset of E^m, and let $C = \text{ch}(F)$. Then for positive integers n and k,*

(i) $nC \cap (F_n + kC) = \varnothing$

implies that

(ii) $F_r \cap F_{r+jk} = \varnothing$

for all $r = 1, 2, \cdots, n$ and $j = 1, 2, 3, \cdots$.

Proof. (a) First we prove this theorem for $r = n$. Suppose that (ii) does not hold. Then there are elements f^i and g^i of F such that

(iii) $\sum_{i=1}^{n} f^i = \sum_{i=1}^{n+jk} g^i$.

Since C is the convex hull of F it follows that

$$\frac{1}{n}\sum_{i=1}^{n} f^i \in C \quad \text{and} \quad \frac{1}{n}\sum_{i=1}^{n} g^i \in C.$$

Thus by the convexity of C,

$$c = \frac{1}{jn}\sum_{i=1}^{n} f^i + \frac{j-1}{jn}\sum_{i=1}^{n} g^i \in C = \frac{1}{jn}\sum_{i=1}^{n} (f^i - g^i) + \frac{1}{n}\sum_{i=1}^{n} g^i.$$

Employing (iii), this relation becomes

$$c = \frac{1}{nj}\sum_{i=1}^{jk} g^i + \frac{1}{n}\sum_{i=1}^{n} g^i.$$

There are k elements d^i of C such that

$$\frac{1}{j}\sum_{i=1}^{jk} g^{n+i} = \sum_{i=1}^{k} d^i.$$

Hence the previous relation becomes

$$nc = \sum_{i=1}^{k} d^i + \sum_{i=1}^{n} g^i.$$

This contradicts (i). Thus if (ii) is false, then (i) is also false and we have proven the theorem for $r = n$.

(b) Now suppose, for some positive integer $r < n$ and some $f \in F_r$ and some $g \in F_{r+jk}$, that

$$f = g.$$

Then for any h in F_{n-r}

$$h + f = h + g$$

which implies that (ii) is false for $r = n$. Thus if (ii) holds for $r = n$, then (ii) holds for $r < n$, which completes the proof. Q.E.D.

We can obtain a much stronger result by imposing a certain restriction on the set F. This requires the following definitions. For any set F, the boundary of F is the set of all points p for which every neighborhood of p intersects both F and its complement. The boundary of F is denoted bd (F).

DEFINITION 5. Let F be a nonempty compact subset of E^m. F is *self-bordered* if every element of the boundary of the convex hull of F is either in F or may be expressed as the convex combination of two elements of F.

The theorem of Caratheodory states that for any subset S of E^m, each element of $\text{ch}(S)$ may be expressed as the convex combination of $m+1$ or fewer elements of S [8], [9]. We see that the property of self-borderedness bears some relation to Caratheodory's theorem. The following lemma, which is a specialization of Caratheodory's theorem, illustrates this relationship. It shows, for example, that every compact subset of E^1 and E^2 is self-bordered, though in higher dimensions this need not be so.

LEMMA 5. *If F is a compact subset of E^m, then every element of the boundary of* $\text{ch}(F)$ *can be expressed as the convex combination of m or fewer elements of F.*

Proof. Let $C = \text{ch}(F)$. By the theorem of Caratheodory, every element of C can be expressed as the convex combination of $m+1$ or fewer elements of F. Suppose that g belongs to $\text{bd}(C)$ and that g can be represented by no fewer than $m+1$ elements of F. Let one such representation of g be

$$g = \sum_{i=1}^{m+1} a_i f^i, \quad \sum_i a_i = 1, \qquad a_i > 0, \quad i = 1, \cdots, m+1.$$

The positivity of the a_i's results from g not being representable as a convex combination of fewer than $m+1$ elements of F. Let F' represent the set

$$F' = \{f^1, \cdots, f^{m+1}\}.$$

If F' is an affinely independent set, then $\text{ch}(F')$ is a convex body [8]. Let h be in the interior of $\text{ch}(F')$, and hence in the interior of C. Let a representation of h as a convex combination of elements of F' be

$$h = \sum_{i=1}^{m+1} c_i f^i, \quad \sum c_i = 1, \quad c_i \geqq 0, \quad i = 1, \cdots, m+1.$$

Now, since the a_i are all positive, we can choose a positive number x which is small enough so that

$$a_i + x(a_i - c_i) \geqq 0, \qquad i = 1, \cdots, m+1.$$

Also,

$$\sum_{i=1}^{m+1} (a_i + x(a_i - c_i)) = 1.$$

Thus the point u defined by

$$u = \sum_{i=1}^{m+1} (a_i + x(a_i - c_i) f^i$$

belongs to $\text{ch}(F')$. Now u is an affine combination of g and h:

$$u = (1+x)g - xh.$$

Since $x > 0$, g is strictly between u and h:

$$g = \frac{1}{1+x} u + \frac{x}{1+x} h.$$

Thus, since u belongs to C and h belongs to the interior of C, g must be in the interior

of C. This contradicts g being in the boundary of C. Thus the supposition that F' is an affinely independent set is false.

Now we may proceed more or less as in the proof of the theorem of Caratheodory [8], [9]. Since F' is affinely dependent, there exist numbers $y_1, y_2, \cdots, y_{m+1}$, not all zero, such that

$$\text{(i)} \quad \sum_{i=1}^{m+1} y_i f^i = 0 \quad \text{and} \quad \sum_{i=1}^{m+1} y_i = 0.$$

It is evident that there is a value of t such that

$$t = \max \left\{ \frac{a_i}{y_i} : i = 1, \cdots, m+1, y_i < 0 \right\}.$$

Let us relabel the points of F', if necessary, so that

$$\text{(ii)} \quad t = \frac{a_{m+1}}{y_{m+1}}.$$

Then, from (i), we have

$$\text{(iii)} \quad g = \sum_{i=1}^{m+1} a_i f^i - t \sum_{i=1}^{m+1} y_i f^i = \sum_{i=1}^{m+1} (a_i - t y_i) f^i.$$

It is readily shown that the coefficients $a_i - t y_i$ are all nonnegative. Also,

$$\sum (a_i - t y_i) = 1.$$

Thus (iii) is a convex combination of g in terms of the elements of F'. However, from (ii) we see that $a_{m+1} - t y_{m+1} = 0$, so (iii) is in fact a convex combination of only m elements of F'. This contradicts the supposition that g can be represented by no fewer than $m+1$ elements of F. Hence this supposition is false and the theorem is proved. Q.E.D.

By employing the property of self-borderedness we are able to prove a stronger version of Theorem 3.

THEOREM 4. *Let F be a nonempty compact and self-bordered subset of E^m, and let $C = \text{ch}(F)$. Then for positive integers n and k*

$$\text{(i)} \quad nC \cap (F_n + kC) = \varnothing$$

implies that

$$\text{(ii)} \quad F_r \cap F_s = \varnothing \quad \textit{for any} \quad \frac{s}{r} \geqq \frac{n+k}{n}.$$

Proof. (a) First we prove that, since F is self-bordered,

$$\text{(iii)} \quad \text{bd}((n+k)C) \subset F_n + kC.$$

For any element c belonging to the boundary of $(n+k)C$ there is an element d in the boundary of C such that

$$c = (n+k)d.$$

Since F is self-bordered, there are elements f and g in F and there is a number $0 \leqq a \leqq 1$ such that

$$\begin{aligned} c &= (n+k)(af + (1-a)g) \\ &= (n+1)af + (n+1)(1-a)g + (k-1)(af + (1-a)g). \end{aligned}$$

There are nonnegative integers u and v and a number $0 \leqq b \leqq 1$ such that

$$(n+1)a = u+b \quad \text{and} \quad (n+1)(1-a) = v+1-b$$

which implies that $u+v=n$. Hence

$$\begin{aligned} c &= uf+vg+bf+(1-b)g+(k-1)(af+(1-a)g) \\ &\in F_n + kC \end{aligned}$$

which completes the proof of (iii).

(b) Now (i) and (iii) imply that

$$nC \cap \operatorname{bd}((n+k)C) = \varnothing.$$

Thus, since C is the convex hull of a compact set it is compact and convex. Thus, either

(iv) $nC \cap (n+k)C = \varnothing$

or

(v) $nC \subset \operatorname{in}((n+k)C)$.

Let us suppose that inclusion (v) holds. Let 0 represent the coordinate origin: the point in E^m whose coordinates all equal zero. We shall show that inclusion (v) implies that 0 belongs to C. Suppose that 0 does not belong to C. Then since nC is a closed subset of E^m, 0 has a foot in nC [8]. That is, if $d(g)$ represents the distance from point g to the origin, there is an element f in nC such that

$$d(f) \leqq d(g) \quad \text{for all } g \in nC.$$

Since

$$d(f) = \left(\sum_{i=1}^{m} f_i^2 \right)^{1/2}$$

and since $f \neq 0$, we see that

$$d(f) < d\left(\frac{n+k}{n} f \right).$$

This implies that $((n+k)/n)f$ is not a foot of 0 in $(n+k)C$, since f belongs to $(n+k)C$ (by inclusion (v)). Let g be a foot of 0 in $(n+k)C$. Thus

$$d(g) < d\left(\frac{n+k}{n} f \right)$$

which implies that

$$d\left(\frac{n}{n+k} g \right) < d(f).$$

But since $(n/(n+k))g$ belongs to nC, f cannot be a foot of 0 in nC. But since f is by definition a foot of 0 in nC, the supposition that 0 does not belong to C must be false.

Now let us continue to examine the supposition that inclusion (v) holds. Since 0 belongs to C, it is evident that

$$F_n \subset (F_n + kC)$$

which implies that

(vi) $(F_n + kC) \cap nC \neq \varnothing$

since

$$F_n \subset nC.$$

But inequality (vi) contradicts the condition of the theorem, (i). Thus the supposition that inclusion (v) holds must be false. Hence (iv) is true.

From (iv) and Theorem 1 we obtain

$$rC \cap sC = \varnothing \quad \text{for any } \frac{s}{r} \geqq \frac{n+k}{n}$$

from which the theorem results by the inclusions

$$F_r \subset rC \quad \text{and} \quad F_s \subset sC$$

which completes the proof. Q.E.D.

The crux of this proof is that when F is self-bordered the boundary of $(n+k)C$ is contained in $F_n + kC$ (inclusion (iii)). From this it follows that nC and $(n+k)C$ are disjoint (iv).

Since any set in E^1 or E^2 is self-bordered, Theorem 4 is true for any such set. However, in more than two dimensions examples of nonself-bordered sets can be found for which the inclusion in (iii) is not valid. For example, let F be the four vertices of a regular tetrahedron. Let the faces of this tetrahedron not contain the origin, and let it have one face perpendicular at its center to a ray from the origin. Let a denote the vertex furthest from the origin, and let b, c and d be the other three vertices. Let $C = \text{ch}(F)$. The set $F + C$ is the union of four tetrahedra, each pair of which touch at just one point. For a properly chosen distance of the point a from the origin, a will just lie in the triangle defined by the points $b+c$, $b+d$, $c+d$. This triangle is contained in $\text{bd}(C+C)$ but not in $F+C$. Thus

$$C \cap (F+C) = \varnothing$$

but

$$C \cap (C+C) \neq \varnothing.$$

On the other hand, self-borderedness is not necessary for Theorem 4 to hold. For example, it is not necessary for F to be self-bordered on all of its surfaces, in order for relation (iii) in part (a) of the proof to be valid.

3. Probabilistic reduction theorems.

3.1. Preliminaries. Our basic deterministic reduction theorem states that if any spatial distribution of n identical particles is distinguishable from any spatial distribution of $n+k > n$ identical particles, then r and s particles are always distinguishable if $s/r \geqq (n+k)/n$. This criterion of distinguishability is useful for characterizing the degree of resolution of the assay system. However, it sometimes occurs that not all spatial distributions of n and $n+k$ particles are distinguishable. In such a case one wishes to evaluate the probability of occurrence of indistinguishable spatial distributions, and to include this probability in characterizing the resolution capability of the assay system. In this section we shall develop several theorems for this purpose.

Our first task is to characterize the allowed class of spatial distributions of the assayed particles, and to define the type of function which will describe the probability of these spatial distributions.

Throughout the previous sections we have defined F as the set of all vector measurements obtainable, for a given assay system, from any positioning in the

container of a single assayed particle. We shall continue with this definition, and sometimes we shall refer to this set as F_1. Now let W_1 represent the σ-ring of all Borel subsets of F_1. The union of the elements of W_1 equals F_1, so (F_1, W_1) is a measurable space. Furthermore, let us suppose that we have a nonnegative measurable set function p_1 defined on the elements of W_1 such that $p_1(F_1) = 1$. Thus (F_1, W_1, p_1) is a probability space. We may envision the following procedure for choosing p_1. Each element of F_1 represents a specific elementary event (or set of events): the measurement obtained from a single particle located at a certain position (or any of a set of equivalent positions) in the assayed container. Elements of W_1 represent sets of events. We may choose p_1 so as to give physically meaningful statistical weight to those sets of events of practical interest. For further discussion of the calculation of p_1, see [2].

As in Definition 1, F_n represents the set of all vector measurements obtainable from any spatial distribution in the container of n identical particles. Let W_n represent the σ-ring of all Borel subsets of F_n. Thus (F_n, W_n) is a measurable space. Our aim at present is to define a probability measure p_n on the elements of W_n in terms of the probability measure p_1 defined on W_1. To do this we shall assume that the assayed particles are randomly and independently located in the container. This assumption is physically reasonable for many applications, and is analytically quite convenient.

The following considerations motivate the choice of p_n which we shall present in Lemma 7. Let $Q(x)$ be a probability distribution with density $q(x)$. Let $X_1, \cdots, X_n$ be independent random variables each with density q and distribution Q. The probability distribution for $X_n + \cdots + X_n$ is given by the convolution [7]:

$$Q_n(x) = \int Q_{n-1}(x-y) q_1(y)\, dy.$$

From this result we see that p_n—the probability measure on the set of n-particle measurements—should be related to p_1 by a convolution, if the particles are positioned randomly and independently.

Before we are able to define the measure-theoretic analog of the above convolution, we require a preliminary definition and lemma. We define the section of a set analogous to the concept employed for Cartesian products [6].

DEFINITION 6. Let E be a subset of F_{n+1} and let f be an element of F_1. Then the section of E with respect to f is the set of elements g of F_n for which $g+f$ belongs to E. That is,

$$E_f = \{g \in F_n : g + f \in E\}.$$

We shall adopt the convention that if E is the null set, then so is E_f.

LEMMA 6. *Let G_i, $i = 1, 2, 3, \cdots$ be a sequence of disjoint subsets of F_{n+1}. For any f in F_1*

(i) $(G_1)_f \cap (G_j)_f = \varnothing$ *for all* $i \neq j$,

(ii) $\bigcup_{i=1}^{\infty} (G_i)_f = \left(\bigcup_{i=1}^{\infty} G_i \right)_f$.

Proof. Part (i) is clearly true if either G_i or G_j is null. If neither is null, suppose there is a point z such that

$$z \in (G_i)_f \cap (G_j)_f$$

where $i \neq j$. Thus

$$f + z \in G_i \quad \text{and} \quad f + z \in G_j$$

which contradicts the disjointness of G_i and G_j.

Now to prove part (ii), suppose

$$z \in \left(\bigcup_{i=1}^{\infty} G_i \right)_f.$$

Thus

$$f+z \in \bigcup_{i=1}^{\infty} G_i.$$

Hence there is a set G_k such that

$$f+z \in G_k$$

which implies that

$$z \in (G_k)_f.$$

Consequently

$$\text{(a)} \qquad \left(\bigcup_{i=1}^{\infty} G_i \right)_f \subset \bigcup_{i=1}^{\infty} (G_i)_f.$$

Now suppose

$$y \in \left(\bigcup_{i=1}^{\infty} G_i \right)_f.$$

Hence there is a set G_k for which

$$y+f \in G_k.$$

Thus

$$y+f \in \bigcup_{i=1}^{\infty} G_i$$

which implies that

$$y \in \left(\bigcup_{i=1}^{\infty} G_i \right)_f.$$

Thus

$$\text{(b)} \qquad \bigcup_{i=1}^{\infty} (G_i)_f \subset \left(\bigcup_{i=1}^{\infty} G_i \right)_f.$$

Combining inclusions (a) and (b) completes the proof of part (ii). Q.E.D.

Now we are able to define p_n, and to prove that it is indeed a probability measure.

LEMMA 7. *Let E belong to W_n, and let p_1 be a probability measure on (F_1, W_1). For any $n \geqq 2$, the following recursive relation defines a probability measure on (F_n, W_n).*

$$p_n(E) = \int_{F_1} p_{n-1}(E_f)\, dp_1(f).$$

Proof. We begin by proving the theorem for $n=2$. It is evident that p_2 is a nonnegative extended real valued set function, since p_1 has these properties. First we must show that p_2 is countably additive. That is, given a disjoint sequence of sets G_i,

$i = 1, 2, 3, \cdots$ in W_2 we must show that

$$p_2\left(\bigcup_{i=1}^{\infty} G_i\right) = \sum_{i=1}^{\infty} p_2(G_i).$$

From the definition of p_2 we obtain

$$p_2\left(\bigcup_{i=1}^{\infty} G_i\right) = \int p_1\left(\bigcup_{i=1}^{\infty} G_i\right)_f dp_1(f).$$

Employing Lemma 6 and the countable additivity of p_1 we conclude that

$$\int p_1\left(\bigcup_{i=1}^{\infty} G_i\right)_f dp_1(f) = \int p_1\left(\bigcup_{i=1}^{\infty} (G_i)_f\right) dp_1(f)$$

$$= \int \sum_{i=1}^{\infty} p_1(G_i)_f \, dp_1(f).$$

Thus

$$p_2\left(\bigcup_{i=1}^{\infty} G_i\right) = \sum_{i=1}^{\infty} p_2(G_i).$$

To prove that p_2 is a probability measure on (F_2, W_2) we must show that $p(F_2) = 1$. For any f belonging to F_1, we note that $(F_2)_f = F_1$. Thus

$$p_2(F_2) = \int_{F_1} p_1((F_2)_f)\, dp_1(f) = \int_{F_1} p_1(F_1)\, dp_1(f) = 1$$

which completes the proof for $n = 2$. The proof by induction for higher values of n is analogous and will not be elaborated. Q.E.D.

We now prove a useful recursive relation between p_n and p_{n-1}. Before doing so we need the following standard definitions.

DEFINITION 7. *Let G and H be any two sets in E^m. The difference $G - H$ is the set of all elements of G which are not elements of H. Likewise, the sum $G + H$ is the set of all pairwise sums of elements of G with elements of H.*

LEMMA 8. *Let A belong to W_1 and let B belong to W_{n-1}. Then*

$$p_n(A+B) = p_1(A)p_{n-1}(B) + \int_A p_{n-1}((A+B)_f - B)\, dp_1(f)$$

$$+ \int_{F_1 - A} p_{n-1}((A+B)_f)\, dp_1(f).$$

Proof. From the definition of p_n in Lemma 7 we obtain

$$p_n(A+B) = \int_{F_1} p_{n-1}((A+B)_f)\, dp_1(f)$$

$$= \int_A p_{n-1}((A+B)_f)\, dp_1(f)$$

$$+ \int_{F_1 - A} p_{n-1}((A+B)_f)\, dp_1(f).$$

Since B and $(A+B)_f - B$ are disjoint and p_{n-1} is an additive set function we obtain

$$\int_A p_{n-1}((A+B)_f)\, dp_1(f) = \int_A p_{n-1}(B)\, dp_1(f) + \int_A p_{n-1}((A+B)_f - B)\, dp_1(f).$$

Combining this with the previous equation yields the desired result. Q.E.D.

3.2. Convex sets. The proof of the deterministic Theorem 1 rests on Lemmas 1 and 4. In order to generalize Theorem 1 we should seek measure-theoretic analogues of these results. While Lemma 4 has a simple and useful analogue, we shall see that the analogue of Lemma 1 is not so simple.

We shall have occasion to refer to the following nonnegative set function defined on elements G of W_1. Let G_n represent the n-fold sum of G with itself.

$$w_n(G) = \int_G p_{n-1}((G_n)_f - G_{n-1})\, dp_1(f) + \int_{F_1-G} p_{n-1}((G_n)_f)\, dp_1(f).$$

From Lemma 8 we see that $p_n(G_n)$ can be related to $p_1(G)$ as

$$p_n(G_n) = p_1(G)p_{n-1}(G_{n-1}) + w_n(G).$$

From this relation we see that, for arbitrary or even for convex G, $p_n(G_n)$ may be less than or greater than $p_1(G)$. In fact if G is not empty, the measure of G_n may be nonzero while the measure of G is zero. This is illustrated in Fig. 1 for G_2, where we see that the points x and y contribute to the measure of G_2 even though they belong to $F-G$. We see that a further restriction on G is needed in order to obtain an analogue of Lemma 1.

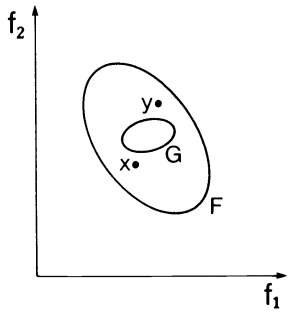


FIG. 1. *Schematic representation of a 2-dimensional response set, F, showing that $p_2(2G)$ depends on the measure of elements of $F-G$.*

A useful restriction is that both G and its complement relative to F be convex. Before developing the resulting analogue of Lemma 1, let us identify those situations for which such a restriction on G will be useful. Figure 2 shows the intersection of a

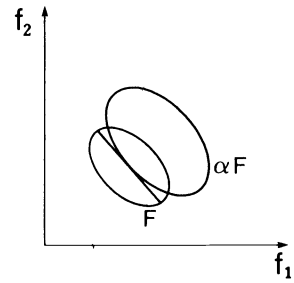


FIG. 2. *Schematic representation of 2-dimensional response sets, F and αF, whose intersection may be approximated by a convex set whose complement is also convex.*

convex set F with some multiple of F. Let this intersection be denoted H, and let G be the subset of F containing H and defined by a chord in F tangent to H. Both G and its complement are convex, since F is convex. The measure of G and its multiples will provide upper limits of the measure of H and its multiples. A reduction theorem for G and its multiples will provide an approximation to the measure of H and its multiples. We now prove an analogue of Lemma 1.

LEMMA 9. *Let F be a subset of E^m, let G be an element of W_1, let $G' = F - G$ and let G, G' and F be convex. Then*

$$p_n(nG) \leqq 1 - (1 - p_1(G))^n.$$

Proof. From Lemma 8 and the definition of w_n we obtain

$$w_n(G) + w_n(G') = p_n(nG) + p_n(nG') - p_1(G)p_{n-1}((n-1)G) - p_1(G')p_{n-1}((n-1)G').$$

Employing the convexity of G, G' and F and noting that

$$p_k(kG) + p_k(kG') = 1 \quad \text{for all } k \geqq 1,$$

the previous equation becomes

$$w_n(G) + w_n(G') = p_1(G) + p_{n-1}((n-1)G) - 2p_1(G)p_{n-1}((n-1)G).$$

Since w is a nonnegative set function, we conclude that $w_n(G)$ does not exceed the right-hand side of the previous equation. That is,

$$w_n(G) \leqq p_1(G) + p_{n-1}((n-1)G) - 2p_1(G)p_{n-1}((n-1)G).$$

For $n = 2$ we may combine this inequality with Lemma 8 to yield

$$p_2(2G) = p_1(G)^2 + w_2(G) \leqq 2p_1(G) - p_1(G)^2$$

which proves the lemma for $n = 2$. Now Lemma 8 and the above inequality on w_n can be used to complete the proof inductively. Q.E.D.

This result implies that, when G and F are convex, the measure of nG is zero if the measure of G vanishes. This is different from the situation where only G is convex, and arises from the fact that, since both G and G' are convex, there cannot be elements

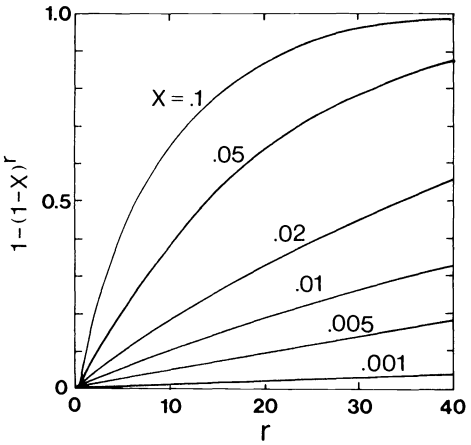


FIG. 3. *Graphical illustration of the upper bound of $p_n(nG)$ which is established by Lemma 9.*

of G' which sum together to yield an element of nG. That is, the situation depicted in Fig. 1 cannot occur.

We now present an analogue of Lemma 4, after which we will be ready to generalize Theorem 1.

LEMMA 10. *Let F be a subset of E^m and let G be any convex element of W_n. Then for $1 \leqq a < b$*

(i) $G \cap bG \subset G \cap aG$,

(ii) $p_n(G \cap bG) \leqq p_n(G \cap aG)$.

Proof. Part (i) is trivial if the left-hand intersection is null, so suppose that there is a point x belonging to $G \cap bG$. Thus there are elements f and g in G such that $x = f = bg$. Hence

$$x = \frac{b-a}{b-1} f + \frac{a-1}{b-1} bg = a\left(\frac{b-a}{a(b-1)} f + \frac{b(a-1)}{a(b-1)} g\right).$$

The last expression is a times an element of G since G is convex. Hence x belongs to aG, which concludes the proof of part (i). Part (ii) results from part (i) and the fact that a measure on a ring is a monotone set function [6]. Q.E.D.

We are now ready to present a partial generalization of Theorem 1.

THEOREM 5. *Let F be a convex subset of E^m, let G belong to W_1, let $G' = F - G$, let G and G' be convex and let $a \geqq 1$. If*

$$F \cap aF \subset G$$

then for any r and any $s \geqq ar$

$$p_r(rF \cap sF) \leqq 1 - (1 - p_1(G))^r.$$

Proof. From the conditions of the theorem we obtain

$$rG \supset r(F \cap aF) = rF \cap arF.$$

From Lemma 10 we find that

$$r(F \cap aF) \supset r\left(F \cap \frac{s}{r} F\right) = rF \cap sF.$$

By the monotonicity of the measure and by Lemma 9 we conclude

$$p_r(rF \cap sF) \leqq p_r(rG) \leqq 1 - (1 - p_1(G))^r$$

which completes the proof. Q.E.D.

This theorem is less than a complete generalization of Theorem 1 for two reasons. Firstly, it relates intersections with rF only to intersections with F rather than to nF. Secondly, it only provides an upper bound for the measure of these intersections. It does, however, provide useful information on the degree of distinguishability of r from s particles, on the basis of analysis of the set of one-particle measurements. If $p_1(G)$ is much less than 1, or if r is not too much greater than 1, then Theorem 5 provides an upper limit on the measure of $p_n(rF \cap sF)$ that is still much less than unity. Figure 3 illustrates this. This means that, for properly chosen G, the measure $p_1(G)$ yields an upper limit of the probability that r identical particles will be spatially distributed in the assayed container in such a way as to be indistiguishable from s identical particles.

3.3. Nonconvex sets. We now present one final probabilistic reduction theorem which is true for arbitrary sets F.

THEOREM 6. *Let F be a subset of E^m. Then for any positive integer k*

$$p_n(F_n \cap F_{n+k}) \leqq p_{n+1}(F_{n+1} \cap F_{n+k+1}).$$

Proof. We begin by proving the relation

$$\text{(i)} \quad F_1 + (F_n \cap F_{n+k}) \subset F_{n+1} \cap F_{n+k+1}.$$

This inclusion is trivial if $F_1 + (F_n \cap F_{n+k})$ is null, so suppose that it is not empty, and contains an element g. Then $g = f + h$, where f belongs to F_1 and h belongs to $F_n \cap F_{n+k}$. Since $h \in F_n$, we obtain $g \in F_{n+1}$. Likewise, since $h \in F_{n+k}$, we obtain $g \in F_{n+k+1}$. Thus

$$g \in F_{n+1} \cap F_{n+k+1}$$

which completes the proof of the inclusion. Now, from Lemma 8,

$$p_{n+1}(F_1 + (F_n \cap F_{n+k})) = p_n(F_n \cap F_{n+k}) + \int_{F_1} p_n((F_1 + (F_n \cap F_{n+k}))_f - (F_n \cap F_{n+k}))\, dp_1(f).$$

From inclusion (i) and the monotonicity of measure we have

$$p_{n+1}(F_1 + (F_n \cap F_{n+k})) \geqq p_{n+1}(F_{n+1} \cap F_{n+k+1}).$$

Combining the last two relations establishes the desired result. Q.E.D.

The practical utility of this theorem results from the following interpretation. The theorem asserts that the probability of n particles being spatially distributed in the assayed container in such a way as to be indistinguishable from $n+k$ particles, never exceeds the probability of $n+1$ particles being indistinguishable from $n+k+1$ particles, whether or not F is convex.

4. Summary and discussion. In the nondestructive assay of sparsely dispersed identical particles, one strives to resolve the number of particles contained in the sample. A perfect assay system can distinguish any spatial configuration of k particles from any configuration of n particles, where $k \neq n$. We can refer to the *resolving power* of an assay system, to qualitatively express the degree to which the system approaches this ideal. The set F comprises all possible values which the response function f may obtain. If F has a nonempty interior, then Theorem 2 states that the system can never have complete resolving power. However, when F is convex, Theorem 1 provides an efficient analytical criterion for evaluating the degree of resolving power of the assay system. One important attribute of the formulation of Theorem 1 is that a finite number of relations on elements of F are sufficient to determine an infinite number of relations specified in (1). Furthermore, by applying Theorem 1 for $n = 1, 2, \cdots, n_{\max}$, one obtains a criterion for assuring that any number of particles not exceeding $n_{\max}$ will be unambiguously resolvable. This is important in the analysis of sparse systems since in practice one is often able to ascertain an a priori upper limit to the number of particles which may be encountered in any measured container. In such a case, the capability to resolve a greater number of particles is unnecessary.

It may happen that complete resolution of the number of particles is not possible for any value of n. One then resorts to the probability of not distinguishing k from n particles. It often occurs in practice that the assayed particles are located randomly in the container. In such a situation the probability measure for the n-particle distributions can be related by convolution to the probability measure for one-particle distributions. This relation is expressed in Lemma 7. Theorem 5 then establishes an upper limit for the probability that r particles will be spatially distributed so as to be indistinguishable from s particles. While Theorem 5 requires the convexity of F, Theorem 6 does not. This last theorem establishes that the probability of not distinguishing n from $n+k$

particles does not exceed the probability of not distinguishing $n+1$ from $n+k+1$ particles.

Acknowledgment. The author is indebted to the helpful comments of Prof. Zvi Artstein.

REFERENCES

[1] Y. BEN-HAIM AND E. ELIAS, *Probabilistic interpretation of nondestructive assay of nuclear materials*, Ann. Nucl. Energy, 9 (1982), pp. 1-9.

[2] Y. BEN-HAIM, *Probabilistic nondestructive assay of radioactive waste*, Ann. Nucl. Energy, 10 (1983), pp. 57-64.

[3] A. KNOLL, A. NOTEA AND Y. SEGAL, *Probabilistic interpretation of nuclear waste assay by passive gamma technique*, Nucl. Tech., 56 (1982), pp. 351-360.

[4] T. GOZANI, *Active nondestructive assay of nuclear materials*, U.S. National Technical Information Service, Springfield, VA, NUREG/CR-0602, 1981.

[5] R. SHER AND S. UNTERMEYER, *The detection of fissionable materials by nondestructive means*, Amer. Nucl. Soc., LaGrange Park, IL, 1980.

[6] P. R. HALMOS, *Measure Theory*, Springer-Verlag, New York, 1974.

[7] W. FELLER, *An Introduction to Probability Theory and its Applications, Vol.* 2, John Wiley, New York, 1971.

[8] P. J. KELLY AND M. L. WEISS, *Geometry and Convexity*, John Wiley, New York, 1979.

[9] S. R. LAY, *Convex Sets and Their Applications*, John Wiley, New York, 1982.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 4, October 1985

016

AN $O(|V|^2)$ ALGORITHM FOR THE PLANAR 3-CUT PROBLEM*

DORIT S. HOCHBAUM† AND DAVID B. SHMOYS‡

Abstract. A 3-cut for a connected graph G is a set of edges which, when deleted, separate G into 3 components. In this paper we present an $O(|V|^2)$ algorithm to find the minimum 3-cut for a *planar* graph G.

AMS(MOS) subject classifications. F.2.2., G.2.1., G2.2

Given a connected (simple) graph $G=(V, E)$, a *k-cut* is a subset of edges E_1 such that the graph $G_1=(V, E-E_1)$ contains exactly k components. The problem of finding the smallest 3-cut is interesting, not only as an extension of the ordinary minimum (2-)cut problem, but also because of applications in cutting plane methods for the traveling salesperson problem. In this paper we present a polynomial-time algorithm to find a minimum size 3-cut for *planar* graphs. It is easy to see that the 3-cut problem is polynomial for planar graphs; since the minimum degree of a planar graph is at most five, there is a trivial cut of size 10. To find the minimum cut we can simply check all subsets of edges of size less than 10. To simplify notation, for a graph $G=(V, E)$, $|G|$ will be used to denote $|E|$; note that for a planar graph $|G|=O(|V|)$. Thus, the simple enumerative algorithm requires $O(|G|^{10})$ time. In this paper we present a simple efficient algorithm that requires only $O(|G|^2)$ time.

One simple approach for finding an optimal 3-cut that might be considered is the greedy approach; that is, choose a triple of vertices (v_1, v_2, v_3). First find a minimum (2-)cut between v_1 and v_2. This cut leaves v_3 in a component with either v_1 or v_2; suppose, without loss of generality, that v_1 lies in the same component as v_3. Next find the minimum cut in this component between v_1 and v_3. This yields a 3-cut, but is it an optimal one? A further extension of this would be to try this greedy heuristic for all ordered triples of vertices. Unfortunately, even in the planar case, this procedure does not yield this optimum solution. Consider the graph given in Fig. 1. The minimum 3-cut is the set of six edges separating the three square-like sets of four vertices each. Attempting to find such a cut by enumerating all possible triples of vertices, we check the triple (v_1, v_2, v_3) given in Fig. 1. Suppose that we find a minimum 2-cut

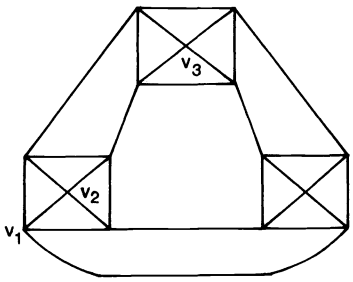


FIG. 1

* Received by the editors November 15, 1983, and in revised form June 28, 1984.

† School of Business Administration, University of California, Berkeley, California 94720. The research of this author was supported in part by the National Science Foundation under grant ECS-8204695.

‡ Department of Computer Science, Harvard University, Cambridge, Massachusetts 02138. The research of this author was supported in part by the National Science Foundation by a graduate fellowship and under grant MCS-8311422, and in part by DARPA order 4031, monitored by Naval Electronic System Command under contract N00039-C-0235.

between v_1 and v_2; one such cut consists of the four edges incident to v_1. Now find a minimum cut between v_2 and v_3; this cut consists of the three remaining edges incident to v_2. In total, we have found a 3-cut of size seven. By a careful examination of all cases, it is straightforward to verify that for any choice of 3 vertices, there is a cut that can be the result of this approach that is not optimal.

We will solve the minimum 3-cut problem in G by solving the corresponding problem in G^*, the dual graph of G. Note that the dual graph G^* might not be a simple graph; both self-loops and multiple edges can occur.

Recall that for a planar graph G, there is a 1-1 correspondence between 2-cuts in G and cycles in G^*; we want to generalize this idea. A cycle is a minimal graph with 2 faces. For each face of a subgraph of G^*, there are some vertices of G that are "contained" in that face that are separated from vertices "contained" in the other faces of the subgraph. This leads us to make the following observation.

Observation. G has a 3-cut of size k if and only if G^* contains a subgraph H with exactly three faces such that H has k edges.

Since G^* can be found in $O(|G|)$ time, we will focus attention on the problem of finding a minimum size subgraph with 3 faces. Thus, for the remainder of the paper, we will assume that we are given both G^*, and an embedding of it. We begin by examining the structure of possible optimal solutions. First of all, the optimal subgraph H may or may not be connected. Suppose that H is disconnected; since it contains exactly three faces, it must be precisely two disjoint cycles (see Fig. 2(a)). If H is connected then for a given embedding either the boundary of the exterior face includes all of the edges of H or it does not. By considering these cases separately, it is not hard to see that H must be one of the configurations depicted in Fig. 2(b) and 2(c).

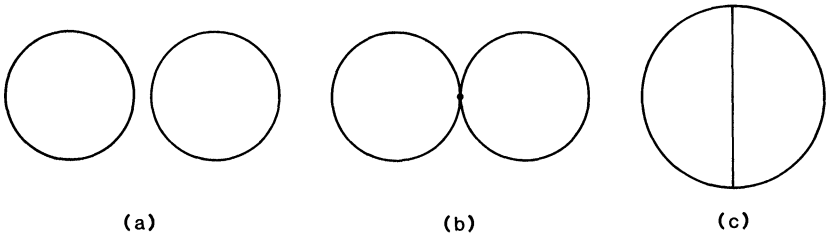


FIG. 2

We will show that a variant of the greedy approach works in either of the cases 2(a) and 2(b). Let $G \cdot e$ denote the graph formed from G by contracting the edge e, i.e., identifying the endpoints of e and deleting e. In terms of the dual graph, the variant of the greedy heuristic that we shall use is the following:

```
procedure greedy (G*)
  begin
    find a shortest cycle C1 in G*
    for all edges e in the cycle C1 do G* ← G* · e
    find a shortest cycle C2 in G*
    output C1 ∪ C2
  end
```

In this variant we avoid trying all triples of vertices, which of course improves the efficiency of the algorithm. Although this algorithm does not always give an optimal

solution, there are a number of cases in which it does work, including those cases when the optimal solution is of either of the configurations given in Fig. 2(a, b). In fact, our procedure to find an optimal 3-cut works by first performing *greedy* (G^*), and then performing a special procedure designed to handle the case depicted in Fig. 2(c). The following theorem justifies this approach.

THEOREM 1. *Let C_3 be any shortest cycle of the graph G^*. If there exists an optimal 3-face subgraph that is the union of 2 cycles with at most one vertex in common, then there exists some other cycle C such that C_3 and C share at most one vertex and $C_3 \cup C$ is an optimal 3-face subgraph.*

Proof. Suppose not. Let $C_1 \cup C_2$ be an optimal 3-face subgraph, where C_1 and C_2 have at most one common vertex. Note that by our assumptions, C_3 must have at least 2 vertices in common with C_i for either $i=1$ or 2. Recall that $|G|$ denotes the number of edges in G. Notice that a cycle C has $|C|$ vertices.

If C_i and C_3 are edge disjoint but have at least two vertices in common, then, by using Euler's formula $|F|=2+|E|-|V|$, we see that the number of faces of $C_i \cup C_3$ is at least

$$2+(|C_i|+|C_3|)-(|C_i|-(|C_3|-2))=4.$$

Therefore, $C_i \cup C_3$ contains a *proper* subgraph with 3 faces and fewer edges than $C_1 \cup C_2$, which is a contradiction.

If C_i and C_3 have an edge in common, then

$$|C_i \cup C_3| < |C_i|+|C_3| \leqq |C_1|+|C_2|.$$

Furthermore, once again using Euler's formula, it is not hard to see that $C_i \cup C_3$ must have at least 3 faces. Therefore, $C_i \cup C_3$ is a 3-face subgraph with fewer edges than $C_1 \cup C_2$, which is a contradiction. □

It is important to note that if the optimal 3-face subgraph consists of 2 cycles sharing at most one common vertex, Theorem 1 says that *any* shortest cycle is part of an optimal solution. Thus we may greedily contract it, and search for its partner.

Note that it is very simple to use breadth-first search (BFS) to find the shortest cycle of a graph that contains a specified vertex u. Recall that BFS partitions the vertex set of a graph into levels, and the edge set into cross edges and level edges. (For a comprehensive treatment of BFS, see [AHU].) Furthermore, for each vertex, each edge incident to it either goes to a vertex of the next level closer to the root, the same level or the next level further from the root; therefore, it is possible to partition the degree of any vertex v into the *up-degree*, *up* (v), the *cross-degree*, *cross* (v), and the *down-degree*, *down* (v), respectively. Note that it is possible to compute all three parameters as part of the BFS without increasing the order of the running time of the search algorithm. We detect the smallest cycle containing u by conducting a BFS from u until we find a vertex v that has *up* (v) + *cross* (v) $\geqq 2$. Since BFS requires only $O(|G|)$ time, the procedure outlined above requires only $O(|G|^2)$ time.

We next consider the case where the optimal solution is of the form depicted in Fig. 2(c). This graph may be viewed as a pair of points connected by three vertex-disjoint paths. The following result will be very useful in gaining a better perspective of the cases in which *greedy* (G^*) is successful.

THEOREM 2. *Suppose that there exists an optimal 3-face subgraph G^* which is the union of three paths P_1, P_2 and P_3, between vertices u and v. If one of the paths P_i is at least as long as the shortest cycle C in G^*, then greedy (G^*) delivers an optimal solution.*

Proof. Consider the solution produced by *greedy* (G^*), $C_1 \cup C_2$. Suppose without loss of generality that P_1 is at least as long as C_1, which is a shortest cycle of G^*.

There are two possible cases: either $C_2 = P_2 \cup P_3$, or $C_2 \neq P_2 \cup P_3$. In the first case the result is trivial since then it is clear that

$$|C_1 \cup C_2| = |C_1| + |C_2| \leqq |P_1| + (|P_2 \cup P_3|) = |P_1 \cup P_2 \cup P_3|.$$

If $C_2 \neq P_2 \cup P_3$ then after contracting C_1 the subgraph corresponding to $P_2 \cup P_3$ must still be a cycle (since not all of the edges of $P_2 \cup P_3$ have been contracted). Since C_2 is no longer than this cycle the total length of $C_1 \cup C_2$ is no more than that of $P_1 \cup P_2 \cup P_3$. □

We can use Theorem 2 to gain considerable insight into the structure of the optimal solution when *greedy* (G^*) does *not* deliver an optimal solution. Such optimal solutions, which must be of the type given in Fig. 2(c), can be characterized by the length of the three paths P_1, P_2 and P_3. More specifically, since *greedy* (G^*) always delivers a solution of size no more than 10, we know that the three path lengths sum to at most 9, where the longest is strictly shorter than the length of the shortest cycle. From these two conditions we know that the shortest path is of length at most three. This means that if we were conducting BFS from u and the optimal 3-face subgraph consisted of three disjoints paths from u to v, v must be, at most, in level three. (We will say from here on, that v is "labeled" 3.)

We will consider these cases, based on the length of the shortest cycle C_1 in G^*. We shall use the notation (x, y, z) to denote the configuration where P_1, P_2, P_3 have lengths x, y, z where $x \leqq y \leqq z$.

Case 1. $|C_1| = 1$. This case is vacuous since it is impossible for the longest path to have length 0.

Case 2. $|C_1| = 2$. In this case, the only possible configuration is $(1, 1, 1)$, or equivalently that there are three edges between some pair of vertices u and v. Notice that if we were conducting BFS from u we would find that $up\ (v) \geqq 3$.

Case 3. $|C_1| = 3$. Here there are two possible configurations, $(1, 2, 2)$ and $(2, 2, 2)$. These are depicted in Fig. 3.

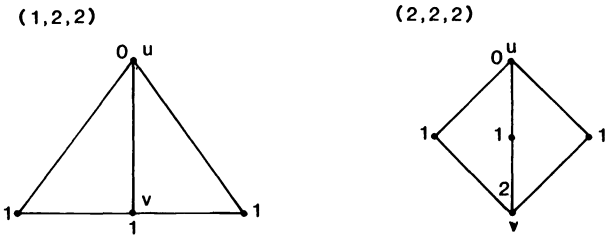


FIG. 3

(Notice that $(1, 1, 1)$ and $(1, 1, 2)$ cannot occur because then the shortest cycle would have at most 2 edges.) The labels assigned to the nodes in these figures, and the ones that follow, are the BFS labels of the nodes on the assumption that the given configuration is an optimal 3-face subgraph. For the first situation, in performing BFS from u, we find a vertex labeled 1 with $cross\ (v) \geqq 2$. In the second we find a vertex labeled 2 with $up\ (v) \geqq 3$. Notice that in both of these cases we have found vertices with $cross\ (v) + up\ (v) \geqq 3$.

Case 4. $|C_1| = 4$. In this case, there are the configurations $(2, 2, 2)$, $(2, 2, 3)$, $(1, 3, 3)$, $(2, 3, 3)$ (see Fig. 4).

(Note that $(3, 3, 3)$ need not be considered here, since *greedy* (G^*) is sure to find C_2 with at most 5 edges.) As above, $(2, 2, 2)$ can be detected by finding a vertex v

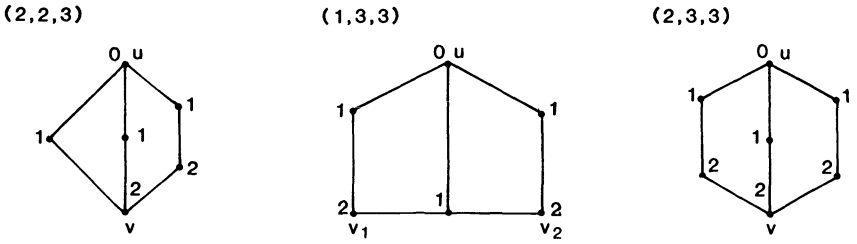


FIG. 4

labeled with a 2 with *up* $(v) \geqq 3$. For the case (2, 2, 3), we find a vertex v with *up* $(v) = 2$ and *cross* $(v) \geqq 1$. For (1, 3, 3), we find a vertex labeled 1 with two neighbors v_1 and v_2 with *up* $(v_i) = 2$. For (2, 3, 3), we find a vertex v with *cross* $(v) = 2$ (and *up* $(v) \geqq 1$).

Case 5. $|C_1| = 5$. For this final case, only the configurations (2, 3, 3), (3, 3, 3), (2, 3, 4) and (1, 4, 4) must be considered. The first configuration is handled exactly as in Case 4, and the remaining ones are depicted in Fig. 5.

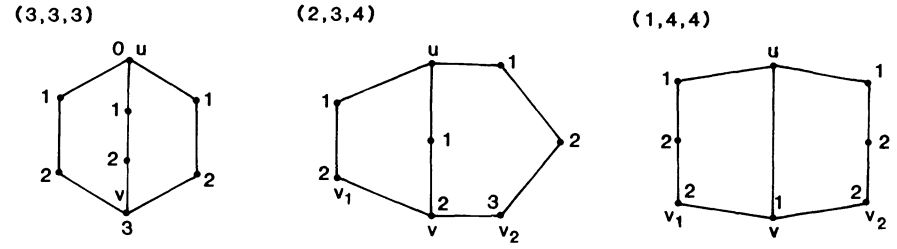


FIG. 5

For (3, 3, 3) we find a vertex labeled 3 with *up* $(v) = 3$. For (2, 3, 4), a vertex v labeled 2 with *cross* $(v) = 1$ has a neighbor v_2 with *up* $(v_2) \geqq 2$. Finally, for (1, 4, 4) a vertex v labeled with a 1 has two neighbors, v_1 and v_2 with *cross* $(v_i) \geqq 1$.

Summarizing what we have done, there are only 2 different types of configurations that we must search for in performing a BFS from u:

1. A vertex v with *cross* $(v) +$ *up* $(v) \geqq 3$.
2. A vertex with two neighbors v_1 and v_2 with *cross* $(v_i) +$ *up* $(v_i) \geqq 2$.

Therefore, *if greedy* (G^*) does not find an optimal solution, then by performing BFS from each vertex and searching for one of these two configurations, we are guaranteed to find an optimal 3-face subgraph. It is quite straightforward to modify BFS to detect these configurations. Therefore, we see that it can be implemented in $O(|V|^2)$ time, since BFS from each node requires $O(|V|)$ time for planar graphs. Furthermore, by performing *greedy* (G^*) and this procedure as well, and then choosing the best solution found, we find an optimal 3-face subgraph in $O(|V^2|)$ time.

It is significant to note that the planarity of G is used in two ways. Most importantly, it insures the existence of the dual graph; it is used secondarily in guaranteeing an "easy" cut of size 10, and thus limiting the search considerably.

Another interesting point is that the same argument as was used above, can be used to show that a greedy approach never delivers a k-cut of size more than $5k$. Thus, for fixed k the minimum k-cut problem is easily seen to be polynomial using an enumerative approach. Can a simpler approach, analogous to the one given here be used instead? Such an approach is complicated by the fact that the types of possible

configurations become more complicated (and much more numerous) than what was given in Fig. 2 for the 3-cut problem.

The complexity of the 3-cut problem for arbitrary graphs remains a challenging open problem. Recently, Hochbaum and Tsai [HT] have shown that a greedy heuristic for the 3-cut problem gives a solution, even for the weighted case, that is at most 4/3 the optimal solution, and that this bound is tight. Independently, Johnson, Papadimitriou, Seymour and Yannakakis [JPSY] proved this result in more general form, that for the weighted k-cut problem a greedy approach yields a solution no more than $(2-2/k)$ times the optimal solution. They also consider a more general problem: the problem of finding a minimum k-cut with the additional constraint that they specify k vertices that must be in different components. They showed that for planar weighted graphs the minimum 3-cut problem with specified vertices is polynomial. Their algorithm, although polynomial, is not efficient, since the polynomial is a polynomial in $k!$ and the degree of the polynomial depends linearly on k. Furthermore, they showed that the 3-cut problem with specified vertices for arbitrary graphs is NP-complete. This does not appear to imply anything about the complexity of the 3-cut problem for arbitrary graphs. If the general 3-cut problem were to be polynomial, it would be a remarkable extension of one of the oldest results in combinatorial optimization.

Acknowledgment. We are grateful to the anonymous referee whose suggestions greatly enhanced the clarity and simplicity of this paper.

REFERENCES

[AHU] A. V. AHO, J. E. HOPCROFT AND J. D. ULLMAN, *The Design and Analysis of Computer Algorithms*, Addison-Wesley, Reading, MA, 1974.

[JPSY] D. S. JOHNSON, C. H. PAPADIMITRIOU, P. D. SEYMOUR AND M. YANNAKAKIS, Private communication, 1983.

[HT] D. S. HOCHBAUM AND L.-H. TSAI, *A greedy heuristic for the minimum 3-cut problem*, manuscript, 1983.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 4, October 1985

017

EXAMPLES OF JUMP-CRITICAL ORDERED SETS*

M. H. EL-ZAHAR† AND I. RIVAL†

Abstract. For an ordered set P and for a linear extension L of P let $s(P, L)$ stand for the number of ordered pairs (x, y) of elements of P such that x is an immediate successor of y in L but x is not even above y in P. Put $s(P) = \min\{s(P, L) | L \text{ linear extension of } P\}$, the *jump number* of P. Call an ordered set P *jump-critical* if $s(P-\{x\}) < s(P)$ for any $x \in P$. *There are precisely seventeen jump-critical ordered sets with jump number at most three.*

Key words. ordered set, linear extension, jump number, jump-critical

AMS(MOS) subject classifications. primary 06A10, 06A05; secondary 90B35

The purpose of this note is to stimulate activity on the *jump number* of an ordered set by recording several important examples. Every linear extension L of a finite ordered set P can be expressed as the linear sum $C_1 \oplus C_2 \oplus \cdots \oplus C_m$ of chains C_i of P so labelled that $\sup_P C_i \not\leqq \inf_P C_{i+1}$. (The *linear sum* $A \oplus B$ of ordered sets A and B is the set $A \cup B$ ordered so that $a \leqq b$ provided that $a \in A$ and $b \in B$, or else, $a \leqq b$ in A or, $a \leqq b$ in B.)

Let $C_i = \{a_i = a_{i1} < a_{i2} < \cdots < a_{ik_i} = b_i\}$. Then $b_i \not\geqq a_{i+1}$ and such a pair (b_i, a_{i+1}) is called a *jump* (or *setup*) of the linear extension L, which is said to have $m-1$ jumps. We write $s(P, L) = m-1$. Note that a_{i+1} covers b_i in L, although $a_{i+1} \not\geqq b_i$ in P itself. We put $s(P) = \min\{s(P, L) | L \text{ linear extension of } P\}$, the *jump number* of P.

In the language of scheduling this may be rendered as follows. Suppose we are to schedule a set of tasks for processing, one at a time, by a single machine. Precedence constraints, due perhaps to technological dictates, prohibit the start of certain tasks until certain others are already completed. A task which is executed immediately after one which is not constrained to precede it requires a "setup" (jump)—entailing some additional cost. The simplest variation is this: schedule the tasks to minimize the number of jumps.

Observe that $s(P) \geqq s(P-\{x\}) \geqq s(P)-1$ for any $x \in P$. Call an ordered set K *jump-critical* if $s(K-\{x\}) < s(K)$ for each $x \in K$. It is easy to see that every ordered set P contains a jump-critical subset K with $s(P) = s(K)$. It may be that jump-critical ordered sets tell us much about the problem of determining $s(P)$—even about constructing "optimal" linear extensions for P, that is, ones for which $s(P, L) = s(P)$. The ordered set illustrated in Fig. 1 is jump-critical. Obviously, $s(P-\{a_{51}\}) < s(P)$. But to verify that $s(P-\{a_{12}\}) < 4$, for instance, requires a different chain decomposition: $P-\{a_{12}\} = C_2 \oplus C_4 \oplus \{a_{31} < a_{51}\} \oplus \{a_{11} < a_{32} < a_{33}\}$. It is a good exercise to check all ten cases.

An n-element antichain is jump-critical. In fact, it is fairly obvious that the disjoint sum of jump-critical ordered sets is jump-critical. In addition, $s(P_1 + P_2) = s(P_1) + s(P_2) + 1$. It is equally obvious that the linear sum of jump-critical ordered sets is jump-critical. Also, $s(P_1 \oplus P_2) = s(P_1) + s(P_2)$. These are special cases of a more general construction. Let P be an ordered set and for each $x \in P$, let P_x be an ordered set. The *lexicographic sum* $\sum_{x \in P} P_x$ is the set $\bigcup_{x \in P} P_x$ ordered so that $u \leqq v$ if, for some $x \in P$, $u \in P_x$, $v \in P_x$, and $u \leqq v$ in P_x, or else, $u \in P_x$, $v \in P_y$, for some $x < y$ in P. It is

* Received by the editors July 28, 1983, and in revised form March 26, 1984.
† Department of Mathematics and Statistics, University of Calgary, Calgary, Alberta, Canada T2N 1N4.

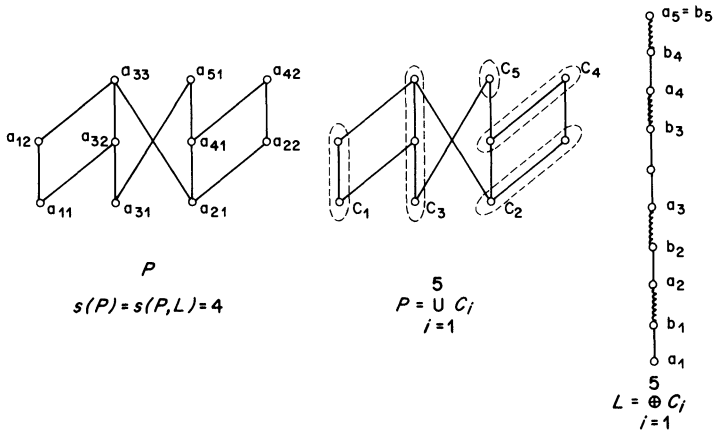


FIG 1

implicit in M. Habib [6] that the lexicographic sum $\sum_{x \in P} P_x$ of critical ordered sets P_x is itself critical, as long as each $|P_x| > 2$.

M. H. El-Zahar and J. H. Schmerl [4] have shown that a jump-critical ordered set P with jump number m has at most $(m+1)!$ elements. Obviously, the only jump-critical ordered set P with $s(P) = 0$ is the singleton. If $s(P) = 1$ then, of course, P must contain a noncomparable pair of elements. So, if P is jump-critical then P must be a two-element antichain. Suppose P is jump-critical and $s(P) = 2$. P may be a three-element antichain. The only other possibility is that P is the "four-cycle." Thus, either $P \cong 1+1+1$ or $P \cong (1+1) \oplus (1+1)$.

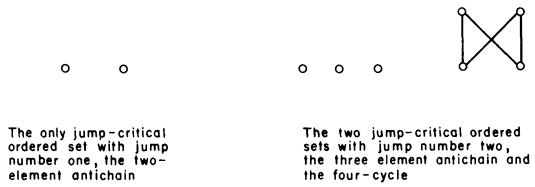


FIG. 2

THEOREM 1. *There are precisely fourteen jump-critical ordered sets with jump number three. These are, up to duality, the ordered sets illustrated in Fig.* 3.

There is of course an obvious intrinsic interest in such a list. However, the particular ordered sets are at times quite interesting, too. For instance, M. Pouzet [8] had conjectured that a jump-critical ordered set P with $s(P) = m$ would satisfy $|P| \leqq 2m$. It is true for any "cycle." M. M. Syslo [9] has shown that this bound holds too for any jump-critical, N-free ordered set. M. Habib [6] disproved this conjecture by exhibiting the seven-element ordered set **G**. A plausible variation on this conjecture was this: *if P is jump-critical and $s(P) = m$ then* $|P| \leqq 2(m+1)$. But that too is false as Habib has already shown. Indeed, let $P = P_1 \oplus P_2 \oplus P_3$, each $P_i \cong \mathbf{G}$. Then $s(P) = 9$ and yet $|P| = 21$. The nine-element ordered set **K** is a simpler counterexample. Even this conjecture is false: *if P is jump-critical and $s(P) = m$ then* $|P| \leqq 3m$. A counterexample can be constructed by using three copies of **K** and "gluing" together successively the two maximal elements with the two minimal elements (see Fig. 4). This ordered set is jump-critical with jump number 7 and yet it has 23 elements. It remains an open question, however, whether the $(m+1)!$ bound of [4] can be (substantially) improved.

In fact, this "gluing" construction can be carried out more generally to manufacture further jump-critical ordered sets. The construction is much like one common in lattice

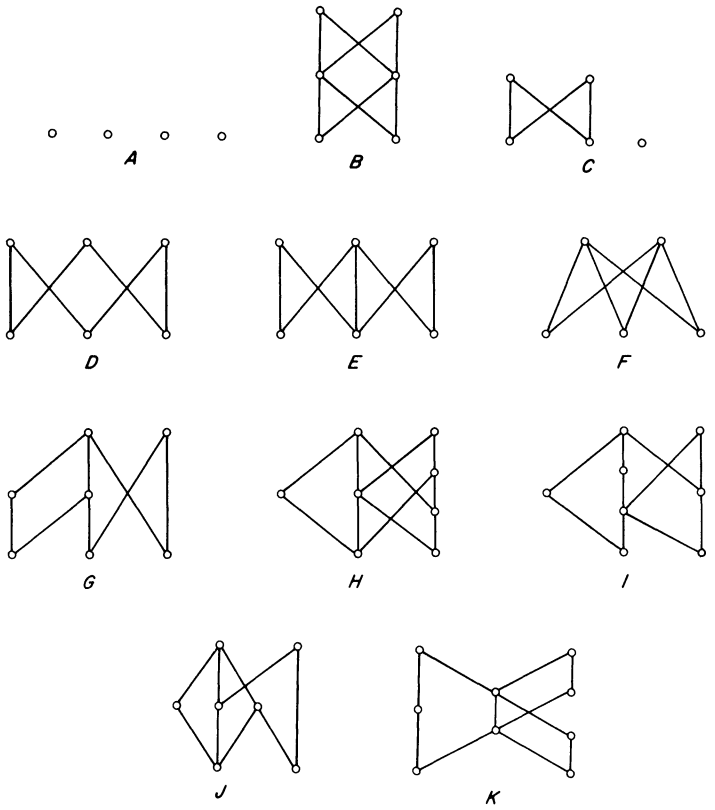


FIG. 3

theory (cf. R. P. Dilworth and M. Hall, Jr. [3], C. Herrmann [7]). Let P_1, P_2 be ordered sets and let S_1 be an up set of P_1 (that is, if $x \in S_1$ and $x \leqq y$ then $y \in S_1$) and let S_2 be a down set of P_2 (that is, if $x \in S_2$ and $y \leqq x$ then $y \in S_2$). Furthermore, suppose that there is an isomorphism f of S_1 onto S_2. Construct an ordered set P on $P_1 \cup (P_2 - S_2)$ by $x \leqq y$ in P if (i) $x \in P_1$, $y \in P_1$, and $x \leqq y$ in P_1; or (ii) $x \in P_1$, $y \in P_2$, $x \leqq u$ for some $u \in S_1$, and $f(u) \leqq y$ in P_2; or (iii) $x \in P_2$, $y \in P_2$, and $x \leqq y$ in P_2. We say, in any case, that P is obtained from P_1 and P_2 by *gluing* $S_1 \subseteq P_1$ with $S_2 \subseteq P_2$. This construction is particularly interesting to us in the case that $S_1 \subseteq \max P_1$, the maximal elements of P_1, and $S_2 \subseteq \min P_2$, the minimal elements of P_2. The ordered set illustrated in Fig. 4 is obtained by gluing max **K** with min **K**, "two times in succession."

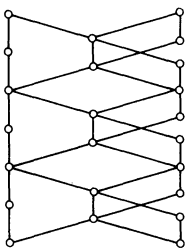


FIG. 4

THEOREM 2. *Let P_1 and P_2 be finite jump-critical ordered sets. Any ordered set obtained from P_1 and P_2 by gluing a maximal element of P_1 with a minimal element of P_2 is jump-critical and, in this case, the jump number is $s(P_1)+s(P_2)$. If $|\max P_1| = |\min P_2| = 2$ then the ordered set obtained from P_1 and P_2 by gluing* $\max P_1$ *with* $\min P_2$ *is jump-critical and, in this case, the jump number is $s(P_1)+s(P_2)-1$.*

This gluing construction can be used to construct an example of a jump-critical ordered set in which an "optimal" linear extension uses a "long" chain (see Fig. 5).

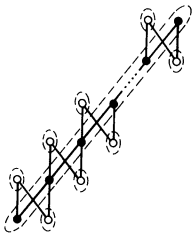


FIG. 5

There is an obvious question that arises from the second part of Theorem 2: does the gluing the construction produce a jump-critical ordered set if there are more than two maximal elements? In Fig. 6 we have illustrated the gluing of two copies of **D** to produce an ordered set P with jump number four. However, P is not jump-critical: it contains the jump-critical ordered set $\max P \cup \min P \cong (1+1+1)\oplus(1+1+1)$ which also has jump number four.

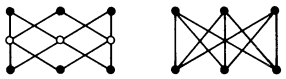


FIG. 6

There is also a natural extension of the gluing construction using the up set of one ordered set with the up set of another.

This is illustrated in Fig. 7. The obvious analogue of Theorem 2 still holds in this case.

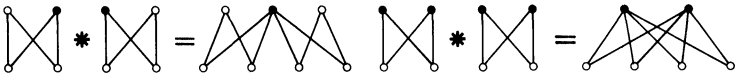


FIG. 7

Before we proceed to the proof of Theorem 1 we present some terminology that is of use to us.

Let P be a finite ordered set. For an element a in P put $D(a)=\{x\in P|x\leqq a\}$, a down set in P, and $U(a)=\{x\in P|x\geqq a\}$, an up set in P. Following M. H. El-Zahar and J. H. Schmerl [4] call the element *a accessible* in P if $D(a)$ is a chain in P. For instance, each minimal element of P is accessible. Let $w(P)$ stand for the *width* of P, the size of a maximum-sized antichain. According to Dilworth's chain decomposition theorem [1], P is the (disjoint) union of $w(P)$ chains. For maximum-sized antichains A, B in P we write $A\leqq B$ whenever for each $a\in A$ there is $b\in B$ satisfying $a\leqq b$. (It follows, in this case that, for each $b\in B$ there is $a\in A$ satisfying $a\leqq b$, too.) In this way the set of maximum-sized antichains of P is ordered: there is a greatest (highest)

antichain and a least (lowest) antichain. As a matter of fact, the set of maximum-sized antichains is a distributive lattice in which $A \vee B = \max(A \cup B)$ and $A \wedge B = \min(A \cup B)$ (R. P. Dilworth [2], cf. R. Freese [5]). Finally, note that a jump-critical ordered set P contains at least two maximal elements and at least two minimal elements. (For, if P has a unique maximal element, say x, then $s(P-\{x\}) = s(P)$.)

Proof of Theorem 1. It is straightforward, if somewhat laborious, to verify that each of the ordered sets illustrated in Fig. 3 has jump number three and that each is jump-critical.

Let P be a jump-critical ordered set with jump number three. For contradictions, suppose that P contains no subset isomorphic, or dually isomorphic, to any of the ordered sets illustrated in Fig. 3.

If P contains a four-element antichain then, of course, that must be all of P, that is, $P \cong \mathbf{A}$. Therefore, $w(P) = 2$ or $w(P) = 3$.

Let $w(P) = 2$. Then P is the union of two maximal chains C_1 and C_2. Put $a_i = \inf_P C_i$ and $b_i = \sup_P C_i$, for $i = 1, 2$. As P is jump-critical, $b_1 \neq b_2$ and $a_1 \neq a_2$ (in fact $C_1 \cap C_2 = \varnothing$). Of course, b_1 cannot be accessible, for otherwise $C_1 \oplus C_2$ would be a linear extension of P (in spite of $s(P) = 3$). Moreover, $s(D(b_1)) = 2$, for if $s(D(b_1)) = 1$ then $s(P) = 2$. As $D(b_1)$ has width two it must contain a four-cycle, say with minimal elements c_1, d_1 and maximal elements e_1, f_1; similarly, $D(b_2)$ will contain a four-cycle with minimal elements c_2, d_2 and maximal elements e_2, f_2. Then

$$(\{c_1, d_1\} \wedge \{c_2, d_2\}) \cup (\{e_1, f_1\} \wedge \{e_2, f_2\}) \cup \{b_1, b_2\} \cong \mathbf{B}.$$

This leaves the case that P is the disjoint union of three chains C_1, C_2, C_3. Again, put $a_i = \inf_P C_i$ and $b_i = \sup_P C_i$, for $i = 1, 2, 3$.

Let us suppose that both $\{a_1, a_2, a_3\}$ and $\{b_1, b_2, b_3\}$ are antichains. If b_1, say, is accessible then $s(P - D(b_1)) = 2$ and, as the width of $P - D(b_1)$ is two, it must contain a four-cycle. This four-cycle together with b_1 constitutes a subset of P isomorphic to $\mathbf{C}$. We may then suppose that none of the b_i's is accessible. Then each $a_i \neq b_i$, $|D(b_i) \cap \{a_1, a_2, a_3\}| \geq 2$ and, dually, $|U(a_i) \cap \{b_1, b_2, b_3\}| \geq 2$. It follows that $\{a_1, a_2, a_3, b_1, b_2, b_3\}$ is isomorphic to $\mathbf{D}$ or $\mathbf{E}$, or that $\{a_1, a_2, a_3, b_1, b_2, b_3\}$ contains $\mathbf{F}$ or $\mathbf{F}^d$ (up to isomorphism).

Next we handle the case that either $\{a_1, a_2, a_3\}$ or $\{b_1, b_2, b_3\}$ is not an antichain, say $\{b_1, b_2, b_3\}$ is not an antichain. Let $\{c_1, c_2, c_3\}$ be the supremum of all three-element antichains in P. One of the c_i's must be accessible for otherwise the proper subset $\bigcup_{i=1}^{3} D(c_i)$ of P has jump number three. Let c_1 be accessible. If $P - D(c_1)$ contains a three-element antichain $\{x_1, x_2, x_3\}$ then c_1 must be comparable to one of these x_i's, say x_1. But $x_1 \not< c_1$ since $x_1 \notin D(c_1)$ and if $c_1 < x_1$ then $\{c_1, c_2, c_3\}$ is not the highest three-element antichain in P. Therefore, $w(P - D(c_1)) = 2$ and we can assume that $P - D(c_1) = C_2 \cup C_3$ so $D(c_1) = C_1$. Let $\{d_2, d_3\}$, $\{e_2, e_3\}$ be, respectively, the lowest, highest, two-element antichains in $C_2 \cup C_3$ where, say, $d_i, e_i \in C_i$ for both $i = 2, 3$. Since $s(C_2 \cup C_3) = 2$ then $\{d_2, d_3, e_2, e_3\}$ is a four-cycle.

Neither d_i is below c_1. Also, c_1 cannot be below either d_i. To see this let $c_1 < d_2$. According to the maximality of $\{c_1, c_2, c_3\}$, either $c_2 < d_2$ or $c_3 < d_2$. Therefore $\{d_2, d_3\} \not< \{c_2, c_3\}$ which is a contradiction. Moreover, either $c_1 < e_2$ or $c_1 < e_3$, for otherwise $\{c_1, d_2, d_3, e_2, e_3\} \cong \mathbf{C}$. Therefore, $\max P = \max(C_2 \cup C_3) = \{e_2, e_3\}$ for otherwise P would have a unique maximal element.

If $c_1 < e_2$ and $c_1 < e_3$ then $\{c_1, d_2, d_3, e_2, e_3\} \cong \mathbf{F}$.

We may then suppose that $c_1 < e_2$ and therefore, that $c_1 \not< e_3$. Put $c_0 = \inf_P C_1$. Let us suppose that $c_0 \not< d_2$ and $c_0 \not< d_3$. Then there must be an element $e \in C_2 \cup C_3$, such that $e \neq e_2$, $e > c_0$ and e noncomparable with c_1. Otherwise, c_0 is accessible in the dual

P^d of P and, as $P - U(c_0)$ has width two and jump number two, it must contain a four-cycle which, with c_0 is a subset of P isomorphic to **C**. If $c_0 \leqq e_3$ then $\{c_0, d_2, d_3, e_2, e_3\} \cong \mathbf{F}$. Otherwise, $d_2 < e < e_2$ and e, e_3 are noncomparable. Then $\{c_0, c_1, d_2, d_3, e_3, e\} \cong \mathbf{G}$. Therefore, $c_0 < d_2$ or $c_0 < d_3$.

By the minimality of $\{d_2, d_3\}$ either $\{d_2, d_3\} = \min(C_2 \cup C_3)$ or else there is a unique element $d_0 = \min(C_2 \cup C_3)$. Suppose that $c_0 < d_2$ and $c_0 < d_3$. Since P is jump-critical it cannot have a least element so there is a unique minimal element d_0 in $C_2 \cup C_3$ and $\min P = \{c_0, d_0\}$. It follows that $\{c_0, d_0, d_2, d_3, e_2, e_3\} \cong \mathbf{B}$. Therefore, c_0 is below exactly one of d_2, d_3.

At this stage of the proof it is helpful to visualize schematically the structure of P. In fact, P now resembles one of the four ordered sets illustrated in Fig. 8. We shall below refer to the corresponding cases (a), (b), (c), (d) as illustrated in Fig. 8.

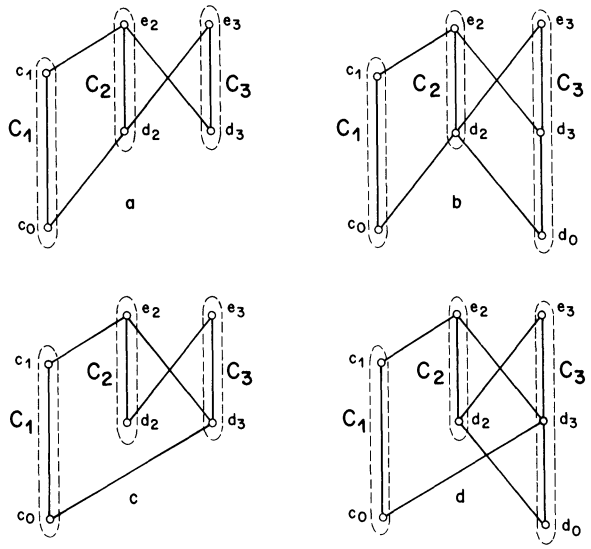


FIG. 8

If $s(D(e_3)) = 1$ and $D(e_2) - D(e_3)$ is a chain then $s(P) = 2$. Therefore, either $s(D(e_3)) = 2$ or else $D(e_2) - D(e_3)$ is not a chain.

Case (i). Let $s(D(e_3)) = 2$. In this case either $D(e_3)$ contains a three-element antichain or it contains a four-cycle. Suppose $D(e_3)$ contains a three-element antichain $\{x_1, x_2, x_3\}$. Then

$$(\{x_1, x_2, x_3\} \wedge \{c_1, d_2, d_3\}) \cup \{e_2, e_3\} \cong \mathbf{F}.$$

(Note that $\{c_1, d_2, d_3\}$ is an antichain.)

Let us suppose then that $D(e_3)$ contains a four-cycle with minimal elements x_0, x_1 and maximal elements x_2, x_3. None of the x_i's is above c_1, for otherwise $e_3 > c_1$. If each x_i is noncomparable with c_1 then $\{c_1, x_0, x_1, x_2, x_3\} \cong \mathbf{C}$. Since c_1 is accessible in P, $c_1 \not> x_2$ and $c_1 \not> x_3$. Therefore, c_1 is above exactly one of x_0, x_1, say $c_1 > x_1$, that is, $x_1 \in C_1$.

We may suppose that $x_2 \in C_2$ and $x_3 \in C_3$. If $e_2 > x_3$ then $\{x_0, x_1, x_2, x_3, e_2, e_3\} \cong \mathbf{B}$. Therefore, $d_3 < x_3 < e_3$ and $d_2 \leqq x_2 < e_2$.

(a) If $x_3 > d_2$ then $\{c_1, d_2, d_3, x_2, x_3\} \cong \mathbf{C}$. Suppose then that $x_3 \not> d_2$. Notice that, as $\min(C_2 \cup C_3) = \{d_2, d_3\}$, either $x_0 \geqq d_2$ or $x_0 \geqq d_3$. But $x_0 \geqq d_2$ implies that $x_3 > d_2$; hence, $x_0 \geqq d_3$. In this case, $x_2 > d_3$. It follows now that $\{c_0, c_1, d_2, d_3, x_2, x_3, e_2, e_3\} \cong \mathbf{I}^d$.

(b) If $x_2 = d_2$ then $\{c_0, c_1, d_2, e_2, d_0, d_3, x_3, e_3\} \cong \mathbf{H}$. If $x_3 > d_2$ and $x_2 \not> d_3$ then $\{c_0, c_1, d_2, x_2, e_2, d_0, d_3, x_3\} \cong \mathbf{I}$. If $x_3 > d_2$ and $x_2 > d_3$ then $\{c_1, d_2, d_3, x_2, x_3\} \cong \mathbf{C}$. If $x_3 \not> d_2$ and $x_2 > d_3$ then $\{c_0, c_1, d_2, x_2, e_2, d_3, x_3, e_3\} \cong \mathbf{I}^d$. If $x_3 \not> d_2$ and $x_2 \not> d_3$ then $\{c_0, c_1, d_2, e_2, d_0, d_3, x_3, e_3\} \cong \mathbf{H}$.

(c) Note that $x_2 \neq d_2$ for otherwise $x_2 > x_1 \geqq c_0$ implies $c_0 \leqq d_2$, which does not hold in this case. If $x_2 > d_3$ then $\{c_1, d_2, x_2, d_3, x_3\} \cong \mathbf{C}$. If $x_2 \not> d_3$ then $\{c_0, c_1, d_2, x_2, e_2, d_3, x_3\} \cong \mathbf{J}$. (Observe that either $x_0 \geqq d_2$ or $x_0 \geqq d_3$ and, if $x_2 \not> d_3$ then $x_0 \geqq d_2$ so $x_3 > d_2$.)

(d) If $x_2 \not> d_3$ then $\{c_0, d_0, x_2, d_3, e_2, e_3\} \cong \mathbf{B}$. If $x_2 > d_3$ and $x_3 > d_2$ then $\{c_1, d_2, x_2, d_3, x_3\} \cong \mathbf{C}$. If $x_2 > d_3$ and $x_3 \not> d_2$ then $\{c_0, c_1, d_2, x_2, e_2, d_0, d_3, x_3, e_3\} \cong \mathbf{K}$.

Case (ii). Let $s(D(e_3)) = 1$ and suppose that $D(e_2) - D(e_3)$ is not a chain.

Let $\{f_2, f_3\}$ be the highest two-element antichain in $D(e_3)$. None of the f_i's belong to C_1, for otherwise, we would not have $\{d_2, d_3\} \leqq \{f_2, f_3\}$. Let $f_i \in C_i$ for $i = 2, 3$. The set $\{x \in C_2 | x \not< e_3 \text{ and } x \not> c_1\}$ is nonempty since $D(e_2) - D(e_3)$ is not a chain. Let $g = \max\{x \in C_2 | x \not< e_3 \text{ and } x \not> c_1\}$. Observe that $D(e_3) - D(g)$ is a chain since $D(f_2) \subseteq D(g)$. The element g cannot be accessible, for otherwise

$$D(g) \oplus (D(e_3) - D(g)) \oplus (D(e_2) - (D(e_3) \cup D(g)))$$

is a linear extension of P with only two jumps. Finally, if $g > d_3$ then $\{c_1, d_2, g, d_3, e_3\} \cong \mathbf{C}$. Therefore, we assume that $g \not> d_3$.

(a) Since g is not accessible, there must exist an element $h \in C_1$ satisfying $h < g$ and $h \not< d_2$. Then $\{h, d_2, d_3\}$ is an antichain. Therefore, $\{h, c_1, d_2, g, e_2, d_3, e_3\} \cong \mathbf{G}$.

(b) In this case $\{c_0, c_1, d_2, g, e_2, d_0, d_3, e_3\} \cong \mathbf{I}$.

(c), (d) We must have $g > c_0$ and, therefore, $\{c_0, c_1, d_2, g, e_2, d_3, e_3\} \cong \mathbf{J}$.

Proof of Theorem 2. Let P_1, P_2 be jump-critical ordered sets, let $a_1 \in \max P_1$ and $a_2 \in \min P_2$. Let P be the ordered set obtained by gluing a_1 with a_2. First, we see that $s(P) = s(P_1) + s(P_2)$. In any linear extension of P the elements of P_1 appear in at least $s(P_1)+1$ chains and the elements of P_2 in at least $s(P_2)+1$ chains. And, at most one chain of P contains elements from both P_1 and P_2—the chain containing $a_1 (= a_2)$. Hence $s(P) \geqq s(P_1) + s(P_2)$. On the other hand, we can construct an "optimal" linear extension of P by taking first $s(P_1)+1$ chains in a linear extension of P_1 and then $s(P_2)$ chains in a linear extension of $P_2 - \{a_2\}$. (Note that $s(P_2 - \{a_2\}) = s(P_2) - 1$ since P_2 is jump-critical.) It follows that $s(P) \leqq s(P_1) + s(P_2)$. Now to see that P is jump-critical let $x \in P$, say $x \in P_1$. As P_1 is jump-critical there is a linear extension of $P_1 - \{x\}$ using $s(P_1)$ chains and this can be followed by $s(P_2)$ chains in a linear extension of $P_2 - \{a_2\}$ to produce a linear extension of P with jump number $s(P) - 1$. The case that $x \in P_2$ is similar.

Let P_1, P_2 be jump-critical ordered sets, let $\max P_1 = \{a_1, b_1\}$, $\min P_2 = \{a_2, b_2\}$, and let P be the ordered set obtained by gluing $\max P_1$ with $\min P_2$, say a_1 to a_2 and b_1 to b_2. As above we first verify that $s(P) = s(P_1) + s(P_2) - 1$. In any linear extension of P the elements of P_1 appear in at least $s(P_1)+1$ chains and the elements of P_2 in at least $s(P_2)+1$ chains. At most two chains of P contain elements from both P_1 and P_2, namely, the chains containing a_1 $(= a_2)$ and b_1 $(= b_2)$. Therefore, $s(P) \geqq s(P_1) + s(P_2) - 1$. We can also construct a linear extension with $s(P_1) + s(P_2) - 1$ jumps. To this end let L_1 be a linear extension of P_1 with $s(P_1, L_1) = s(P_1)$. We may suppose that $a_1 < b_1$ in L_1. In this case b_1 is the top element of L_1. Now, we can construct a linear extension L_2 of P_2 in which the smallest element is a_2 (just follow the chain $\{a_2\}$ by the $s(P_2)$ chains in an "optimal" linear extension of $P_2 - \{a_2\}$). Let x cover a_2 in L_2. Obviously (a_2, x) is a jump in L_2 and, in fact, $x = b_2$. Then $L_1 \oplus (L_2 - \{a_2, b_2\})$ is a linear extension of P with jump number $s(P_1) + s(P_2) - 1$. Therefore $s(P) \leqq$

$s(P_1)+s(P_2)-1$. Finally, to see that P is jump-critical let $x \in P$, say $x \in P_1$. There is a linear extension of $P_1-\{x\}$ using $s(P_1)$ chains and if $x \in \max P_1$, say $x = a_1$, then this linear extension can be so constructed that b_1 is the top element. Then we can "blend" this linear extension of $P_1-\{x\}$ with one for $P_2-\{a_2\}$, say $L_2-\{a_2\}$ in which b_2 is the bottom element. This produces a linear extension of $P-\{x\}$ with $s(P_1)+s(P_2)-1$ chains. Again the case that $x \in P_2$ is similar.

REFERENCES

[1] R. P. DILWORTH (1950), *A decomposition theorem for partially ordered sets*, Ann. Math., 51, pp. 161-166.
[2] ———, (1960), *Some combinatorial problems on partially ordered sets* in Proc. Symposium in Applied Mathematics, 10, American Mathematical Society, Providence, RI, pp. 85-90.
[3] R. P. DILWORTH AND M. HALL, JR. (1944), *The embedding problem for modular lattices*, Ann. Math., 45, pp. 450-456.
[4] M. H. EL-ZAHAR AND J. H. SCHMERL (1984), *On the size of jump-critical ordered sets*, Order 1, pp. 3-5.
[5] R. FREESE (1974), *An application of Dilworth's lattice of maximal antichains*, Discrete Math., 7, pp. 107-109.
[6] M. HABIB (1983), *Comparability invariants*, preprint.
[7] C. HERRMANN (1973), *S-verklebte Summen von Verbänden*, Math. Z., 130, pp. 225-274.
[8] M. POUZET (1982), *Problem* 4.10, in Ordered Sets, I. Rival, ed., D. Reidel, Dordrecht, Holland.
[9] M. M. SYSLO (1984), *A graph-theoretic approach to the jump-number problem* in Graphs and Order, I. Rival, ed., D. Reidel, Dordrecht, Holland.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 4, October 1985

018

TOTALLY-BALANCED AND GREEDY MATRICES*

A. J. HOFFMAN†, A. W. J. KOLEN‡ AND M. SAKAROVITCH§

Abstract. Totally-balanced and greedy matrices are (0, 1)-matrices defined by excluding certain submatrices. For a $n \times m$ (0, 1)-matrix A we show that the linear programming problem $\max\{by \mid yA \leqq c, 0 \leqq y \leqq d\}$ can be solved by a greedy algorithm for all $c \geqq 0$, $d \geqq 0$ and $b_1 \geqq b_2 \geqq \cdots \geqq b_n \geqq 0$ if and only if A is a greedy matrix. Furthermore we show constructively that if b is an integer, then the corresponding primal problem $\min\{cx + dz \mid Ax + z \geqq b, x \geqq 0, z \geqq 0\}$ has an integer optimal solution. A polynomial-time algorithm is presented to transform a totally-balanced matrix into a greedy matrix as well as to recognize a totally-balanced matrix. This transformation algorithm together with the result on greedy matrices enables us to solve a class of integer programming problems defined on totally-balanced matrices. Two examples arising in tree location theory are presented.

AMS(MOS) subject classifications. 05C50, 90C05, 90C10

1. Introduction. A (0, 1)-matrix is *balanced* if it does not contain an odd square submatrix with all row and column sums equal to two. Balanced matrices have been studied extensively by Berge [3] and Fulkerson et al. [7]. We consider a more restrictive class of matrices called totally-balanced (Lovász [11]). A (0, 1)-matrix is *totally-balanced* if it does not contain a square submatrix which has no identical columns and its row and column sums equal to two.

Example 1.1. Let $T = (V, E)$ be a tree with vertex set $V = \{v_1, v_2, \cdots, v_n\}$ and edge set E. Each edge $e \in E$ has a positive length $l(e)$. A *point* on the tree can be a vertex or a point anywhere along the edge. The *distance* $d(x, y)$ between the two points x and y on T is defined as the length of the path between x and y. A *neighborhood subtree* is defined as the set of all points on the tree within a given distance (called *radius*) of a given point (called *center*). Let x_i $(i = 1, 2, \cdots, m)$ be points on T and let r_i $(i = 1, 2, \cdots, m)$ be nonnegative numbers. Define the neighborhood subtrees T_i by $T_i = \{y \in T \mid d(y, x_i) \leqq r_i\}$. Let $A = (a_{ij})$ be the $n \times m$ (0, 1)-matrix defined by $a_{ij} = 1$ if and only if $v_i \in T_j$. It was first proved by Giles [8] that A is totally-balanced. This result was generalized by Tamir [13]: Let Q_i $(i = 1, 2, \cdots, n)$ and R_j $(j = 1, 2, \cdots, m)$ be neighborhood subtrees and let the $n \times m$ (0, 1)-matrix $B = (b_{ij})$ be defined by $b_{ij} = 1$ if and only if $Q_i \cap R_j \neq \emptyset$. Then B is totally-balanced.

Motivation for the types of problems to be studied in this paper is given by the following two examples from tree location theory stated in Example 1.2.

Example 1.2. Let $T = (V, E)$ be a tree, let T_j $(j = 1, 2, \cdots, m)$ be neighborhood subtrees and let $A = (a_{ij})$ be the (0, 1)-matrix as defined in Example 1.1. We interpret x_j as the possible location of a facility, and T_j as the service area of a facility at x_j, i.e., x_j can only serve clients located at T_j (we assume clients to be located at vertices). We assume there is a cost c_j associated with establishing a facility at x_j $(j = 1, 2, \cdots, m)$. The *minimum cost covering problem* is to serve all clients at minimum cost. This problem can be formulated as

$$\begin{aligned} &\min \sum_{j=1}^{m} c_j x_j \\ &\text{s.t.} \sum_{j=1}^{m} a_{ij} x_j \geqq 1, \qquad i = 1, 2, \cdots, n, \\ &\qquad x_j \in \{0, 1\}, \qquad j = 1, 2, \cdots, m. \end{aligned} \tag{1.3}$$

* Received by the editors September 18, 1981, and in final revised form July 23, 1984.
† IBM T. J. Watson Research Center, Yorktown Heights, New York 10598.
‡ Econometric Institute, Erasmus University, Rotterdam, the Netherlands.
§ IMAG, Grenoble, France.

Let us relax the condition in this problem that each client has to be served by assuming that if a client located at vertex v_i is not served by a facility, then a penalty cost of d_i $(i = 1, 2, \cdots, n)$ is charged. The *minimum cost operating problem* is to minimize the total cost of establishing facilities and not serving clients, i.e.,

$$\begin{aligned} &\min \sum_{j=1}^{m} c_j x_j + \sum_{i=1}^{n} d_i z_i \\ &\text{s.t.} \sum_{j=1}^{m} a_{ij} x_j + z_i \geqq 1, \qquad i = 1, 2, \cdots, n, \\ &x_j \in \{0, 1\}, \qquad j = 1, 2, \cdots, m, \\ &z_i \in \{0, 1\}, \qquad i = 1, 2, \cdots, n. \end{aligned} \tag{1.4}$$

Let $A = (a_{ij})$ be a $(0, 1)$-matrix. We can associate a subset of rows to each column, namely those rows which have a one in this column. An $n \times m$ $(0, 1)$-matrix is called *greedy* if for all $i = 1, 2, \cdots, n$ the following holds; all columns having a one in row i can be totally ordered by inclusion when restricted to the rows $i, i+1, \cdots, n$. An equivalent definition is to say that the two 3×2 submatrices

$$\begin{bmatrix} 1 & 1 \\ 0 & 1 \\ 1 & 0 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} 1 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{1.5}$$

do not occur. Why the name "greedy" is chosen will become clear in the next section. It is a trivial observation that each greedy matrix is totally-balanced. We will prove in § 3 that, conversely, the rows of a totally-balanced matrix can be permuted in such a way that the resulting matrix is greedy. The proof will be constructive.

Let the $n \times m$ $(0, 1)$-matrix $A = (a_{ij})$ be greedy. Consider the problem (P) given by

$$\begin{aligned} &\min \sum_{j=1}^{m} c_j x_j + \sum_{i=1}^{n} d_i z_i \\ &\text{s.t.} \sum_{j=1}^{m} a_{ij} x_j + z_i \geqq b_i, \qquad i = 1, 2, \cdots, n, \\ &x_j \geqq 0, \qquad j = 1, 2, \cdots, m, \\ &z_i \geqq 0, \qquad i = 1, 2, \cdots, n. \end{aligned} \tag{P}$$

The dual problem D is given by

$$\begin{aligned} &\max \sum_{i=1}^{n} b_i y_i \\ &\text{s.t.} \sum_{i=1}^{n} y_i a_{ij} \leqq c_j, \qquad j = 1, 2, \cdots, m, \\ &0 \leqq y_i \leqq d_i, \qquad i = 1, 2, \cdots, n. \end{aligned} \tag{D}$$

We will show in § 2 that problem (D) can be solved by a greedy algorithm for all $c \geqq 0$, $d \geqq 0$ and $b_1 \geqq b_2 \geqq \cdots \geqq b_n \geqq 0$ if and only if the matrix A is greedy. Further we construct an optimal solution to the primal problem (P) which has the property that it is an integer solution whenever b is integer. This means that after we use the algorithm of § 3 to transform a totally-balanced matrix into a greedy matrix we can solve the two location problems using the result of § 2.

After we submitted the first version of the paper we found out about the work done by Farber. Farber [5], [6] studies strongly chordal graphs and gives polynomial-time algorithms to find a minimal weighted dominating set and minimal weighted independent dominating set. In these algorithms Farber uses the same approach as described in § 2. In another paper Anstee and Farber [1] relate strongly chordal graphs to totally-balanced matrices. This paper contains the relationship between totally-balanced and greedy matrices described in § 3 as well as a recognition algorithm for a totally-balanced matrix which, however, is less efficient than the one described here.

2. The algorithm. A greedy (0, 1)-matrix is in *standard greedy form* if it does not contain $\left[\begin{smallmatrix}1 & 1\\ 1 & 0\end{smallmatrix}\right]$ as a submatrix. Any $n \times m$ greedy matrix can be transformed into a matrix in standard greedy form by a permutation of the columns in $O(nm)$ time as follows. Consider the columns as (0, 1) vectors and sort them lexicographically reading in reverse, from bottom to top. This gives the desired permutation, for suppose $\left[\begin{smallmatrix}1 & 1\\ 1 & 0\end{smallmatrix}\right]$ occurs as a submatrix with rows i_1, i_2 $(i_1 < i_2)$ and columns j_1, j_2 $(j_1 < j_2)$. Since the columns are ordered lexicographically we know that there exists a row i_3 $(i_3 > i_2)$ such that $a_{i_3 j_1} = 0$ and $a_{i_3 j_2} = 1$, but this contradicts the fact that the matrix is greedy. The algorithm of § 3 applied to a totally-balanced matrix also produces a matrix in standard greedy form. In this section we will assume that the matrix is in standard greedy form. This assumption does not affect the dual solution obtained but facilitates the description of the primal solution.

Let $A = (a_{ij})$ be an $n \times m$ (0, 1)-matrix in standard greedy form, let c_j $(j = 1, 2, \cdots, m)$ and d_i $(i = 1, 2, \cdots, n)$ be positive numbers (the case when one of these numbers is zero can be treated similarly) and let $b_1 \geqq b_2 \geqq \cdots \geqq b_n \geqq 0$. A feasible solution $\bar{y}$ of problem (D) is obtained by a greedy algorithm. The values of $\bar{y}_i$ are determined in order of increasing i and taken to be as large as possible. A constraint j is *tight* if $\sum_{i=1}^{n} \bar{y}_i a_{ij} = c_j$. The index $\alpha(j)$ denotes the largest index of a positive $\bar{y}$-value in the tight constraint j, J denotes a set of tight constraints. The greedy procedure is formulated in Algorithm D.

ALGORITHM D
begin $J := \varnothing$; $\hat{c} := c$;
  for $i := 1$ **step** 1 **to** n
  do $y_i := \min\{d_i, \min_{j: a_{ij}=1}\{\hat{c}_j\}\}$;
    if $\bar{y}_i > 0$ **then if** $\bar{y}_i = \hat{c}_j$ for some j **then** choose the largest j
      and let $J := J \cup \{j\}$; $\alpha(j) := i$
      fi;
      $\hat{c}_j := \hat{c}_j - \bar{y}_i$ for all j such that $a_{ij} = 1$
    fi
  od
end

For the solution $\bar{y}$ constructed by Algorithm D the following hold:

Property 2.1. If $\bar{y}_k = d_k$, then either there is no $j \in J$ such that $a_{kj} = 1$ or there is a $j \in J$ such that $a_{kj} = 1$ and $\alpha(j) \geqq k$; and

Property 2.2. If $\bar{y}_k = 0$, then there is a $j \in J$ such that $a_{kj} = 1$ and $\alpha(j) < k$.

Property 2.1 follows immediately from the algorithm. If $\bar{y}_k = 0$, then there exists an index i, $i < k$ and a constraint $\hat{j}$ such that constraint $\hat{j}$ is tight, $a_{i\hat{j}} = a_{k\hat{j}} = 1$, and i is the largest index of a positive $\bar{y}$-value in constraint $\hat{j}$. During the iteration in which $\bar{y}_i$ was determined we have added an index $j \geqq \hat{j}$ with $\alpha(j) = i$ to J. Since A is a standard greedy form we have $a_{kj} = 1$. This proves Property 2.2.

Example 2.3. The matrix and costs of the example as well as the results of Algorithm D are given in Fig. 2.4. We assume $d_i = 2$ $(i = 1, \cdots, 9)$ and $(b_1, b_2, b_3, b_4, b_5, b_6, b_7, b_8, b_9) = (6, 5, 4, 3, 3, 2, 2, 2, 1)$.

$$\hat{c}_1 = 3, \hat{c}_2 = 4, \hat{c}_3 = 5, \hat{c}_4 = 2, \hat{c}_5 = 3, \hat{c}_6 = 5, \hat{c}_7 = 3.$$

$$\begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}$$

$\bar{y}_1 = 2, \hat{c}_1 = 1, \hat{c}_2 = 2.$
$\bar{y}_2 = 1, J = \{1\}, \alpha(1) = 2, \hat{c}_1 = 0, \hat{c}_2 = 1.$
$\bar{y}_3 = 0.$
$\bar{y}_4 = 2, \hat{c}_3 = 3, \hat{c}_6 = 3.$
$\bar{y}_5 = 2, \hat{c}_3 = 1, \hat{c}_6 = 1.$
$\bar{y}_6 = 2, J = \{1, 4\}, \alpha(4) = 6, \hat{c}_4 = 0, \hat{c}_7 = 1.$
$\bar{y}_7 = 1, J = \{1, 4, 7\}, \alpha(7) = 7, \hat{c}_2 = 0, \hat{c}_5 = 2, \hat{c}_7 = 0.$
$\bar{y}_8 = 0.$
$\bar{y}_9 = 0.$

FIG. 2.4. *Example of Algorithm* D.

The value of the feasible dual solution $\bar{y}$ is 35.

The primal solution $\bar{x}, \bar{z}$ is constructed by Algorithm P which has as input the set of tight constraints J and the indices $\alpha(j)$ $(j \in J)$.

ALGORITHM P
begin $\hat{b} := b$; $\bar{x}_i := 0$ for all $j \notin J$;
  while $J \neq \emptyset$
  do (let k be the last column of J)
    $\bar{x}_k := \hat{b}_{\alpha(k)}$;
    $\hat{b}_i := \hat{b}_i - \bar{x}_k$ for all i such that $a_{ik} = 1$;
    $J := J \backslash \{k\}$
  od;
  for $i := 1$ **step** 1 **to** n **do** $\bar{z}_i := \max(0, \hat{b}_i)$ **od**
end

Example 2.5. Apply Algorithm P to Example 2.3.
$\bar{x}_2 = \bar{x}_3 = \bar{x}_5 = \bar{x}_6 = 0$, $\hat{b} = b$.
Iteration 1: $\bar{x}_7 = 2$, $\hat{b}_6 = \hat{b}_7 = \hat{b}_8 = 0$, $\hat{b}_9 = -1$.
Iteration 2: $\bar{x}_4 = 0$.
Iteration 3: $\bar{x}_1 = 5$, $\hat{b}_1 = 1$, $\hat{b}_2 = 0$, $\hat{b}_3 = -1$.
$\bar{z}_1 = 1$, $\bar{z}_4 = \bar{z}_5 = 3$, all other $\bar{z}_i$ values are zero.
It is easy to check that $\bar{x}, \bar{z}$ is a feasible primal solution with value 35. Since the values of the feasible primal and dual solutions are equal they are both optimal.

If we prove that $\bar{x}_j \geqq 0$ for all $j \in J$, then it is clear that $\bar{y}$ and $\bar{x}, \bar{z}$ are feasible solutions. In order to prove that they are optimal solutions we show that the complementary slackness relations of linear programming hold. These conditions are given by

$$\bar{x}_j \left(\sum_{i=1}^{n} \bar{y}_i a_{ij} - c_j \right) = 0, \qquad j = 1, 2, \cdots, m, \tag{2.6}$$

$$\bar{y}_i \left(\sum_{j=1}^{m} a_{ij} \bar{x}_j + \bar{z}_i - b_i \right) = 0, \qquad i = 1, 2, \cdots, n, \tag{2.7}$$

$$\bar{z}_i (\bar{y}_i - d_i) = 0, \qquad i = 1, 2, \cdots, n. \tag{2.8}$$

Let us denote by $\hat{J}$ the set of column indices in Algorithm P which is initially equal to J and decreases by one element at each iteration. Accordingly let $b_i(\hat{J}) = b_i - \sum_{j \in J \setminus \hat{J}} a_{ij}\bar{x}_j$, $i = 1, 2, \cdots, n$. Define I by $I = \{i \mid \exists j \in J\, \alpha(j) = i\}$.

The following properties hold for Algorithm P.

Property 2.9. If $a_{ij} = a_{lj} = 1$, $i < l$, $j \in \hat{J}$, then $b_i(\hat{J}) \geqq b_l(\hat{J})$.

Proof. This is true at the start of the algorithm since $b_i \geqq b_l$, $i < l$. Let k be the last column of $\hat{J}$. Property 2.9 could be altered only if $a_{ik} = 1$ and $a_{lk} = 0$, which is ruled out by the fact that A is in standard greedy form.

Property 2.10. $b_i(\hat{J}) \geqq 0$ for all $i \in I$.

Proof. This is true at the start of the algorithm since $b_i \geqq 0$. Let k be the last column of $\hat{J}$. Using Property 2.9 we know that Property 2.10 could be altered only if $a_{ik} = 1$ and $i > \alpha(k)$, which is ruled by definition of $\alpha(k)$.

Property 2.11. $b_i(\varnothing) = 0$ for all $i \in I$.

Proof. Let $i \in I$. There exists a $j \in J$ such that $\alpha(j) = i$. At the iteration at which j was the last column of $\hat{J}$ we define $\bar{x}_j = b_i(\hat{J})$ and hence after this iteration we have $b_i(\hat{J}) = 0$. Combining this with Property 2.10 we get $b_i(\varnothing) = 0$.

Property 2.12. If $\bar{y}_k > 0$, $k \notin I$, $\sum_{j \in J} a_{kj}\bar{x}_j \leqq b_k$.

Proof. If $\bar{y}_k > 0$, $k \notin I$, then according to Property 2.1 we have to consider two cases:

1. There is no $j \in J$ such that $a_{kj} = 1$, In this case we have

$$\sum_{j \in J} a_{kj}\bar{x}_j = 0 \leqq b_k.$$

2. There is a $j \in J$ such that $a_{kj} = 1$ and $\alpha(j) > k$ (note that since $k \notin I$ we can rule out $\alpha(j) = k$). Using Properties 2.9 and 2.11 we get $b_k(\varnothing) \geqq b_{\alpha(j)}(\varnothing) = 0$.

Property 2.13. If $\bar{y}_k = 0$, then $\sum_{j \in J} a_{kj}\bar{x}_j \geqq b_k$.

Proof. If $\bar{y}_k = 0$, then according to Property 2.2 there exists a $j \in J$ such that $a_{kj} = 1$ and $\alpha(j) < k$. Using Properties 2.9 and 2.11 we get $b_k(\varnothing) \leqq b_{\alpha(j)}(\varnothing) = 0$.

It follows from Property 2.10 that $\bar{x}_j \geqq 0$ for all $j \in J$. Hence $\bar{x}, \bar{z}$ is a feasible solution. For the complementary slackness relations (2.6) follows by construction, (2.7) and (2.8) follow from Properties 2.11, 2.12 and 2.13.

THEOREM 2.14. *Problem* (D) *is solved by Algorithm* D *for all* $c \geqq 0$, $d \geqq 0$ *and* $b_1 \geqq b_2 \geqq \cdots \geqq b_n \geqq 0$ *if and only if* A *is greedy.*

Proof. If A is greedy, then we transform A into standard greedy form as indicated by a permutation of the columns. This permutation does not affect the dual solution, which was shown to be optimal.

If A is not greedy, then there exists a 3×2 submatrix of the form

$$\begin{bmatrix} 1 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \quad \text{or} \quad \begin{bmatrix} 1 & 1 \\ 0 & 1 \\ 1 & 0 \end{bmatrix}.$$

Let the rows be given by $i_1 < i_2 < i_3$ and columns by $j_1 < j_2$. Set $d_i = 0$ for all $i \notin \{i_1, i_2, i_3\}$, $d_{i_1} = d_{i_2} = d_{i_3} = 1$. $c_j = 3$ for all j except $c_{j_1} = c_{j_2} = 1$, $b_i = 1$ for all $i = 1, 2, \cdots, n$. If we apply Algorithm D we get $\bar{y}_{i_2} = 1$, all other $\bar{y}_i$ are zero. The value of this solution is 1. However $\bar{y}_{i_2} = \bar{y}_{i_3} = 1$ and all other $\bar{y}_i$ are zero is a feasible solution with value 2. This shows that Algorithm D does not solve this instance of problem (D).

3. Standard greedy form transformation. In this section we present an $O(nm^2)$ algorithm to transform an $n \times m$ totally-balanced matrix into standard greedy form. Since a matrix is in standard greedy form if and only if its transpose is in standard greedy form we may assume without loss of generality that $m \leqq n$.

Let us call a (0, 1)-matrix *lexical* if the following two properties hold.

Property 3.1. If rows i_1, i_2 $(i_1 < i_2)$ are different, then the last column in which they differ has a zero in row i_1 and a one in row i_2.

Property 3.2. If columns j_1, j_2 $(j_1 < j_2)$ are different, then the last row in which they differ has a zero in column j_1 and a one in column j_2.

The algorithm we will present in this section transforms any (0, 1)-matrix into a lexical matrix by permuting the rows and by permuting the columns of the matrix. Theorem 3.3 states that a totally-balanced matrix which is lexical is in standard greedy form. Since a totally-balanced matrix is still totally-balanced after a permutation of the rows and a permutation of the columns all we have to do to transform the matrix into standard greedy form is to transform it into a lexical matrix.

THEOREM 3.3. *If a totally-balanced matrix $A = (a_{ij})$ is lexical, then it is in standard greedy form.*

Proof. Suppose A is not in standard greedy form. Then there exist rows i_1, i_2 $(i_1 < i_2)$ and columns j_1, j_2 $(j_1 < j_2)$ such that $a_{i_1 j_1} = a_{i_1 j_2} = a_{i_2 j_1} = 1$ and $a_{i_2 j_2} = 0$ (see Fig. 3.4).

Let i_3 be the last row in which columns j_1 and j_2 differ, and let j_3 be the last column in which rows i_1 and i_2 differ. Since A is lexical we have $a_{i_1 j_3} = 0$, $a_{i_2 j_3} = 1$ and $a_{i_3 j_1} = 0$, $a_{i_3 j_1} = 1$. Since A does not contain a 3×3 submatrix with row and column sums equal to two we know that $a_{i_3 j_3} = 0$. In general we have the submatrix of A given by Fig. 3.4 with ones on the lower and upper diagonal and the first element of the diagonal and zeros everywhere else. The rows and columns have the following properties.

	j_1	j_2	j_3	j_4	⋯	⋯	j_k	⋯
i_1	1	1	0	0	⋯	⋯	0	
i_2	1	0	1	0			0	
i_3	0	1	0	1				
i_4	0	0	1	0				
·	·						0	
·	·						1	
i_k	0				0	1	0	
·	·							

FIG. 3.4. *Submatrix of Theorem* 3.3.

Property 3.5. i_p is the last row in which columns j_{p-2} and j_{p-1} differ $(3 \leqq p \leqq k)$.

Property 3.6. j_p is the last column in which rows i_{p-2} and i_{p-1} differ $(3 \leqq p \leqq k)$.

We shall prove that we can extend this $k \times k$ submatrix to a $(k+1) \times (k+1)$ submatrix with the same properties. So we can extend this submatrix infinitely. This contradicts the fact that A has finite dimensions. Let i_{k+1} be the last row in which j_{k-1} and j_k differ, and let j_{k+1} be the last column in which i_{k-1} and i_k differ. Since A is lexical we have $a_{i_{k+1} j_{k-1}} = 0$, $a_{i_{k+1} j_k} = 1$ and $a_{i_{k-1} j_{k+1}} = 0$, $a_{i_k j_{k+1}} = 1$. By definition of i_p and j_p $(3 \leqq p \leqq k)$ we know that $a_{i_{k+1} j_{p-2}} = a_{i_{k+1} j_{p-1}}$ and $a_{i_{p-2} j_{k+1}} = a_{i_{p-1} j_{k+1}}$. Using this for $p = k, \cdots, 3$ respectively we get $a_{i_{k+1} j_q} = a_{i_q j_{k+1}} = 0$ for $q = 1, 2, \cdots, k-1$. Since A does not contain a $(k+1) \times (k+1)$ submatrix with rows and column sums equal to two we have $a_{i_{k+1} j_{k+1}} = 0$.

Let us now describe the algorithm which transforms any (0, 1)-matrix into a lexical matrix. Let $A = (a_{ij})$ be any (0, 1)-matrix without zero rows and columns. Let us denote

column j by E_j, i.e., $E_j = \{i \mid a_{ij} = 1\}$. We assume that the matrix A is given by its columns $E_1, E_2, \cdots, E_m$. The algorithm produces a 1-1 mapping $\sigma: \{1, 2, \cdots, n\} \to \{1, 2, \cdots, n\}$ corresponding to a transformation of the rows of A ($\sigma(i) = j$ indicates that row i becomes row j in the transformed matrix) and a 1-1 mapping $\tau: \{E_1, \cdots, E_m\} \to \{1, \cdots, m\}$ corresponding to a transformation of the columns of A ($\tau(E_i) = j$ indicates that column i becomes column j in the transformed matrix). We present the algorithm in an informal way and give an example to demonstrate it.

The algorithm consists of m iterations. At iteration i we determine the column E for which $\tau(E) = m - i + 1$ ($1 \leqq i \leqq m$). At the beginning of each iteration the rows are partitioned into a number of groups, say $G_r, \cdots, G_1$. If $i < j$, then rows in G_i will precede rows in G_j in the transformed matrix. Rows j and k belong to the same group G at the beginning of iteration i if and only if for all columns E we have determined so far, i.e. all columns E for which $\tau(E) \geqq m - i + 2$, we cannot distinguish between rows j and k, i.e., $j \in E$ if and only if $k \in E$. At the beginning of iteration 1 all rows belong to the same group. Let $G_r, \cdots, G_1$ be the partitioning into groups at the beginning of iteration i ($1 \leqq i \leqq m$). For each column E not yet determined we calculate the vector d_E of length r, where $d_E(j) = |G_{r-j+1} \cap E|$ ($j = 1, 2, \cdots, r$). A column E for which d_E is a lexicographically largest vector is the column determined at iteration i and $\tau(E) = m - i + 1$. After we have determined E we can distinguish between some rows in the same group G if $1 \leqq |G \cap E| < G$. If this is the case we shall take rows in $G \setminus E$ to precede rows in $G \cap E$ in the transformed matrix. This can be expressed by adjusting the partitioning into groups in the following way. For $j = r, r-1, \cdots, 1$ respectively we check if the intersection of G_j with E is not empty and not equal to G_j. If this is the case we increase the index of all groups with index greater than j by one and partition the group G_j into two groups called G_j and $G_{j+1}$where $G_{j+1} = G_j \cap E$ and $G_j = G_j \setminus E$. The algorithm ends after m interations with a partitioning into groups, say $G_r, \cdots, G_1$. The permutation σ is defined by assigning for $i = 1, 2, \cdots, r$ the values $\sum_{j=1}^{i-1} |G_j| + 1, \cdots, \sum_{j=1}^{i} |G_j|$ in an arbitrary way to the elements in group G_i. The number of computations we have to do at each iteration is $O(mn)$. Therefore the time complexity of this algorithm is $O(nm^2)$.

Example 3.7. The 9×7 (0, 1)-matrix A is given by its columns $E_1 = \{1, 2, 3\}$, $E_2 = (1, 2, 3, 5\}$, $E_3 = \{4, 5\}$, $E_4 = \{3, 4, 5, 9\}$, $E_5 = \{5, 8, 9\}$, $E_6 = \{6, 7, 8, 9\}$, $E_7 = \{6, 7, 8\}$.

Iteration 1: $G_1 = (1, 2, 3, 4, 5, 6, 7, 8, 9)$.
$d_{E_i} = (|E_i|)$, choose E_4, $\tau(E_4) = 7$.

Iteration 2: $G_2 = (3, 4, 5, 9)$, $G_1 = (1, 2, 6, 7, 8)$.

E	E_1	E_2	E_3	E_5	E_6	E_7
d_E	(1, 2)	(2, 2)	(2, 0)	(2, 1)	(1, 3)	(0, 3)

Choose E_2, $\tau(E_2) = 6$.

Iteration 3: $G_4 = (3, 5)$, $G_3 = (4, 9)$, $G_2 = (1, 2)$, $G_1 = (6, 7, 8)$.

E	E_1	E_3	E_5	E_6	E_7
d_E	(1, 0, 2, 0)	(1, 1, 0, 0)	(1, 1, 0, 1)	(0, 1, 0, 3)	(0, 0, 0, 3)

Choose E_5, $\tau(E_5) = 5$.

Iteration 4: $G_7=(5)$, $G_6=(3)$, $G_5=(9)$, $G_4=(4)$, $G_3=(1,2)$, $G_2=(8)$, $G_1=(6,7)$.

E	d_E
E_1	(0, 1, 0, 0, 2, 0, 0)
E_3	(1, 0, 0, 1, 0, 0, 0)
E_6	(0, 0, 1, 0, 0, 1, 2)
E_7	(0, 0, 0, 0, 0, 1, 2)

Choose E_3, $\tau(E_3)=4$.

From now on the groups do not change.

Therefore $\tau(E_1)=3$, $\tau(E_6)=2$, $\tau(E_7)=1$. A mapping σ is given by $\sigma:(6,7,8,1,2,4,9,3,5)\to(1,2,3,4,5,6,7,8,9)$. The mapping τ is given by $\tau:(E_7,E_6,E_1,E_3,E_5,E_2,E_4)\to(1,2,3,4,5,6,7)$.

The transformed matrix is the one used in Example 2.3.

Let us now prove that a matrix transformed by the algorithm is a lexical matrix. When we say that row i is the largest row with respect to σ satisfying a property we mean that there is now row j with $\sigma(j)>\sigma(i)$ satisfying the same property. The same terminology is also used for columns with respect to τ.

LEMMA 3.8. *If rows i and j $(\sigma(i)<\sigma(j))$ are different, then for the largest column E with respect to τ in which they differ we have $i\notin E$, $j\in E$.*

Proof. Consider the last iteration in which i and j are in the same group G and let E be the column determined at this iteration. Since i and j were in the same group during all previous iterations we know that rows i and j are identical when restricted to columns which are larger than E with respect to τ. Since $\sigma(i)<\sigma(j)$ we have that after this iteration row j is in a group with larger index than the group containing row i. This implies that $j\in G\cap E$ and $i\in G\backslash E$, i.e., $i\notin E$ and $j\in E$.

LEMMA 3.9. *If columns E_k and E_l $(\tau(E_k)<\tau(E_l))$ are different, then for the largest row i with respect to σ in which they differ we have $i\notin E_k$ and $i\in E_l$.*

Proof. If E_i is strictly contained in E_j for some i,j, then we always have $\tau(E_i)<\tau(E_j)$. If $E_k\subseteq E_l$, then the lemma holds. So we may assume that $E_k\not\subseteq E_l$ and $E_l\not\subseteq E_k$. Let i be the largest row with respect to σ in $E_l\backslash E_k$, and let j be the largest row with respect to σ in $E_k\backslash E_l$. We have to prove that $\sigma(i)>\sigma(j)$. Consider the iteration in which E_l was determined. Let p be the largest index for which $G_p\cap E_k\neq G_p\cap E_l$. Since E_l was determined before E_k we know that $|G_p\cap E_l|\geqq|G_p\cap E_k|$. We conclude that $i\in G_p$. If $j\in G_f$ with $f<p$, then $\sigma(j)<\sigma(i)$. If $j\in G_p$, then after this iteration G_p is partitioned into two groups $G_p\cap E_l$ and $G_p\backslash E_l$ where $G_p\backslash E_l$ precedes $G_p\cap E_l$. Since $j\in G_p\backslash E_l$ and $i\in G_p\cap E_l$ we have $\sigma(j)<\sigma(i)$.

It follows from Lemmas 3.8 and 3.9 that the transformed matrix is lexical.

In a previous paper (Brouwer and Kolen [4], see also Kolen [10]) it was shown that there exists a row of a totally-balanced matrix such that all columns covering this row can be totally ordered by inclusion. The algorithm presented gives a constructive proof that such a row exists, namely row one of the transformed matrix. As indicated by one of the referees the existence of such a row can be used to derive an $O(n^2m)$ algorithm to transform a totally-balanced matrix into standard greedy form as compared to the $O(nm^2)$ algorithm presented (note $m\leqq n$). The algorithm we gave produces a lexical matrix in standard greedy form. This is important if we consider the following result. Let A be a $n\times m$ $(0,1)$-matrix. The row intersection matrix $B=(b_{ij})$ of A is a $n\times n$ $(0,1)$-matrix defined by $b_{ij}=1$ if and only if there exists a column of A which

covers both row i and j. It is an easy exercise to show that if A is a lexical matrix in standard greedy form, then the row intersection matrix is in standard greedy form. This is not true for any $(0,1)$-matrix A in standard greedy form as is shown by the following example:

$$A=\begin{bmatrix}0&1&1&1\\1&0&0&0\\1&1&1&1\\0&1&1&1\end{bmatrix} \qquad B=\begin{bmatrix}1&0&1&1\\0&1&1&0\\1&1&1&1\\1&0&1&1\end{bmatrix}.$$

Using the results of this section we have proved the following theorem which was first proved by Lubiw [12] by showing that the row intersection matrix of a totally-balanced matrix does not contain one of the forbidden submatrices.

THEOREM 3.10 (Lubiw [12]). *The row intersection matrix of a totally-balanced matrix is totally-balanced.*

If a matrix contains a $k \times k$ submatrix with no identical columns and row and columns sums equal to two, then the matrix transformed by the algorithm still contains such a submatrix and therefore contains $\left[\begin{smallmatrix}1&1\\1&0\end{smallmatrix}\right]$ as a submatrix. Using Theorem 3.3 we conclude that a matrix is totally-balanced if and only if the algorithm transforms the matrix into standard greedy form. We can check in $O(nm^2)$ time whether a matrix is in standard greedy form by comparing each pair of columns.

We finish discussing the relationship between totally-balanced matrices and chordal bipartite graphs. A *chordal bipartite graph* is a bipartite graph for which every cycle of length strictly greater than four has a chord, i.e., an edge connecting two vertices which are not adjacent in the cycle. Chordal bipartite graphs were discussed by Golumbic [9] in relation with perfect Gaussian elimination for nonsymmetric matrices. Chordal bipartite graphs and totally-balanced matrices are equivalent in the following sense:

(3.11) Given a chordal bipartite graph $H=(\{1,2,\cdots,n\},\{1,2,\cdots,m\},E)$ define the $n \times m$ $(0,1)$-matrix $A=(a_{ij})$ by $a_{ij}=1$ if and only if $(i,j) \in E$. Then A is totally-balanced.

Given an $n \times m$ totally-balanced matrix $A=(a_{ij})$ define the bipartite graph $H=(\{1,2,\cdots,n\},\{1,2,\cdots,m\},E)$ by $E=\{(i,j) \mid a_{ij}=1\}$. Then H is a chordal bipartite graph.

An edge (i,j) of a bipartite graph is *bisimplicial* if the subgraph induced by all vertices adjacent to i and j is a complete bipartite graph. Let $M=(m_{ij})$ be a nonsingular nonsymmetric matrix. We can construct a bipartite graph from M equivalent to (3.11) where edges correspond to nonzero elements m_{ij}. If (i,j) is a simplicial edge in the bipartite graph, then using m_{ij} as a pivot in the matrix M to make m_{ij} to one and all other entries in the ith row and jth column equal to zero does not change any zero element into a nonzero element. This is important since sparse matrices are represented in computers by its nonzero elements. Golumbic [9] proved that a chordal bipartite graph has a bisimplicial edge. This result immediately follows from our result. The first one in the first row corresponds to a bisimplicial edge.

REFERENCES

[1] R. P. ANSTEE AND M. FARBER (1982), *Characterizations of totally balanced matrices*, Research Report CORR 82-5, Faculty of Mathematics, Univ. Waterloo, Waterloo, Ontario, Canada.

[2] A. V. AHO, J. E. HOPCROFT AND J. D. ULLMAN (1974), *The Design and Analysis of Computer Algorithms*, Addison-Wesley, Reading, MA.
[3] C. BERGE (1972), *Balanced matrices*, Math. Programming, 2, pp. 19-31.
[4] A. E. BROUWER AND A. KOLEN (1980), *A super-balanced hypergraph has a nest point*, Report ZW 146/80, Mathematisch Centrum, Amsterdam.
[5] M. FARBER (1982), *Characterization of strongly chordal graphs*, Discrete Math., 43 (1983), pp. 173-189.
[6] ———, (1982), *Domination, independent domination and duality in strongly chordal graphs*, Research Report CORR 82-2, Faculty of Mathematics, Univ. Waterloo, Waterloo, Ontario, Canada.
[7] D. R. FULKERSON, A. J. HOFFMAN AND R. OPPENHEIM (1974), *On balanced matrices*, Math. Programming Study, 1, pp. 120-132.
[8] R. GILES (1978), *A balanced hypergraph defined by certain subtrees of a tree*, Ars Combinatoria, 6, pp. 179-183.
[9] M. C. GOLUMBIC (1980), *Algorithmic Graph Theory and Perfect Graphs*, Academic Press, New York.
[10] A. KOLEN (1982), *Location problems on trees and in the rectilinear plane*, Ph.D. thesis, Mathematisch Centrum, Amsterdam.
[11] L. LOVÁSZ (1979), *Combinatorial Problems and Exercises*, Akadémiai Kiadó, Budapest, p. 528.
[12] A. LUBIW (1982), *Γ-free matrices*, Master thesis, Univ. Waterloo, Waterloo, Ontario, Canada.
[13] A. TAMIR (1980), *A class of balanced matrices arising from location problems*, this Journal, 4 (1983), pp. 363-370.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 4, October 1985

019

ON THE MATRIX ADJOINT (ADJUGATE)*

RICHARD D. HILL† AND E. EUGENE UNDERWOOD‡

Abstract. A unified treatment of rank, eigenvalues, minimal polynomial, minors and factorizations of the adjoint is given. The roles of the adjoint in determinantal identities and the adjoints of characteristic matrices and modified matrices are discussed.

1. Introduction. Let $M_n(F)$ denote the space of $n \times n$ matrices with entries from the field F. The adjoint (adjugate) of a matrix $A \in M_n(F)$ is defined by giving its (i,j)th element $(\operatorname{adj} A)_{ij} = (-1)^{i+j} \det A(j|i)$, where $A(j|i)$ is the $n-1$ square submatrix of A formed by deleting its jth row and ith column; i.e., adj A is the transpose of the matrix of cofactors of the elements of A.

Information on the adjoint occurs infrequently in linear algebra texts and literature. Most adjoint papers concern the (nonrelated) linear operator whose matrix representation is A^* (conjugate transpose).

In this paper we collect some basic results. Then we give a unified treatment of the rank, eigenvalues, minimal polynomial and minors of the adjoint. Next we consider different factorizations followed by the appearance of the adjoint in determinantal identities. Finally we summarize information on adjoints of characteristic matrices and modified matrices.

2. Standard results. Computationally we may verify a result found in many linear algebra texts:

$$A(\operatorname{adj} A) = (\operatorname{adj} A)A = (\det A)I_n. \tag{1}$$

This result immediately gives us a standard formulation for the inverse, viz.,

$$\text{if } A \text{ is nonsingular, } A^{-1} = \frac{1}{\det A} \operatorname{adj} A, \tag{2}$$

i.e.,

$$\operatorname{adj} A = (\det A)A^{-1}; \tag{2'}$$

which says that if A is nonsingular, adj A is a multiple of A^{-1}. Taking determinants of (1) above gives us that

$$\det(\operatorname{adj} A) = (\det A)^{n-1}. \tag{3}$$

Application of (1) and (3) leads to

$$\operatorname{adj}(\operatorname{adj} A) = (\det A)^{n-2}A \quad \text{for } n > 2. \tag{4}$$

Taking determinants in (4) we get

$$\det[\operatorname{adj}(\operatorname{adj} A)] = \det A^{(n-1)^2}. \tag{5}$$

Although (1) may be used to prove the reverse multiplicative property

$$\operatorname{adj} AB = (\operatorname{adj} B)(\operatorname{adj} A), \tag{6}$$

a powerful but seldomly used theorem, the Cauchy-Binet theorem [9, p. 38], [10,

* Received by the editors June 13, 1983, and in revised form July 1, 1984.
† Department of Mathematics, Idaho State University, Pocatello, Idaho 83209.
‡ Utah State University, Logan, Utah 84322.

p. 128], gives us a quick proof as follows:

$$\begin{aligned}((\text{adj } B)(\text{adj } A))_{ij} &= \sum_{k=1}^{n} (\text{adj } B)_{ik}(\text{adj } A)_{kj} \\ &= \sum_{k=1}^{n} (-1)^{i+2k+j} \det B(k|i) \det A(j|k) \\ &= (-1)^{i+j} \sum_{k=1}^{n} \det A(j|k) \det B(k|i) \\ &= (-1)^{i+j} \det AB(j|i) \\ &= (\text{adj } AB)_{ij}, \qquad i,j = 1, \cdots, n.\end{aligned}$$

Some basic results on the adjugate are now immediate. Considering adj: $M_n(F) \to M_n(F)$, we see that adj is not linear. In fact, adj is neither additive nor homogeneous; adj $cA = c^{n-1}$ adj A. We observe that adj is neither injective nor surjective if $n > 2$. Nevertheless, adj $A^{-1} = (\text{adj } A)^{-1}$ for nonsingular A; adj $A^r = (\text{adj } A)^r$ for all positive integers r; and adj $A^* = (\text{adj } A)^*$ where * denotes the conjugate transpose.

If two matrices A and B commute, then adj A and adj B commute; if A and B are similar, then adj A and adj B are similar.

Considering special matrices, if A is normal, Hermitian or unitary, then adj A necessarily possesses the same property. Also, if A is skew-Hermitian and n (the order of A) is even (odd), then adj A is skew-Hermitian (Hermitian). Further, if A is simple (diagonalizable), then adj A is simple. The converses of these results are in general false.

3. Rank, eigenvalues and minimal polynomial. We now characterize the rank of A, $\rho(A)$, in terms of the rank of adj A. Using the natural representation of our rank characterization (Theorem 1), all the results of this section (and one of the next section) partition corresponding to $\rho(A)$ being $\leqq n-2$, $n-1$ or n. A more formal setting is natural for these results.

THEOREM 1. *If $A \in M_n(F)$, then*

(i) $\rho(\text{adj } A) = 0$ iff $\rho(A) \leqq n-1$;

(ii) $\rho(\text{adj } A) = 1$ iff $\rho(A) = n-1$;

(iii) $\rho(\text{adj } A) = n$ iff $\rho(A) = n$.

Proof. First we observe that $\rho(A) \leqq n-2$ iff adj $A = 0$. If $\rho(A) = n-1$, then there exist nonsingular P and Q such that $A = P(I_{n-1} \oplus 0)Q$, where $\oplus$ denotes the direct sum. Then adj $A = $ adj Q adj $(I_{n-1} \oplus 0)$ adj P. Since adj Q and adj P are nonsingular and $\rho(\text{adj } (I_{n-1} \oplus 0))$ is one, we have that $\rho(\text{adj } A) = 1$. Since $A(\text{adj } A) = (\det A) I_n$, $\rho(A) = n$ iff $\rho(\text{adj } A) = n$. □

If $\rho(A) = n-1$, we note that we have the dyad product representation adj $A = uv^*$ where u spans the null space of A and v spans the null space of A^*.

We next investigate the minimal polynomial of A (the monic annihilating polynomial of least degree). We use triag $\{a_1, \cdots, a_n\}$ to denote an upper triangular matrix whose main diagonal elements $a_1, \cdots, a_n$ are the only elements of concern.

THEOREM 2. *If $A \in M_n(F)$, the minimal polynomial of* adj A *is*

(i) x *if* $\rho(A) \leqq n-2$;

(ii) $x^2 - \Lambda x$, *where* $\Lambda = \prod_{i=1}^{n-1} \lambda_i$ *with* $\{\lambda_1, \cdots, \lambda_{n-1}, 0\}$, *the family of eigenvalues of* A, *if* $\rho(A) = n-1$;

(iii)

$$x^k + \frac{a_1}{a_0}\Delta x^{k-1} + \frac{a_2}{a_0}\Delta^2 x^{k-2} + \cdots + \Delta^k,$$

where $x^k + a_{k-1}a^{k-1} + \cdots + a_0$ *is the minimal polynomial of* A *and* $\Delta = \det A$, *if* $\rho(A) = n$.

Proof. If $\rho(A) \leqq n-2$, adj $A = 0$, which gives us that the minimal polynomial of adj A is x. If $\rho(A) = n-1$, we assume adj A to be in Jordan normal form, since similar matrices have the same minimal polynomials. Then adj $A = \text{triag}\{0, \cdots, 0, \Lambda\}$, where $\{\lambda_1, \cdots, \lambda_{n-1}, 0\}$ is the family of eigenvalues of A and $\Lambda = \prod_{i=1}^{n-1} \lambda_i$. Thus, the first $n-1$ columns of adj A consist entirely of zeros. Since $(\text{adj } A)^2 = \Lambda \text{ adj } A$, adj A has $x^2 - \Lambda x$ as its minimal polynomial.

If $\rho(A) = n$, A is nonsingular, so that the constant term is nonzero in the minimal polynomial of A. If $x^k + a_{k-1}x^{k-1} + \cdots + a_0$ is the minimal polynomial, then $A^k + a_{k-1}A^{k-1} + \cdots + a_0 I = 0$ implies that

$$(A^{-1})^k + \frac{a_1}{a_0}(A^{-1})^{k-1} + \cdots + \frac{1}{a_0} I_n = 0.$$

Thus, $x^k + (a_1/a_0)x^{k-1} + \cdots + 1/a_0$ is the minimal polynomial for A^{-1}. Letting $M(x) = x^k + (a_1/a_0)\Delta x^{k-1} + (a_2/a_0)\Delta^2 x^{k-1} + \cdots + \Delta^k$, where $\Delta = \det A$, adj $A = \Delta A^{-1}$, gives us that $M(\text{adj } A) = 0$. □

Let $Q_{k,n}$ denote the set of all strictly increasing sequences of length k chosen from $\{1, \cdots, n\}$. As above, if necessary, we consider $\bar{F}$, the algebraic closure of F, to insure a family of n eigenvalues for $A \in M_n(F)$. To express the eigenvalues of adj A in terms of the eigenvalues of A, we again use our rank characterization theorem to split into cases.

THEOREM 3. *If $\lambda_1, \cdots, \lambda_n$ are the eigenvalues of $A \in M_n(F)$, then*

(i) *all eigenvalues of* adj A *are zero if $\rho(A) \leqq n-2$;*

(ii) *the eigenvalues of* adj A *are zero with multiplicity $n-1$ and Λ (cf. Theorem 2) (0 has multiplicity n if $\Lambda = 0$) if $\rho(A) = n-1$;*

(iii) *the eigenvalues of* adj A *are $\prod_{i=1}^{n-1} \lambda_{\omega_i}$, where $\lambda_1, \cdots, \lambda_n$ are the eigenvalues of A and $\omega \in Q_{n-1,n}$ (equivalently, $(\det A)/\lambda_i$, $i = 1, \ldots, n$) if $\rho(A) = n$.*

Proof. If $\rho(A) \leqq n-2$, then $\rho(\text{adj } A) = 0$, and all the eigenvalues of adj A are zero. If $\rho(A) = n$, adj $A = (\det A)A^{-1}$; the eigenvalues of adj A are $(\det A)/\lambda_i$, $i = 1, \cdots, n$.

If $\rho(A) = n-1$, then A has a zero eigenvalue and there exists a nonsingular T such that $T^{-1}AT = \text{triag}\{\lambda_1, \cdots, \lambda_{n-1}, 0\}$, where $\lambda_1, \cdots, \lambda_{n-1}$ are the other eigenvalues of A. Thus, adj $T^{-1}AT = \text{triang}\{0, \cdots, 0, \Lambda\}$. Since adj $T^{-1}AT = \text{adj } T \text{ adj } A \text{ adj } T^{-1}$, adj A is similar to triag $\{0, \cdots, 0, \Lambda\}$ and thus has eigenvalues Λ and zero of multiplicity at least $n-1$. □

4. Factorization. The following theorem expresses the adjoint of a block diagonal matrix in terms of the adjoints of the blocks. As well as being of interest in and of itself, it leads to our factorization theorem. We denote the block diagonal matrix diag $\{A_1, \cdots, A_1\}$ (where each A_i is square) by $\sum_{i=1}^{\circ k} A_i = A_i \oplus \cdots \oplus A_k$.

THEOREM 4. *If $A = \sum_{i=1}^{\circ k} A_i \in M_n(F)$, then*

$$\text{adj } A = \sum_{i=1}^{k}{}^{\circ} \left(\prod_{\substack{l=1 \\ l \neq i}}^{k} \det A_l \right) \text{adj } A_i.$$

Proof. If $\rho(A) \leqq n-2$, adj $A = 0$, and the result follows. If $\rho(A) = n$,

$$\text{adj } A = \det A \sum_{i=1}^{k}{}^{\circ} A_i^{-1} = \sum_{i=1}^{k}{}^{\circ} \det A \left(\frac{1}{\det A_i} \text{adj } A_i \right)$$

$$= \sum_{i=1}^{k}{}^{\circ} \left(\prod_{\substack{l=1 \\ l \neq i}}^{k} \det A_l \right) \text{adj } A_i.$$

If $\rho(A)=n-1$, without loss of generality we assume $A_2,\cdots,A_k$ to be nonsingular with the order $\sigma(A_1)=\rho(A_1)+1$. Since $A(\operatorname{adj}A)=(\operatorname{adj}A)A=0$, $\operatorname{adj}A=\operatorname{diag}\{B_1,0,\cdots,0\}$ with $\sigma(B_1)=\sigma(A_1)$. For $a_{ij}\in A_1$, its cofactor in A is

$$(-1)^{i+j}\left[\prod_{l=2}^{k}\det A_l\right]\det A_1(i\,|\,j)=\prod_{l=2}^{k}\det A_l[\text{cofactor of } a_{ij} \text{ in } A_1].$$

Thus, $B_1=(\prod_{l=2}^{k}\det A_l)\operatorname{adj}A_1$. □

Toward our factorization result, let A have exactly p elementary divisors. Then $A=T^{-1}JT$, where $J=\sum_{i=1}^{p\circ}J_i$ is the Jordan normal form of A. By (6), $\operatorname{adj}A=\operatorname{adj}T\operatorname{adj}J\operatorname{adj}T^{-1}$, where

$$\operatorname{adj}J=\sum_{i=1}^{p\circ}\left(\prod_{\substack{l=1\\ l\neq i}}^{p}\det J_l\right)\operatorname{adj}J_i.$$

Letting

$$F_r=\operatorname{diag}\left\{I,\cdots,I,\left(\prod_{\substack{i=1\\ i\neq r}}^{p}\det J_i\right)\operatorname{adj}J_r,I,\cdots,I\right\},$$

where the rth block is the nonidentity block, $r=1,\cdots,p$; we have $(\operatorname{adj}T)F_r(\operatorname{adj}T^{-1})$ as p factors of $\operatorname{adj}A$ which can be collapsed into q factors, where q is any integer such that $1\leqq q\leqq p$. We state our result as follows:

THEOREM 5. *If $A\in M_n(F)$ has exactly p nontrivial elementary divisors (Jordan blocks), then* $\operatorname{adj}A$ *admits a factorization into q factors, where q is an integer such that $1\leqq q\leqq p$.*

We conclude this section with an interesting factorization by Taussky [15] of $\operatorname{adj}A$ into $n-1$ factors *each of which has determinant equal to* $\det A$. Let $x^n+c_1x^{n-1}+\cdots+c_{n-1}x+c_n$ be the characteristic polynomial of A; let $\lambda_1,\cdots,\lambda_n$ be the eigenvalues of A; and let $\mu_2,\cdots,\mu_n$ be the zeros of $x^{n-1}+c_1x^{n-2}+\cdots+c_{n-1}$ with $\mu_1=0$.

Assuming $\rho(A)=n$, $A^{n-1}+c_1A^{n-2}+\cdots+c_{n-1}I=(-1)^{n-1}\det A\cdot A^{-1}=(-1)^{n-1}\operatorname{adj}A$. Thus, $\operatorname{adj}A=\prod_{i=2}^{n}(\mu_iI-A)$. Now the algebraic equation $\prod_{i=1}^{n}(\lambda_i-x)=\prod_{i=1}^{n}\lambda_i$ of degree n in x has n roots, viz., $\mu_1=0,\mu_2,\cdots,\mu_n$. Since $\det A=\prod_{i=1}^{n}\lambda_i$ and $\prod_{i=1}^{n}(\lambda_i-x)=\det(A-xI)$, we have the following theorem. (The cases where A is singular follow by specialization.)

THEOREM 6. *If $A\in M_n(F)$,* $\operatorname{adj}A=\prod_{i=2}^{n}(\mu_iI-A)$, *where* $\det(A-\mu_iI)=\det A$, $i=2,\cdots,n$.

5. Minors of the adjoint. Our next result gives an arbitrary minor of $\operatorname{adj}A$ in terms of a minor of A. For $\beta\in Q_{m,n}$ define $s(\beta)$ to be the sign of the permutation which takes β to $\{1,2,\cdots,m\}$ elementwise. See [10, p. 126] for submatrix notation.

THEOREM 7. *If $\beta,\gamma\in Q_{m,n}$, $m>2$, and $A\in M_n(F)$, then* $\det(\operatorname{adj}A[\beta\,|\,\gamma])=s(\beta)s(\gamma)\det A(\beta\,|\,\gamma)(\det A)^{m-1}$.

Proof. The result obviously holds for singular A. Assuming A nonsingular, $\operatorname{adj}A=(\det A)A^{-1}$ gives us that

$$\det(\operatorname{adj}A[\beta\,|\,\gamma])=(\det A)^m\det A^{-1}[\beta\,|\,\gamma]=s(\beta)s(\gamma)\det A(\gamma\,|\,\beta)(\det A)^{m-1},$$

since in general [8, p. 14], $s(\beta)s(\gamma)\det A=\det A(\gamma\,|\,\beta)/\det A^{-1}[\beta\,|\,\gamma]$. □

Older texts (e.g. Browne [3]) call $s(\beta)s(\gamma)\det(\beta\,|\,\gamma)$ the algebraic complement of $\det[\beta\,|\,\gamma]$. For a computational proof of this result see [3, pp. 43–44] or [1, pp. 51–52].

If $A=\left(\begin{smallmatrix}E & F\\ G & H\end{smallmatrix}\right)$ is partitioned so that $E=A[1,\cdots,k\,|\,1,\cdots,k]$ is nonsingular, $H-GE^{-1}F=A/E$ is said to be the Schur complement of E in A. In general, $\det A=(\det E)(\det A/E)$.

Specializing to $k=n-1$ so that H is the 1×1 matrix $[a_{nn}]$ and F and G are column and row vectors respectively, Brualdi and Schneider [4, p. 772] observe that $\det A=(\det E)(a_{nn}-GE^{-1}F)=a_{nn}\det E-G(\operatorname{adj} E)F$, which is called the Cauchy expansion of $\det A$.

6. Adjoints in determinantal identities. Brualdi and Schneider [4] give a self-contained development of a large number of determinantal identities using only the basic facts that Gaussian elimination does not alter the determinant and that a determinant has a Laplacian expansion.

In addition to the aforementioned Cauchy expansion of $\det A$, they have an immediate specialization of Theorem 7, viz.,

$$\det(\operatorname{adj} A)[k+1,\cdots,n\,|\,k+1,\cdots,n]=(\det A)^{n-k-1}\det A[1,\cdots,k\,|\,1,\cdots,k]. \tag{7}$$

Brualdi and Schneider standardize a definition for a determinantal identity for the minors of a matrix. Applying a determinantal identity to $\operatorname{adj} A$ yields another $(n\times n)$ determinantal identity called a complementary identity or "the law of complementaries" by Muir in his classic [12], which he attributes to Cayley, e.g., the complementary identity to the definition of the determinant is the result (3).

7. Adjoints of characteristic matrices. For $A\in M_n(F)$, $\lambda I-A$ is said to be the characteristic matrix of A. Its determinant is the characteristic polynomial of A; its roots are the eigenvalues of A.

The quotient $q(\lambda)$ of the characteristic polynomial $c(\lambda)$ by the minimal polynomial $m(\lambda)$ of A is the greatest common divisor of the elements of $\operatorname{adj}(\lambda I-A)$ [9, p. 135]. Then $\operatorname{adj}(\lambda I-A)=q(\lambda)R(\lambda)$, where $R(\lambda)$ is called the reduced adjoint. If $M(\lambda,\mu)$ is the two-variable polynomial defined by $M(\lambda,\mu)=(m(\lambda)-m(\mu))/\lambda-\mu$, then $M(\lambda I, A)=R(\lambda)$. If $C(\lambda,\mu)=(c(\lambda)-c(\mu))/\lambda-\mu$, then $C(\lambda I, A)=\operatorname{adj}(\lambda I-A)$, i.e., we have another formulation of the adjoint of the characteristic matrix.

We further observe that if λ_j is an eigenvalue of A, then the nonzero columns of $\operatorname{adj}(\lambda_j I-A)$ (and $R(\lambda_j)$ as well) are right eigenvectors of A associated with the eigenvalue λ_j.

The invariant polynomial of largest degree is the quotient of $\det(\lambda I-A)$ by the gcd of the elements of $\operatorname{adj}(\lambda I-A)$; again we have $m(\lambda)$. Also, (1) gives us that

$$(\lambda I-A)\operatorname{adj}(\lambda I-A)=c(\lambda)I, \tag{8}$$

which easily leads to the Hamilton-Cayley theorem [9, p. 131].

Frazer, Duncan, and Collar [6, pp. 165-167] discuss the adjoint of a (general) polynomial matrix and its derivatives as well as their specialization to the characteristic matrix [6, pp. 73-78].

8. Modified matrices. In [5] Elsner and Rozsa study the behavior of the adjoint under the rank one modification $A\mapsto A+uv^*$. The multilinearity of the determinant leads to the one previously known result:

$$\det(A+uv^*)=\det A+v^*\operatorname{adj}(A)u. \tag{9}$$

The results of Elsner and Rozsa partition as the ranks of A and $A+uv^*$ each permute between n and $n-1$.

In particular, if $\rho(A)=n-1$ and $\rho(A+uv^*)=n$, up to a scalar the difference $\operatorname{adj}(A+uv^*)-\operatorname{adj}A=D$ is a $\{1,2\}$-inverse of A, i.e., $ADA=A$ and $DAD=D$. Conversely, any $\{1,2\}$-inverse of A can be expressed as a difference of adjoints in the above form.

If $\rho(A)=\rho(A+uv^*)=n-1$, formulations for adj $(A+uv^*)$ vary accordingly as $u\in,\notin$ Rng A and $v\in,\notin$ Rng A^*.

Further, letting A^+ denote the Moore-Penrose inverse and $C(A, uv^*)=[\text{adj}\,(A+uv^*)-\text{adj}\,A]/v^*(\text{adj}\,A)u$ (note that $v^*(\text{adj}\,A)u\neq 0$), the following sets are equal:

$$
\begin{aligned}
&\{C(A, uv^*)\colon u, v\in F^n;\ v(\text{adj}\,A)u\neq 0\},\\
&\{C\colon CAC=C,\ ACA=A\},\\
&\{(I-rv^*/v^*r)A^+(I-us^*/s^*u)\colon v^*r,\ s^*u\neq 0\},\\
&\{A^+-rg^*-hs^*+g^*Ah\cdot rs^*\colon g^*s=r^*h=0\},\\
&\{(A^+-rg^*)A(A^+-hs^*)\colon g^*s=r^*h=0\},\\
&\{A^+-rg^*-hs^*\colon g^*s/s^*s+r^*h/r^*r+g^*Ah=0\}.
\end{aligned}
\tag{10}
$$

Finally, Elsner and Rozsa [5, p. 247] derive a representation for the adjoint of the bordered matrix

$$
\text{adj}\begin{pmatrix} A & u\\ v^* & \alpha\end{pmatrix}=\begin{pmatrix}(\alpha+1)\,\text{adj}\,A-\text{adj}\,(A+uv^*) & -\text{adj}\,(A)u\\ -v^*\,\text{adj}\,A & \det A\end{pmatrix}.
\tag{11}
$$

9. Some random comments. 1. Mirski [11, pp. 87-90] gives a development of most of the results found in § 2. Ayres [1, pp. 49-54] also gives many of these results.

2. The rank characterization (Theorem 1) is found in some form (e.g., as a problem) in Frazer, Duncan and Collar [6, p. 21], Thrall and Tornheim [16, p. 130], Nomizu [13, p. 175] and Brinkmann and Klotz [2, p. 262]. The reverse multiplicative property (6) is stated for the nonsingular case in Schneider and Barker [14, p. 205] and Greub [7, p. 115] (incorrectly), and for the general case in Mirsky [11, p. 90].

3. Frazer, Duncan and Collar [6, pp. 121-125] discuss at some length the computation of the adjoint of a matrix of rank $n-1$.

4. Schneider and Barker [14, p. 204] credit L. A. Gavin in the American Mathematical Monthly, Jan. 1966, with the following result: Let $A\in M_n(F)$. If there exists an integer m such that $A^m=I$, then $[(\text{adj}\,A)^T]^m=I$.

REFERENCES

[1] F. Ayres, Jr., *Theory and Problems of Matrices*, Schaum, New York, 1962.

[2] H. Brinkmann and E. Klotz, *Linear Algebra and Analytic Geometry*, Addison-Wesley, Reading, MA, 1971.

[3] E. T. Browne, *The Theory of Determinants and Matrices*, Univ. North Carolina Press, Chapel Hill, 1958.

[4] R. A. Brualdi and H. Schneider, *Determinantal identities: Gauss, Schur, Cauchy, Sylvester, Kronecker, Jacobi, Binet, Laplace, Muir and Cayley*, Linear Algebra and Appl., 52/53 (1983), pp. 769-791.

[5] L. Elsner and P. Rozsa, *On eigenvectors and adjoints of modified matrices*, Linear Multilinear Algebra, 10 (1981), pp. 235-247.

[6] R. A. Frazer, W. J. Duncan and A. R. Collar, *Elementary Matrices*, Macmillan, New York, 1947.

[7] W. H. Greub, *Linear Algebra*, Springer-Verlag, New York, 1967.

[8] A. S. Householder, *The Theory of Matrices in Numerical Analysis*, Blaisdell, New York, 1964.

[9] P. Lancaster, *Theory of Matrices*, Academic Press, New York, 1969.

[10] M. Marcus and H. Minc, *Elementary Linear Algebra*, Macmillan, New York, 1968.

[11] L. Mirsky, *An Introduction to Linear Algebra*, Clarendon Press, Oxford, 1955.

[12] T. Muir, *The Theory of Determinants in the Historical Order of Development*, 4 vols., Macmillan, New York, 1906, 1911, 1920, 1923; Dover, New York, 1966.

[13] K. NOMIZU, *Fundamentals of Linear Algebra*, McGraw-Hill, New York, 1966.
[14] H. SCHNEIDER AND G. P. BARKER, *Matrices and Linear Algebra*, Holt, Rinehart and Winston, New York, 1973.
[15] O. TAUSSKY, *The factorization of the adjugate of a finite-matrix*, Linear Algebra and Appl., 1 (1968), pp. 39–41.
[16] R. M. THRALL AND L. TORNHEIM, *Vector Spaces and Matrices*, John Wiley, New York, 1957.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 4, October 1985

020

ORDER PRESERVING MAPS AND LINEAR EXTENSIONS OF A FINITE POSET*

DAVID E. DAYKIN† AND JACQUELINE W. DAYKIN‡

Abstract. We study order preserving maps from a finite poset to the integers. When these maps are bijective they are called linear extensions. For both kinds we give many elementary properties and inequalities. A positive correlation inequality was proved by Graham, Yao and Yao. Then contributions were made by Graham, Kleitman, Shearer, Shepp and others. We obtain the corresponding negative correlation inequalities. Most authors have used the FKG inequality; we use an inequality of Daykin instead. Graham made a conjecture concerning range posets so we characterise these, and prove various cases of the conjecture. Finally we give necessary and sufficient conditions for a map defined on a subposet to extend to the whole poset. The results have applications in computer science.

Key words. Daykin's inequality, FKG inequality, linear extension, order preserving map extension, poset, positive correlation inequality, range poset

AMS (1980) subject classification. 06A10

1. Introduction. We are concerned with order preserving maps from a finite poset to a subset of the integers. When these maps are bijective they are called linear extensions.

This paper was motivated by the results in § 3. They are of the form that, if all maps are equally likely, then the probability that a random map will have certain properties is positively correlated. Other authors have proved such results via the classical FKG inequality, but we use an inequality of D. E. Daykin. Also we give some new results on negative correlations. The interesting Conjecture 3.1 involves the concept of a range poset, so we characterise these in § 9. The conjecture is supported by Theorems 3.2, 3.3, 3.4 and 3.7. These include fundamental results of Graham, Yao and Yao and of Shepp.

In §§ 4, 5 and 6 we deal with order preserving maps. First we obtain elementary properties using the rake-down of a map over a set. Secondly we give various inequalities. Lastly we discuss the problem of extending a map defined on a subset of the poset to the whole poset. In §§ 7 and 8 we obtain the parallel results for linear extensions, except that the analogous inequalities do not arise.

The results in § 3 have applications in computer science. Most algorithms for sorting a finite set F of real numbers compare the numbers two at a time. A set of comparisons reveals a partial order P of F. The algorithm halts when P becomes the underlying total order of F. Fundamental quantities in determining the expected efficiency of such algorithms are those appearing in (3.1) below (cf. [G1], [G2], [GYY]).

2. Notation and definitions. We let $\mathbb{Z}$ denote the integers, and by an *interval* I we mean a subset of $\mathbb{Z}$ of the form $I=[i,j]=\{k\in\mathbb{Z}: i\leqq k\leqq j\}$. The letter J denotes a finite interval $[1,j]$ so j is the cardinality $|J|$ of J.

We let P be a finite poset with $|P|\neq 0$. A map $\omega: P\to\mathbb{Z}$ is *strict order preserving* if $p, q\in P$ and $p<q$ imply $\omega p<\omega q$, but we omit the word strict. We let Ω denote the set of all order preserving maps $\omega: P\to J$. Notice that Ω depends on the given J. By a *linear extension* of P we mean a bijective order preserving map $\lambda: P\to J$ so $|J|=|P|$.

* Received by the editors November 8, 1983, and in final form August 8, 1984
† Mathematics Department, University of Reading, Berkshire, England RG6 2AX.
‡ Computer Science Department, University of Warwick, Coventry, England CV4 7AL.

The set of all such λ is Λ. The *pre-image* of $K \subset J$ under $\omega \in \Omega$ is the subset $\omega^{-1}K = \{p \in P: \omega p \in K\}$, and under Ω it is $\Omega^{-1}K = \bigcup\{\omega^{-1}K: \omega \in \Omega\}$. For $S \subset P$ the *range* of S over Ω is the subset $\Omega S = \{\omega p: \omega \in \Omega, p \in S\}$ of J. There are similar pre-images and ranges for Λ.

We call $D \subset P$ a *down-set* if $p < q \in D$ imply $p \in D$. Similarly $U \subset P$ is an *up-set* if $p > q \in U$ imply $p \in U$. For $S \subset P$ the intersection of all down-sets containing S is called *below* S. Then *above* S is the corresponding intersection of up-sets. The *convex hull* $\bar{S}$ of S is (above S) $\cap$ (below S), and S is *convex* if $\bar{S} = S$. Notice that down-sets and up-sets are convex, and the empty set $\varnothing$ is both a down-set and an up-set.

For $S \subset P$ the *height* $h(S)$ of S is the maximum m for which we have a chain $s_1 < s_2 < \cdots < s_m$ in S. Here and elsewhere we count vertices as opposed to edges. For $p \in P$ the *height* $h(p)$ of p is $h(\text{below}\{p\})$, and the *depth* $d(p)$ of p is $h(\text{above}\{p\})$.

For $p, q \in P$ we write $p \sim q$ if either $p < q$ or $p > q$. The opposite of $p \sim q$ is written $p|q$ and means that p and q are *incomparable*. If S, T are posets we write $S|T$ to mean that $s|t$ for all $s \in S$, $t \in T$. Also $S < T$ means $s < t$ for all $s \in S$, $t \in T$.

3. Probability results. We will use a result from [D1], namely:

THEOREM 3.1 (D. E. Daykin). *If A, B are subsets of the elements of a distributive lattice then $|A||B| \leqq |A \vee B||A \wedge B|$, where*

$$A \vee B = \{a \vee b: a \in A, b \in B\}, A \wedge B = \{a \wedge b: a \in A, b \in B\}.$$

A simple proof was given by Ahlswede and Daykin [AD], and modified by Graham [G1]. The theorem has various generalisations (cf. [D2]).

Throughout this section we assume that the poset P has been partitioned as $P = Q \cup R$. We deal with results of the form "$P = Q \cup R$ has the $\mathcal{NP}$ or $\mathcal{PP}$ for Γ", where Γ is Ω or Λ. So we now define these terms.

DEFINITION 3.1. The partition $P = Q \cup R$ has the *$\mathcal{NP}$ negative correlation property* for Γ if, whenever x (respectively y) is a disjunction of conjunctions of inequalities in which each inequality has the form $q < r$ (respectively $q > r$) with $q \in Q$, $r \in R$, we have

$$|\Gamma|\{\Gamma: x \text{ and } y\}| \leqq |\{\Gamma: x\}||\{\Gamma: y\}|.$$

If for example $q_1, q_2, q_3 \in Q$ and $r_1, r_2, r_3 \in R$ and x was the condition "either $q_1 < r_1$ or both $q_2 < r_2$ and $q_3 < r_3$", then $\{\Gamma: x\}$ denotes the set of all γ in Γ such that either $\gamma q_1 < \gamma r_1$ or both $\gamma q_2 < \gamma r_2$ and $\gamma q_3 < \gamma r_3$. Of course it is not assumed that $q_i \sim r_i$ or that $q_i | r_i$ in P.

DEFINITION 3.2. The partition $P = Q \cup R$ has the *$\mathcal{PP}$ positive correlation property* for Γ if, whenever both x and y are like the x in Definition 3.1, we have

$$|\{\Gamma: x\}||\{\Gamma: y\}| \leqq |\Gamma||\{\Gamma: x \text{ and } y\}|. \tag{3.1}$$

We chose the title "Probability Results" for this section because many authors divide (3.1) by $|\Gamma|^2$ and express the result as

$$(\text{Probability: } x)(\text{Probability: } y) \leqq \text{Probability: } (x \text{ and } y).$$

The reader may find it easier to understand our discussion by referring to Table 1, which presents Theorems 3.2–3.6. All our work stems from the result of Graham, Yao and Yao in the first line of the table, the first part of Theorem 3.2. It says that if $P = Q \cup R$ and Q, R are disjoint chains then P has the $\mathcal{PP}$ for Λ. Proofs of this result have also been given by Kleitman and Shearer [KS] and by Shepp [S1]. When $A = \{\Lambda: x\}$ and $B = \{\Lambda: y\}$, the $\mathcal{PP}$ for Λ follows immediately from Theorem 3.1. Secondly one can let $A = \Lambda$ and $B = \{\Lambda: x \text{ and } y\}$ and get the $\mathcal{NP}$ for Λ. So we have Theorem 3.2.

TABLE 1.
Theorems and conjectures.

Theorem	Poset $P = Q \cup R$	Ω		Λ
3.2	Q, R chains			$\mathcal{PP}$ [GYY] $\mathcal{NP}$
3.3	$Q \mid R$	$\mathcal{PP}$ [S1] $\mathcal{NP}$	$\Rightarrow$ $\Rightarrow$	$\mathcal{PP}$ [S1] $\mathcal{NP}$
3.4	Range poset with $m = 1$	$\mathcal{PP}$ $\mathcal{NP}$	$\Rightarrow$ $\Rightarrow$	$\mathcal{PP}$ $\mathcal{NP}$
3.5	$\lvert Q \rvert = 1$	$\mathcal{PP}$ [S2] $\mathcal{NP}$	$\Rightarrow$ $\Rightarrow$	$\mathcal{PP}$ [S2] $\mathcal{NP}$
3.6	If $q \sim r$ then $(\Omega q) \cap (\Omega r) = \varnothing$	$\mathcal{PP}$ [G1] $\mathcal{NP}$	$\not\Rightarrow$ $\not\Rightarrow$	

Range poset conjecture 3.1.

	Ω		Λ
Range poset	$\mathcal{PP}$ if $\lvert J \rvert$ large $\mathcal{NP}$ if $\lvert J \rvert$ large	$\Rightarrow$ $\Rightarrow$	$\mathcal{PP}$ [G1] $\mathcal{NP}$

One feels that as $|J| \to \infty$ the proportion of members of Ω which are injective tends to 1, and this was proved by Shepp [S1], giving all the implications $\Rightarrow$ in Table 1. Further it shows that we can replace Λ by Ω provided that $|J|$ is sufficiently large.

Next consider Theorem 3.6. Here it is assumed that the poset $P = Q \cup R$ is such that

$$q \in Q, r \in R, q \sim r \Rightarrow (\Omega q) \cap (\Omega r) = \varnothing. \tag{3.2}$$

Since Ω depends on J, this assumption depends on J, and by Lemma 5.4, it is not satisfied for $|J|$ large. Hence we have written $\not\Rightarrow$ in Table 1 to indicate that we cannot here deduce a result for Λ by letting $|J| \to \infty$. Lemma 5.4 gives an obvious fast algorithm for checking if P satisfies (3.2), but there does not appear to be a nice characterisation of such posets. The hypothesis of Theorem 3.3 implies (3.2). So Theorem 3.3 follows from Theorem 3.6 and by letting $|J| \to \infty$.

Proof of Theorem 3.6. Let Q, R be disjoint sets and Θ be the set of all maps $\theta: Q \cup R \to J$. For $\theta_1, \theta_2 \in \Theta$ define $\theta_1 \vee \theta_2$, $\theta_1 \wedge \theta_2$ for $q \in Q$, $r \in R$ by

$$(\theta_1 \vee \theta_2)q = \max\{\theta_1 q, \theta_2 q\}, (\theta_1 \wedge \theta_2)q = \min\{\theta_1 q, \theta_2 q\},$$

$$(\theta_1 \vee \theta_2)r = \min\{\theta_1 r, \theta_2 r\}, (\theta_1 \wedge \theta_2)r = \max\{\theta_1 r, \theta_2 r\},$$

and then it follows that

$$\theta_1 q < \theta_1 r \Rightarrow (\theta_1 \wedge \theta_2)q < (\theta_1 \wedge \theta_2)r, \tag{3.3}$$

$$\theta_1 q > \theta_1 r \Rightarrow (\theta_1 \vee \theta_2)q > (\theta_1 \vee \theta_2)r. \tag{3.4}$$

Clearly $\theta_1 \vee \theta_2$, $\theta_1 \wedge \theta_2 \in \Theta$.

Let $q_1, q_2, \cdots, q_m$ and $r_1, r_2, \cdots, r_n$ be the elements of Q and R respectively. With each $\theta \in \Theta$ associate the vector $(\theta q_1, \cdots, \theta q_m, \theta r_1, \cdots, \theta r_n)$ considered as an element of the lattice $L = J^m(J^*)^n$, where J^* is J with its order reversed. Since L is a direct product of copies of J and J^* it is distributive. Also $\theta_1 \vee \theta_2$ and $\theta_1 \wedge \theta_2$ are the usual join and meet respectively in L.

Next we claim that if $\omega_1, \omega_2 \in \Omega \subset \Theta$ then $\omega_1 \vee \omega_2, \omega_1 \wedge \omega_2 \in \Omega$. This fact is easily established with the aid of (3.2), (3.3), (3.4) and Lemma 4.4.

Now we use Theorem 3.1. For the $\mathcal{PP}$ case let $A = \{\Omega: x\}$ and $B = \{\Omega: y\}$. Then $A \wedge B \subset \{\Omega: x \text{ and } y\}$ by (3.3), and trivially $A \vee B \subset \Omega$. For the $\mathcal{NP}$ case let $A = \{\Omega: x \text{ and } y\}$ and $B = \Omega$. Then $A \wedge B \subset \{\Omega: x\}$ by (3.3), while $A \vee B \subset \{\Omega: y\}$ by (3.4). □

As shown in Table 1, the $\mathcal{PP}$ case of Theorem 3.5 is due to Shepp, and it established the XYZ conjecture of Rival and Sands. He used a different lattice, but modifying his proof in the foregoing manner easily yields the $\mathcal{NP}$ case.

Once Graham had written down (3.2) it was natural for him to write down

$$q \in Q, r \in R, q \sim r \Rightarrow (\Lambda q) \cap (\Lambda r) = \varnothing. \tag{3.5}$$

He then made the most interesting part of Conjecture 3.1 in Table 1, suggesting that if $P = Q \cup R$ satisfies (3.5) then it has the $\mathcal{PP}$ for Λ. We studied condition (3.5) and will prove in Theorem 9.1 that the only posets which satisfy this condition are what we now define to be range posets.

DEFINITION 3.3. We say that $P = Q \cup R$ is a *range poset* if there are partitions

$$Q = Q_1 \cup Q_2 \cup \cdots \cup Q_m, \qquad R = R_1 \cup R_2 \cup \cdots \cup R_n$$

such that

$$Q_i < Q_j \quad \text{for } 1 \leqq i < j \leqq m, \tag{3.6}$$

$$R_i < R_j \quad \text{for } 1 \leqq i < j \leqq n, \tag{3.7}$$

$$\text{either } Q_i | R_j \text{ or } Q_i < R_j \text{ or } R_j < Q_i \text{ for } 1 \leqq i \leqq m, 1 \leqq j \leqq n. \tag{3.8}$$

For example, the posets of Theorems 3.2 to 3.4 are all range posets, but not those of Theorems 3.5 and 3.6. Using Lemma 9.1 below it is an easy exercise to deduce Theorem 3.4 from Theorem 3.3. Thus we have Theorems 3.2 to 3.4 supporting the range poset conjecture. For Ω the main interest of the conjecture is in the sufficient size for $|J|$. That some condition on $|J|$ is necessary is shown by the N-shaped poset with 4 elements. The same example shows why there is no entry for Ω in Theorem 3.2.

We end this section by establishing a weak form of Conjecture 3.1. Let P be the range poset in Definition 3.3. We will call $Q_1, Q_2, \cdots, Q_m$ the *blocks* of Q and $R_1, R_2, \cdots, R_n$ the blocks of R.

DEFINITION 3.4. We say that P has the *weak* $\mathcal{NP}$ if Definition 3.1 holds except that now the inequalities for y are between blocks instead of elements. Thus they now have the form $Q_i > R_j$ with $1 \leqq i \leqq m, 1 \leqq j \leqq n$ instead of the form $q > r$ with $q \in Q, r \in R$.

DEFINITION 3.5. We say that P has the *weak* $\mathcal{PP}$ if Definition 3.2 holds except that now the inequalities for y are between blocks. Notice that the inequalities for x are still between elements, and that those for y can be expressed in terms of large numbers of inequalities between elements. Our result is:

THEOREM 3.7. *A range poset has the weak $\mathcal{PP}$ and the weak $\mathcal{NP}$ for Λ.*

Proof. If $\lambda \in \Lambda$ and $S \subset P$ there is a unique ordering $s_1, s_2, \cdots, s_{|S|}$ of the elements of S such that $\lambda s_1 < \lambda s_2 < \cdots < \lambda s_{|S|}$. We call $s_1 < s_2 < \cdots < s_{|S|}$ the chain which λ makes out of S. For all $\lambda, \mu \in \Lambda$ we write $\lambda \rho \mu$ if λ and μ make the same chain out of S for

every block S of P. Thus ρ is an equivalence relation for Λ and we let $\mathscr{E}$ denote the set of equivalence classes of ρ.

Let $E, F \in \mathscr{E}$ be chosen arbitrarily and fixed until we get to (3.9). Given a block S of P let $s_1 < \cdots < s_{|S|}$ be the chain which every $\lambda \in E$ makes out of S. Also let $t_1 < \cdots < t_{|S|}$ be the chain which every $\mu \in F$ makes out of S. Then define $\sigma_S: S \to S$ by $\sigma_S s_i = t_i$ for $1 \leqq i \leqq |S|$. Clearly the map $\mu\sigma_S: S \to \mathbb{Z}$ is order preserving.

Next we define $\pi: P \to P$ by $\pi p = \sigma_S p$ for all p in each block S. Then π is a permutation of P because each σ_S permutes its block S. An important point is that $\mu S = \mu\pi S$ for every $\mu \in F$ and block S. Since y is defined in terms of inequalities between blocks we see that $\mu \in F$ respects y iff $\mu\pi$ respects y. In the last paragraph we showed that $\mu\pi$ is order preserving on each block. Because P is a range poset and $\mu \in F \subset \Lambda$ it easily follows that $\mu\pi \in \Lambda$. This in turn implies that the map $\mu \to \mu\pi$ is an injection $F \to E$. Hence $|F| \leqq |E|$ and so $|E| = |F|$ by symmetry. Further $\mu \to \mu\pi$ is an injection $\{F: y\} \to \{E: y\}$.

Recall that any $\lambda \in E$ makes a chain $s_1 < \cdots < s_{|S|}$ out of each block S. Let P_E be the poset obtained by adjoining to P all the relations in all these chains. Thus P_E is of the form $P_E = Q_E \cup R_E$ where Q_E, R_E are the chains which every $\lambda \in E$ makes out of Q, R respectively. Also E is simply the set of all linear extensions of P_E. Hence we can apply Theorem 3.2 to P_E to get

$$|\{E: x\}||\{E: y\}| \leqq |E||\{E: x \text{ and } y\}|,$$

and so

$$|\{E: x\}||\{F: y\}| \leqq |F||\{E: x \text{ and } y\}|. \tag{3.9}$$

Forming the double sum of (3.9) over all $E, F \in \mathscr{E}$ gives

$$|\{\Lambda: x\}||\{\Lambda: y\}| \leqq |\Lambda||\{\Lambda: x \text{ and } y\}|,$$

which is the weak $\mathscr{PP}$.

Using the above results in the usual way one can easily obtain the weak $\mathscr{NP}$, and the proof is complete. □

4. The set Ω or order preserving maps. The map $p \to h(p)$ is in Ω iff $h(P) \leqq |J|$. Also

$$h(p) \leqq \omega p \leqq |J| + 1 - d(p) \quad \text{if } p \in P, \omega \in \Omega. \tag{4.1}$$

Hence we immediately get the next theorem.

THEOREM 4.1. *We have $\Omega \neq \varnothing$ iff $h(P) \leqq |J|$.*

From now on we think of P and J as fixed with $\Omega \neq \varnothing$. For any $\omega \in \Omega$ and $S \subset P$ we define a map $\pi: P \to \mathbb{Z}$ as follows. If $S = \varnothing$ then $\pi = \omega$. If $S \neq \varnothing$ we let $m = \max\{\omega s: s \in S\}$ and $T = \{s \in S: \omega s = m\}$. If there is a $t \in T$ with $h(t) = m$ then $\pi = \omega$. Otherwise we construct $R \subset P$ by starting with $R = T$ and iterating the rule that, if $p \in P$, $r \in R$, $p < r$ and $1 + \omega p = \omega r$ then p must be adjoined to R. Finally π is defined by $\pi p = (\omega p) - 1$ if $p \in R$ but $\pi p = \omega p$ otherwise. Clearly $\pi \in \Omega$ and we call π the *rake down of ω over S*. The rake up of ω over S is defined similarly. An obvious result is:

LEMMA 4.1. *In the above notation, $|\omega S| - 1 \leqq |\pi S| \leqq |\omega S|$, and if ωS is an interval then πS is an interval.*

The height function $h(p)$ may not map a convex set onto an interval, but it is easy to see:

LEMMA 4.2. *If $D \subset P$ is a down-set then $h(D) = [1, h(D)]$.*

LEMMA 4.3. *Let $S_1, S_2, \cdots, S_n$ be pairwise disjoint subsets of P satisfying*

$$s_i \in S_i, s_j \in S_j, s_i < s_j, i \neq j \Rightarrow i < j. \tag{4.2}$$

Suppose that $(n+1)h(P) \leqq |J|$. *Then there is an* $\omega \in \Omega$ *satisfying both*

$$s_i \in S_i,\ s_j \in S_j,\ i<j \Rightarrow \omega s_i < \omega s_j, \tag{4.3}$$

(4.4) *for* $1 \leqq i \leqq n$ *if* S_i *is convex then* ωS_i *is an interval of length* $h(S_i)$.

Proof. Put $U_0 = P$, $U_{n+1} = \varnothing$ and $U_i = \text{above}(S_i \cup S_{i+1} \cup \cdots \cup S_n)$ for $1 \leqq i \leqq n$ so $U_{n+1} \subset U_n \subset \cdots \subset U_0$. For $0 \leqq i \leqq n$ put $T_i = U_i \setminus U_{i+1}$. Then T_i considered as a possibly empty poset has its own height function h_i. For $1 \leqq i \leqq n$ we have $S_i \subset T_i$ and if S_i is convex in P then S_i is a down-set in T_i and Lemma 4.2 applies. Finally define ω by $\omega p = ih(P) + h_i(p)$ if $p \in T_i$ for $0 \leqq i \leqq n$.

This lemma generalises a result of Mirsky [M].

LEMMA 4.4. *If* $p \in P$ *then* $\Omega p = [h(p), |J| + 1 - d(p)]$.

Proof. Starting with the map $h(p)$ rake up repeatedly over $\{p\}$.

LEMMA 4.5. *If* $V \subset P$ *is convex,* $I \subset J$ *is an interval,* $j \in J$ *and* $\omega \in \Omega$ *then*

(4.5) ΩV *is an interval provided* $2h(P) \leqq |J|$,

(4.6) $\Omega^{-1} I$ *is convex,*

(4.7) $\omega^{-1} I$ *is convex and in particular* $\omega^{-1} j$ *is an antichain.*

Proof. For (4.5) use Lemmas 4.3 and 4.4.

LEMMA 4.6. *Suppose that* $V \subset P$ *is convex, and put* $k = h(V)$, $m = h(\text{below } V)$, $n = h(\text{above } V)$. *Also suppose that* $m + n - k \leqq |J|$. *If the interval* $I \subset [1 + m - k, |J| - n + k]$ *has length* $|I| = k$ *then there is an* $\omega \in \Omega$ *with* $\omega V = I$.

Proof. Starting with the map $p \to |J| + 1 - d(p)$ rake down repeatedly over V. □

5. Inequalities for order preserving maps.

LEMMA 5.1. If $\varnothing \neq S \subset P$ *and* $p < q$ *in* P *but not* $p < r < q$ *in* P *then*

$$h(S) + |J| - h(P) \leqq |\Omega S|, \tag{5.1}$$

$$4 \leqq d(p) + h(q) \leqq 2 + |P|, \tag{5.2}$$

$$h(p) + d(q) \leqq h(P), \tag{5.3}$$

$$h(p) + d(p) \leqq 1 + h(P). \tag{5.4}$$

LEMMA 5.2. *If* $K \subset J$ *then* $|K| \leqq |\Omega^{-1} K| \leqq |P|$ *if* $|K| < h(P)$ *but* $\Omega^{-1} K = P$ *otherwise.*

Proof. We may assume that $|K| \leqq n = h(P)$ and that $K \subset \{k_1, k_2, \cdots, k_n\} \subset J$. The result then follows because the map $p \to k_{h(p)}$ is in Ω. □

Next we prove what Graham and Harper called normalised matching conditions (5.5), (5.6). It is well known that each implies the other.

LEMMA 5.3. *If* $|P| \leqq |J|$ *and* $S \subset P$ *and* $K \subset J$ *then both*

$$|S||J| \leqq |\Omega S||P|, \tag{5.5}$$

$$|K||P| \leqq |\Omega^{-1} K||J|. \tag{5.6}$$

Proof. We prove (5.6) using Lemma 5.2. If $|K| < h(P)$ then $|K| \leqq |\Omega^{-1} K|$ and we multiply this by $|P| \leqq |J|$. If $h(P) \leqq |K|$ then $|P| = |\Omega^{-1} K|$ and we multiply this by $|K| \leqq |J|$.

In view of Theorem 3.6 we mention an easy consequence of Lemma 4.4.

LEMMA 5.4. *If* $p < q$ *in* P *and* $(\Omega p) \cap (\Omega q) = \varnothing$ *then* $|J| \leqq d(p) + h(q) - 2$.

In view of Theorem 3.2 we mention another triviality.

LEMMA 5.5. *Suppose that* $P = Q \cup R$ *where* Q, R *are disjoint chains. If* $q \in Q$, $r \in R$ *and* $q < r$ *then* $h(P) \leqq d(q) + h(r) - 2$.

6. Completion of order preserving maps. Let $S \subset P$ and $Y \subset \mathbb{Z}$ and $\psi: S \to Y$ be order preserving. We say ψ *extends* if there is an order preserving map $\xi: P \to Y$ with $\xi s = \psi s$ for all $s \in S$. It was the concept of *intricacy* [B] that motivated Theorems 6.3 and 8.3.

For $p < q$ in P we let $c(p, q)$ be the maximum integer m for which there is a chain $p = p_1 < p_2 < \cdots < p_m = q$ in P, so $c(p, q) = h(\{p, q\})$.

THEOREM 6.1. *Let $S \subset P$ and $\psi: S \to \mathbb{Z}$ be order preserving. Then ψ extends iff*

$$c(s, t) - 1 + \psi s \leqq \psi t \quad \text{for all } s < t \text{ in } S. \tag{6.1}$$

Proof. Clearly condition (6.1) is necessary for ψ to extend, and it implies that ψ is order preserving. So we assume that (6.1) holds. We let p be any point of $P \setminus S$ and proceed to define ψp so that (6.1) still holds in $S \cup p$. Since P is finite repetition will yield an extension of ψ.

Case. We have $s < p$ for some $s \in S$. Here we put

$$\psi p = \max \{c(s, p) - 1 + \psi s: s \in S, s < p\}.$$

We must show that if $p < t \in S$ then

$$c(p, t) - 1 + \psi p \leqq \psi t.$$

Now there is an $r \in S$ with $r < p$ and

$$\psi p = c(r, p) - 1 + \psi r.$$

Also

$$c(r, p) + c(p, t) - 1 \leqq c(r, t) \leqq 1 - \psi r + \psi t,$$

so the required inequality follows.

Case. We have $p < s$ for some $s \in S$ but not $s' < p$, $s' \in S$. Here we put

$$\psi p = \min \{1 - c(p, s) + \psi s: s \in S, p < s\}.$$

Case. We have $p | s$ for all $s \in S$. Here we give ψp any value in $\mathbb{Z}$, and the proof of Theorem 6.1 is complete.

THEOREM 6.2. *Let $S \subset P$ and $\psi: S \to J$ be order preserving. Then ψ extends to $\omega \in \Omega$ iff both* (6.1) *holds and*

$$h(s) \leqq \psi s \leqq |J| + 1 - d(s) \quad \text{for all } s \in S. \tag{6.2}$$

Proof. In view of (4.1) the conditions are clearly necessary. To prove the sufficiency suppose (6.1), (6.2) hold. Take two new elements r, t not in P. Define a new poset $Q = P \cup \{r, t\}$ by taking the existing relations of P and adding the new relations $r < p < t$ for all $p \in P$. Similarly extend J to $[0, |J| + 1]$. Then define $\psi r = 0$ and $\psi t = |J| + 1$. The result now follows by applying Theorem 6.1 to Q in the obvious way. □

THEOREM 6.3. *Let $S \subset P$ and $\psi: S \to \mathbb{Z}$ be order preserving. Let $k = \lceil h(P)/2 \rceil$. Then there is a partition $S = S_1 \cup S_2 \cup \cdots \cup S_k$ such that ψ extends from any one S_i.*

Proof. For $1 \leqq i \leqq k$ put

$$P_i = \{p \in P: 2i - 1 \leqq h(p) \leqq 2i\} \quad \text{and} \quad S_i = S \cap P_i.$$

In each S_i a chain has at most two vertices, so (6.1) holds, and ψ extends from S_i by Theorem 6.1.

Consider the example where P is $[1, m]$ and S is the odd numbered vertices and $\psi(2i - 1) = i$. This shows that the k in Theorem 6.3 cannot be reduced.

7. The set Λ of linear extensions. Here we have $|P|=|J|$, and our results are numbered to contrast and correlate with those for Ω in § 4. First we have

$$|\text{below}\{p\}| \leqq \lambda p \leqq |P|+1-|\text{above}\{p\}| \quad \text{if } p \in P, \lambda \in \Lambda. \tag{7.1}$$

THEOREM 7.1 (Szpilrajn 1930). *We always have* $\Lambda \neq \varnothing$.

For any $\lambda \in \Lambda$ and $S \subset P$ we define a map $\mu: P \to \mathbb{Z}$ as follows. Put $\beta = \max\{\lambda s: s \in S\}$ with the value 0 by convention if $S = \varnothing$. If $|\text{below } S| = \beta$ then $\mu = \lambda$. Otherwise we put

$$Q = \{p \in P \backslash \text{below } S: \lambda p < \beta\}.$$

Since $|\text{below } S| < \beta$ there is a unique $q \in Q$ with λq maximal. We put

$$R = \{r \in \text{below } S: \lambda q < \lambda r \leqq \beta\}$$

and observe that $q|r$ for all $r \in R$. Finally μ is defined for this case by

$$\mu p = \begin{cases} \beta & \text{if } p = q, \\ (\lambda p) - 1 & \text{if } p \in R, \\ \lambda p & \text{otherwise.} \end{cases}$$

Clearly $\mu \in \Lambda$ and we call μ the *push down of* λ *over* S. The push up of λ over S is defined similarly. An obvious result is:

LEMMA 7.1. *In the above notation, if* λS *is an interval then* μS *is an interval.*

LEMMA 7.2. *If* $D_1 \subset D_2 \subset \cdots \subset D_n$ *are down-sets of* P *there is a* $\lambda \in \Lambda$ *such that for all* $p \in P$, $1 \leqq i \leqq n$ *we have* $\lambda p \leqq |D_i|$ *iff* $p \in D_i$.

Proof. Push down repeatedly over $D_1, D_2, \cdots, D_n$ in any order. □

LEMMA 7.3. *Let* $S_1, S_2, \cdots, S_n$ *be pairwise disjoint subsets of* P *satisfying*

$$s_i \in S_i,\ s_j \in S_j,\ s_i < s_j,\ i \neq j \Rightarrow i < j. \tag{7.2}$$

Then there is a $\lambda \in \Lambda$ *satisfying both*

$$s_i \in S_i,\ s_j \in S_j,\ i < j \Rightarrow \lambda s_i < \lambda s_j, \tag{7.3}$$

$$\text{for } 1 \leqq i \leqq n \text{ if } S_i \text{ is convex then } \lambda S_i \text{ is an interval.} \tag{7.4}$$

Proof. For $1 \leqq i \leqq n$ let $D_i = \text{below}(S_1 \cup S_2 \cup \cdots \cup S_i)$. Then $D_1 \subset D_2 \subset \cdots \subset D_n$ are down-sets with $D_i \cap S_j = \varnothing$ for $1 \leqq i < j \leqq n$. Let λ_0 be the map of Lemma 7.2. Let λ_{01} be λ_0 restricted to D_1. Let λ_{02} be the result of repeatedly pushing λ_{01} up over S_1 in D_1. Notice that if S_1 is convex, then S_1 is an up-set in D_1, and hence $\lambda_{02} S_1$ is an interval by the dual version of Lemma 7.2. Now define λ_1 on P to be λ_{02} on D_1 and λ_0 elsewhere. In general we repeatedly push up λ_{i-1} over S_i in the poset $D_i \backslash D_{i-1}$ and the lemma follows.

LEMMA 7.4. *If* $p \in P$ *then* $\Lambda p = [|\text{below}\{p\}|, |P|+1-|\text{above}\{p\}|]$.

Proof. Push up and down repeatedly over $\{p\}$. □

LEMMA 7.5. *If* $V \subset P$ *is convex,* $I \subset J$ *is an interval and* $\lambda \in \Lambda$ *then*

$$\Lambda V \text{ is an interval,} \tag{7.5}$$

$$\Lambda^{-1} I \text{ is convex,} \tag{7.6}$$

$$\lambda^{-1} I \text{ is convex.} \tag{7.7}$$

LEMMA 7.6. *Suppose that* $V \subset P$ *is convex, and put* $k = |V|$, $m = |\text{below } V|$, $n = |\text{above } V|$. *If the interval* $I \subset [1+m-k, |P|-n+k]$ *has length* $|I| = k$ *then there is a* $\lambda \in \Lambda$ *with* $\lambda V = I$.

Proof. Starting with the λ of Lemma 7.3 push up and down repeatedly over V. □

8. Linear extensions from order preserving maps. This section is parallel to § 6. Again we let $S \subset P$ and $\psi: S \to \mathbb{Z}$ be order preserving. For $R \subset S$ we define

$$\max R = \max\{\psi r: r \in R\}, \qquad \min R = \min\{\psi r: r \in R\}, \tag{8.1}$$

with the convention $\max \varnothing = \min \varnothing = 0$. The main result is:

THEOREM 8.1. *If $S \subset P$ and $\psi: S \to \mathbb{Z}$ is order preserving then ψ extends to an order preserving injection iff*

$$|\bar{V}| \leqq \max V - \min V + 1 \quad \textit{for all } V \subset S. \tag{8.2}$$

Proof. We assume that (8.2) holds and that $\varnothing \neq S \neq P$.

Case 1. There is a $p \in P \backslash S$ for which there is no $s \in S$ with $p < s$. We simply let z be such a p and choose ψz sufficiently large.

Case 2. There is a $p \in P \backslash S$ for which there is no $t \in S$ with $t < p$. Let z be such a p and ψz be sufficiently small.

Case 3. For every $p \in P \backslash S$ there are $s, t \in S$ with $t < p < s$.

(After we had given a preprint of this paper to R. Aharoni he very kindly gave us the following simpler proof of this case.)

We will use some notation. For $p \in P$ put

$$\alpha p = \max\{\psi s: s \in S, s \leqq p\}, \qquad \beta p = \min\{\psi s: s \in S, p \leqq s\},$$

so $\alpha p \leqq \beta p$. Then for each p we have unique $p_*, p^* \in S$ with $\psi p_* = \alpha p$, $\psi p^* = \beta p$ and $p_* \leqq p \leqq p^*$. If $p \in S$ then $p_* = p^* = p$. Let G be the bipartite graph with parts P and $\mathbb{Z}$, where for all $p \in P$ and $i \in \mathbb{Z}$ there is an edge (p, i) iff $\alpha p \leqq i \leqq \beta p$. Each $p \in S$ is on exactly one edge. For any set A of vertices of G let $N(A)$ be the set of neighbouring vertices of A.

Now we show that if $T \subset P$ then $|T| \leqq |N(T)|$. Since $N(T) \subset \mathbb{Z}$ it is a union of disjoint intervals $N(T) = I_1 \cup I_2 \cup \cdots \cup I_n$ where each I_i is maximal by inclusion in $N(T)$. For $1 \leqq i \leqq n$ put $T_i = T \cap N(I_i)$. Then it is easy to see that $T = T_1 \cup T_2 \cup \cdots \cup T_n$ is a partition of T. Put

$$W_i = \cup\{\{p_*, p^*\}: p \in T_i\} \quad \text{and} \quad V_i = \bar{W}_i.$$

Then $W_i \subset S$ and $T_i \subset V_i$ and

$$N(T_i) = N(V_i) = [\min W_i, \max W_i] = I_i.$$

Using (8.2) we get

$$|T_i| \leqq |V_i| \leqq \max W_i - \min W_i + 1 = |I_i|.$$

Then summing over i shows that $|T| \leqq |N(T)|$.

In the last paragraph we merely established the condition for G to have a matching by Hall's theorem (1935). The matching is an injective function $\xi: P \to \mathbb{Z}$ such that $(p, \xi p)$ is an edge of G for every $p \in P$. Thus $\alpha p \leqq \xi p \leqq \beta p$ for all $p \in P$. If $p \in S$ then $\alpha p = \beta p = \psi p$ and there is only one edge on p in G. This shows that $\xi = \psi$ on S.

For $r, t \in P$ let us call the ordered pair (r, t) *bad* for ξ if $r < t$ but $\xi t < \xi r$. If there is no bad pair then ξ is order preserving. So suppose that (r, t) is bad. Then we have

$$\alpha r \leqq \alpha t \leqq \xi t < \xi r \leqq \beta r \leqq \beta t.$$

We define another injection $\xi': P \to \mathbb{Z}$ by $\xi' r = \xi t$, $\xi' t = \xi r$ but $\xi' p = \xi p$ otherwise. By the last inequality $\alpha p \leqq \xi' p \leqq \beta p$ for all $p \in P$, so $\xi' = \psi$ on S. Since $r < t$ we have $h(r) < h(t)$. So if $F(\xi) = \sum\{(\xi p) h(p): p \in P\}$ then clearly $F(\xi) < F(\xi')$. It follows that

using bad pairs as above, a finite number of times, will lead to a choice of ξ having no bad pairs, and hence order preserving. □

THEOREM 8.2. *Let $S \subset P$ and $\psi: S \to [1, |P|]$ be order preserving. Then ψ extends to $\lambda \in \Lambda$ iff for all $V \subset S$ we have* (8.2) *and both*

$$|\text{above } V| \leqq |P| - \min V + 1,$$

$$|\text{below } V| \leqq \max V.$$

Proof. Similar to that of Theorem 6.2. □

THEOREM 8.3. *Let $S \subset P$ and $\psi: S \to \mathbb{Z}$ be an order preserving injection. Let $k = \lceil h(P)/2 \rceil$. Then there are $\lambda_1, \lambda_2, \cdots, \lambda_k \in \Lambda$ and a partition $S = S_1 \cup S_2 \cup \ldots \cup S_k$ such that ψ extends to λ_i from S_i for any one i.*

Proof. Similar to that of Theorem 6.3. □

Problem 8.1. Find the versions of Theorems 6.1 and 8.1 for P countably infinite and for P noncountably infinite.

9. Range posets.

THEOREM 9.1. *If $P = Q \cup R$ is a partition of P then the following are equivalent*:

(9.1) $$q \in Q, r \in R, q \sim r \Rightarrow (\Lambda q) \cap (\Lambda r) = \varnothing,$$

(9.2) $$\begin{cases} q \in Q, r \in R, q < r \Rightarrow P = (\text{above } \{q\}) \cup (\text{below } \{r\}), \\ q \in Q, r \in R, q > r \Rightarrow P = (\text{below } \{q\}) \cup (\text{above } \{r\}), \end{cases}$$

(9.3) *P is a range poset as in Definition* 3.3.

Proof. Part (9.1)⇒(9.2). Suppose that $q \in Q, r \in R, q < r$. We assume the worst circumstances in which there is no $p \in P$ with $q < p < r$. We put $U = \text{above } \{q\}$, $D = \text{below } \{r\}$ and then $|U \cap D| = |\{q, r\}| = 2$. Since $|P| = |Q| + |R|$, by Lemma 7.4 we get $|P| + 2 - |U| \leqq |D|$. On the other hand $|U| + |D| = 2 + |U \cup D| \leqq 2 + |P|$, so we have equality throughout, and the first of the conditions in (9.2) follows. The second then follows in turn by symmetry.

Part (9.2)⇒(9.3). For each $q \in Q$ put

$$U(q) = R \cap \text{above } \{q\} \quad \text{and} \quad D(q) = R \cap \text{below } \{q\}.$$

CLAIM 9.1. *If $q_1, q_2 \in Q$ and either $U(q_1) \setminus U(q_2) \neq \varnothing$ or $D(q_2) \setminus D(q_1) \neq \varnothing$ then $q_1 < q_2$.*

Proof. Suppose that $r \in U(q_1) \setminus U(q_2)$. Then $q_1 < r$ so by (9.2) we have $q_2 \in P = (\text{above } \{q_1\}) \cup (\text{below } \{r\})$. Now $r \notin U(q_2)$ so $q_2 \not< r$ so $q_2 \notin \text{below } \{r\}$ so $q_2 \in \text{above } \{q_1\}$ so $q_1 < q_2$. The rest of the claim follows in the same way.

Next we define an equivalence relation ρ in Q by putting $q_1 \rho q_2$ iff both $U(q_1) = U(q_2)$ and $D(q_1) = D(q_2)$. Then an immediate consequence of Claim 9.1 is:

CLAIM 9.2. *If $q_1, q_2 \in Q$ and $q_1 | q_2$ then $q_1 \rho q_2$.*

Let Q_1, Q_2 be different nonempty equivalence classes of ρ and let $q_1 \in Q_1$, $q_2 \in Q_2$. By Claim 9.2 we have $q_1 \sim q_2$. Since $Q_1 \neq Q_2$ by definition of ρ we have $U(q_1) \neq U(q_2)$ or $D(q_1) \neq D(q_2)$. We may assume either $U(q_1) \setminus U(q_2) \neq \varnothing$ or $D(q_2) \setminus D(q_1) \neq \varnothing$. Then Claim 9.1 shows that $q_1 < q_2$. If $q_3 \in Q_1$ then $q_1 \rho q_3$ so the same argument shows that $q_3 < q_2$. Hence we have proved that the equivalence classes of ρ are totally ordered. In other words we have obtained Claim 9.3.

CLAIM 9.3. *We can let $Q_1, Q_2, \cdots, Q_m$ be the equivalence classes of ρ numbered so that* (3.6) *holds.*

Similarly we can assume that $R_1, R_2, \cdots, R_n$ are equivalence classes of R numbered so that (3.7) holds.

To see that (3.8) holds let $q<r$ with $q\in Q_i$, $r\in R_j$. If q_1 is also in Q_i then $r\in U(q)=U(q_1)$ so $q_1<r$.

Part (9.3)$\Rightarrow$(9.1). Let $q\in Q_i$, $r\in R_j$ with $q<r$. In view of Lemma 7.4, if $\Lambda q=[i,j]$ and $\Lambda r=[i',j']$ then

$$\begin{aligned} j &= |P|+1-|\text{above}\,\{q\}| \\ &\leqq |P|+1-(|\{q\}|+|Q_{i+1}|+\cdots+|Q_m|+|R_j|+\cdots+|R_n|) \\ &= |Q_1|+\cdots+|Q_i|+|R_1|+\cdots+|R_{j-1}| \\ &< |\text{below}\,\{r\}| = i'. \end{aligned}$$

Hence $(\Lambda q)\cap(\Lambda r)=\varnothing$ and (9.1) follows.

LEMMA 9.1. *We can assume that* $m\leqq 2n+1$ *and* $n\leqq 2m+1$ *in Definition* 3.3.

Motivated by (9.2) we make the following conjecture

Conjecture 9.1. Let P be covered by three nonempty disjoint chains T_1, T_2, T_3. Suppose that if $p, q\in P$ are in different chains and $p<q$ then $P=(\text{above}\,\{p\})\cup(\text{below}\,\{q\})$. Then there is a partition $P=R_1\cup\cdots\cup R_n$ such that (3.7) holds, and further for $1\leqq i\leqq n$, either $R_i\cap T_j=\varnothing$ for some j, or if $p, q\in R_i$ are in different chains then $p|q$.

If this conjecture is true then with the help of Theorem 3.2 we get a probability result based on conditions $t_i<t_j$ with $1\leqq i<j\leqq 3$.

Acknowledgment. We are very grateful to M. S. Paterson, the supervisor of J. W. Daykin's Ph.D. thesis, for much valuable assistance.

REFERENCES

[AD] R. AHLSWEDE AND D. E. DAYKIN, *An inequality for the weights of two families of sets, their unions and intersections*, Z. Wahrsch. Verw. Gebiete, 43 (1978), pp. 183-185.

[B] R. A. BAILEY et al., *On the intricacy of combinatorial construction problems*, Discrete Math., 50 (1984), pp. 71-97.

[D1] D. E. DAYKIN, *A lattice is distributive iff* $|A||B|\leqq|A\vee B||A\wedge B|$, Nanta Math., 10 (1977), pp. 58-60.

[D2] ———, *An hierarchy of inequalities*, Stud. Appl. Math., 63 (1980), pp. 263-274.

[DDP] D. E. DAYKIN, J. W. DAYKIN AND M. S. PATERSON, *On log concavity for order-preserving maps of partial orders*, Discrete Math., to appear.

[G1] R. L. GRAHAM, *Linear extensions of partial orders and the* FKG *inequality*, in Ordered Sets, I. Rival ed., D. Reidel, Dordrecht, 1982, pp. 213-236.

[G2] ———, *Applications of the* FKG *inequality and its relatives*, to appear.

[GYY] R. L. GRAHAM, A. C. YAO AND F. F. YAO, *Some monotonicity properties of partial orders*, this Journal, 1 (1980), pp. 251-258.

[J] J. W. DAYKIN, Ph.D. thesis, Dept. Computer Science, Warwick University, England, (in preparation).

[KS] D. J. KLEITMAN AND J. B. SHEARER, *A monotonicity property of partial orders*, Stud. Appl. Math., 65 (1981), pp. 81-83.

[M] L. MIRSKY, *A dual of Dilworth's decomposition theorem*, Amer. Math. Monthly, 78 (1971), pp. 876-877.

[S1] L. A. SHEPP, *The* FKG *inequality and some monotonicity properties of partial orders*, this Journal, 1 (1980), pp. 295-299.

[S2] ———, *The XYZ conjecture and the* FKG *inequality*, Ann. Probab. 10 (1982), pp. 824-827.

[W] P. M. WINKLER, *Correlation among partial orders*, this Journal, 4 (1983), pp. 1-7.

SIAM J. ALG. DISC. METH.
Vol. 6, No. 4, October 1985

021

DYNAMICAL BEHAVIOUR OF NEURAL NETWORKS*

E. GOLES CH.†

Abstract. We characterize the cyclic behaviour of a threshold neural network defined by an iteration with memory. Our study is based on an algebraic approach that consists in defining an invariant associated with the dynamic of the network.

1. Introduction. A mathematical model of neural networks proposed by E. R. Caianello [1], [2] is represented by a set of equations

$$y_i(t) = 1\left[\sum_{j=1}^{n}\sum_{s=1}^{k} a_{ij}(s)y_j(t-s) - \theta_i\right] \quad \text{for } i = 1, \cdots, n,$$

where $y_j(*) \in \{0, 1\}$ for $j = 1, \cdots, n$ and

$$1[u] = \begin{cases} 0 & \text{if } u < 0, \\ 1 & \text{otherwise.} \end{cases}$$

The dynamic of this model has been studied in some particular cases.

- For a single neuronic equation, i.e.

$$y(t) = 1\left[\sum_{s=1}^{k} a(s)y(t-s) - \theta\right],$$

T. Kitagawa [4], Nagami, Kitahashi, Tanaka, Poljak [6], [7] have characterized the dynamical behaviour of the system (convergence to stable configurations, reverberation cycles, etc.) with a hypothesis on the coupling coefficients, $a(s)$. The principal results are the following:

- If the coupling coefficients are all nonnegative, any initial configuration converges to a stable configuration (the fixed points $\mathbf{0}$ or $\mathbf{1}$) [6].
- If the coupling coefficients are all nonpositive, there exist reverberation cycles when $\sum_{s=1}^{k} a(s) < \theta < 0$. If the $a(s)$ are nonpositive and identical, $a(s) = a < 0$, the cycle length, T, is a divisor of $k+1$ [6].
- For a neural equation that evolves in parallel, i.e., $k = 1$ and

$$v_i(t) = 1\left[\sum_{j=1}^{n} a_{ij}y_j(t-1) - \theta_i\right], \qquad i = 1, \cdots, n,$$

we have proved in the symmetrical case, i.e., $A = (a_{ij})$ symmetric, that the cycle length, T, is one or two [3].

- If A is nonsymmetric, the associated dynamic is very complicated because this class of networks is a McCulloch-Pitts net and simulates any finite automaton [5].

In this paper we study some aspects of the dynamical behaviour of neural networks with nontrivial memory ($k > 1$). Our analysis will be based on an algebraic discrete invariant associated with the network.

2. Nonuniform coupling coefficients. Let us consider the set of neural equations

$$y_i(t) = 1\left[\sum_{j=1}^{n}\sum_{s=1}^{k} a_{ij}(s)y_j(t-s) - \theta_i\right] \quad \text{for } i = 1, \cdots, n, \quad y_j(*) \in \{0, 1\}, \quad a_{ij}(*), \theta_i \in \mathbb{R}$$

* Received by the editors July 6, 1983, and in revised form October 24, 1984.

† Departamento de Matemáticas, Esc. de Ingeniería, Universidad de Chile, Santiago, Chile, and IMAG, Centre National de la Recherche Scientifique, 38402 St. Martin d'Hères Cedex, France.

with the hypothesis

$$a_{ij}(k-s+1)=a_{ji}(s) \quad \text{for } s=1,\cdots,k \quad \text{and } i,j=1,\cdots,n.$$

We will prove that, in such a network, the cycle length, T, is a divisor of $k+1$.

Let us denote $y(t)=(y_1(t),\cdots,y_n(t))$. Since $y(t)\in\{0,1\}^n$, it is obvious that for any initial configuration $y(0),\cdots,y(k-1)$ the iteration on the neural network is ultimately periodic; that is to say, there exist r, $T\in\mathbb{N}$ and vectors $y(r),\cdots,y(r+T-1)$ such that

$$y_i(m)=1\left[\sum_{j=1}^{n}\sum_{s=1}^{k}a_{ij}(s)y_j(m-s)-\theta_i\right] \quad \text{for } m=r,\cdots,r+T-1 \text{ and } i=1,\cdots,n$$

where the indexes are taken modulo T.

We can then associate with any cell i of the network the "cycle-vector"

$$x_i=(x_i(0),\cdots,x_i(T-1))=(y_i(r),\cdots,y_i(r+T-1)).$$

Algebraic invariant. Let E be the operator

$$E(x_i,x_j)=\sum_{t=0}^{T-1}x_i(t)\left\{\sum_{s=1}^{k}a_{ij}(k-s+1)x_j(t+s)-\sum_{s=1}^{k}a_{ij}(s)x_j(t-s)\right\} \quad \text{for } i,j=1,\cdots,n,$$

where the indexes are taken modulo T.

Properties of E.

1. $E(x_i,x_j)+E(x_j,x_i)=0$ *for* $i,j=1,\cdots,n$.

Proof. We have

$$E(x_i,x_j)=-\sum_{t=0}^{T-1}x_j(t)\left\{\sum_{s=1}^{k}a_{ij}(s)x_i(t+s)-\sum_{s=1}^{k}a_{ij}(k-s+1)x_i(t-s)\right\}.$$

Since $a_{ij}(s)=a_{ji}(k-s+1)$, $a_{ij}(k-s+1)=a_{ji}(s)$; hence $E(x_i,x_j)=-E(x_j,x_i)$. □

2. *Let* γ_i *the period of the vector* x_i. *Then, if* $y_i \prec k+1$, *then*:

$$E(x_i,x_j)=0 \quad \textit{for any } j=1,\cdots,n.$$

Proof. Since T is the period of the network, we have $T=\beta\gamma_i$ and by hypothesis $k+1=\theta\gamma_i$.

Let C_m be the set

$$C_m=\{m,\ m+\gamma_i,\ m+2\gamma_i,\cdots,\ m+(\beta-1)\gamma_i\}, \qquad m=0,\cdots,\gamma_i-1.$$

Since $T=\beta\gamma_i$, we have

$$E(x_i,x_j)=\sum_{m=0}^{\gamma_i-1}\sum_{t\in C_m}x_i(t)\left\{\sum_{s=1}^{k}a_{ij}(k-s+1)x_j(t+s)-\sum_{s=1}^{k}a_{ij}(s)x_j(t-s)\right\}.$$

Let $M(i,j,m)$ be

$$M(i,j,m)=\sum_{t\in C_m}x_i(t)\left\{\sum_{s=1}^{k}a_{ij}(k-s+1)x_j(t+s)-\sum_{s=1}^{k}a_{ij}(s)x_j(t-s)\right\}.$$

Since γ_i is the period of vector x_i

$$x_i(t)=x_i(t+q\gamma_i) \quad \text{for any } q\in\mathbb{N};$$

hence

$$M(i,j,m)=x_i(m)\sum_{s=1}^{k}a_{ij}(s)\sum_{t\in C_m}\{x_j(t+k+1-s)-x_j(t-s)\}.$$

Since $k+1=\theta\gamma_i$

$$M(i,j,m)=x_i(m)\sum_{s=1}^{k}a_{ij}(s)\left\{\sum_{t=0}^{\beta-1}x_j(m+(t+\theta)\gamma_i-s)-\sum_{t=0}^{\beta-1}x_j(m+t\gamma_i-s)\right\}.$$

Since T is the period of the network and $T=\beta\gamma_i$, it is easy to see that

$$\sum_{t=0}^{\beta-1}x_j(m+(t+\theta)\gamma_i-s)=\sum_{t=0}^{\beta-1}x_j(m+t\gamma_i-s);$$

hence $M(i,j,m)=0$. Then $E(x_i,x_j)=\sum_{m=0}^{\gamma_i-1}M(i,j,m)=0$. □

3. *If* $\gamma_i \not< k+1$, *then* $\sum_{j=1}^{n}E(x_i,x_j)<0$.

Before proving this property let us denote

$$\sup(x_i)=\{t\in\{0,\cdots,T-1\}/x_i(t)=1\},$$

and let $\{C_i^m\}_{m=0}^{p}$ be the partition of $\sup(x_i)$ defined as follows:

Let $C_i^0=\{t/\{t+(k+1),t+2(k+1),\cdots,t-(k+1)\}\subseteq\sup(x_i)$. In particular $C_i^0=\varnothing$, if $(k+1)\not< T$.

Let $C_i^1,\cdots,C_i^p$ the maximal subsets of $\sup(x_i)\backslash C_i^0$ having the form

$$C_i^m=\{t_m,t_m+(k+1),\cdots,t_m+q_m(k+1)\}$$

for $m=1,\cdots,p$.

Example. $k=3$ with $T=\gamma_i=16$.

t	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
x_i	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	0

$$C_i^0=\{1,2,5,6,9,10,13,14\},\qquad C_i^1=\{3,7,11\}.$$

Proof of property 3. From the partition introduced above we have:

$$\sum_{j=1}^{n}E(x_i,x_j)=\sum_{j=1}^{n}\sum_{m=0}^{p}\sum_{t\in C_i^m}x_i(t)\left\{\sum_{s=1}^{k}a_{ij}(k-s+1)x_j(t+s)-\sum_{s=1}^{k}a_{ij}(s)x_j(t-s)\right\}.$$

Let

$$R(i,j,m)=\sum_{t\in C_i^m}x_i(t)\left\{\sum_{s=1}^{k}a_{ij}(k-s+1)x_j(t+s)-\sum_{s=1}^{k}a_{ij}(s)x_j(t-s)\right\}.$$

Clearly

$$R(i,j,m)=\sum_{s=1}^{k}a_{ij}(s)\sum_{t\in C_i^m}(x_j(t+k-s+1)-x_j(t-s)).$$

From definition of C_i^0, it is easy to see that $R(i,j,0)=0$. Since $\gamma_i\not< k+1$, we have at least $C_i^1\neq\varnothing$; hence

$$R(i,j,m)=\sum_{s=1}^{k}a_{ij}(s)\sum_{t=0}^{q_m}(x_j(t_m+(k+1)t+k+1-s)-x_j(t_m+(k+1)t-s)).$$

Then

$$R(i,j,m)=\sum_{s=1}^{k}a_{ij}(s)(x_j(t_m+(k+1)(q_m+1)-s)-x_j(t_m-s));$$

hence

$$\sum_{j=1}^{n} R(i,j,m) = \sum_{j=1}^{n}\sum_{s=1}^{k} a_{ij}(s)x_j(t_m+(k+1)(q_m+1)-s) - \sum_{j=1}^{n}\sum_{s=1}^{k} a_{ij}(s)x_j(t_m-s).$$

Since $x_i(t_m+(k+1)(q_m+1))=0$, $x_i(t_m)=1$ and by definition of the neuronic equation

$$\sum_{j=1}^{n}\sum_{s=1}^{k} a_{ij}(s)x_j(t_m+(k+1)(q_m+1)-s) < \theta_i,$$

$$\sum_{j=1}^{n}\sum_{s=1}^{k} a_{ij}(s)x_j(t_m-s) \geqq \theta_i,$$

it follows that

$$\sum_{j=1}^{n} R(i,j,m) < 0,$$

hence

$$\sum_{j=1}^{n} E(x_i,x_j) = \sum_{j=1}^{n}\sum_{m=0}^{p} R(i,j,m) < 0. \qquad \square$$

The properties above can be summarized as follows:

THEOREM 1. *If we have*:

$$a_{ij}(k-s+1) = a_{ji}(s) \quad \textit{for } i,j=1,\cdots,n \textit{ and } s=1,\cdots,k$$

then, for any initial configuration $y(0),\cdots,y(k-1)\in\{0,1\}^n$ *the neural network converges towards a cycle of length* T, *such that* $T \mid k+1$.

Proof. If T is not a divisor of $k+1$, there exists at least one index $i\in\{1,\cdots,n\}$ such that $\gamma_i \nmid k+1$; hence from properties 2 and 3 we conclude

$$\sum_{i=1}^{n}\sum_{j=1}^{n} E(x_i,x_j) < 0,$$

and from property 1

$$\sum_{i=1}^{n}\sum_{j=1}^{n} E(x_i,x_j) = 0,$$

which is a contradiction. Therefore $T \mid k+1$. $\square$

Comments. If $k=1$ and $A=(a_{ij})$ is a symmetric matrix, we have the two-cycle behaviour studied in the context of nonuniform cellular automata [3].

As a particular case of the preceding result we can consider the case of uniform coupling coefficients with a nonconnected memory structure, i.e., a set of neural equations

$$y_i(t) = 1\left[\sum_{j=1}^{n} a_{ij}\sum_{s=1}^{k} x_j(t-p_s) - \theta_i\right],$$

where (a_{ij}) is a symmetric matrix and the memory steps $p_1>p_2>\cdots>p_k\geqq 1$ verify

(H)
$$\begin{aligned} &p_1+p_{2r} = p_2+p_{2r-1} = \cdots = p_r+p_{r+1} && \text{if } k=2r,\\ &p_1+p_{2r-1} = p_2+p_{2r-2} = \cdots = p_{r-1}+p_{r+1} = 2p_r && \text{if } k=2r-1. \end{aligned}$$

It is easy to see that this system is equivalent to the following:

$$y_i(t) = 1\left[\sum_{j=1}^{n}\sum_{s=1}^{q} a_{ij}(s)x_j(t-s) - \theta_i\right]$$

where

$$a_{ij}(s) = \begin{cases} a_{ij} & \text{if } s \in \{p_1, \cdots, p_k\}, \\ 0 & \text{otherwise,} \end{cases}$$

and $q = p_1 + p_k - 1$.

Furthermore, from hypothesis (H) and A being a symmetric matrix, we have:

$$a_{ij}(q - s + 1) = a_{ji}(s) \quad \text{for } s = 1, \cdots, q.$$

Then we have the corollary:

COROLLARY. *If hypothesis* (H) *holds, then for any initial configuration* $y(0), \cdots, y(p_1 - 1) \in \{0, 1\}^n$, *the neural network converges towards a cycle of length* T, *such that* $T \mid p_1 + p_k$.

Proof. Directly from the previous comment. □

Remark. We can also prove this corollary by introducing a particular algebraic invariant. With the notation above, let us define

$$E(x_i, x_j) = a_{ij} \sum_{t=0}^{T-1} x_i(t) \left(\sum_{s=1}^{k} x_j(t + p_s) - \sum_{s=1}^{k} x_j(t - p_s) \right) \quad \text{for } i, j = 1, \cdots, n.$$

From hypothesis (H) and the symmetrical property of A it is easy to prove that E verifies properties 1, 2, 3; hence $T \mid p_1 + p_k$. □

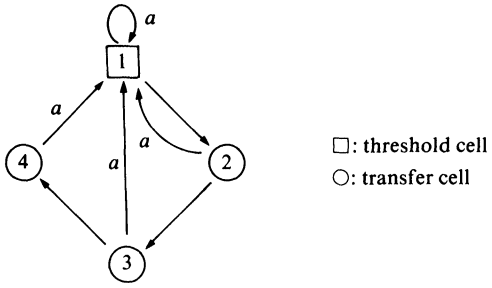
3. Some remarks on a single neuronic equation. In the particular case of a single neuronic equation, i.e.,

$$y(t) = 1\left[\sum_{s=1}^{k} a(s) y(t - s) - \theta \right],$$

we generalize to nonconnected memory structure the result of Nagami [6], obtaining it as a particular case of Theorem 1; i.e.

If $a(s) = a \neq 0$ *for* $s = 1, \cdots, k$, *then* $T \mid (k + 1)$.

Furthermore, we can interpret the single neuronic equation with memory of length k as a directed graph with one threshold cell and $k - 1$ transfer cells which only transmit their binary input. For instance, if we have $k = 4$ and $a(s) = a$ for $s = 1, 2, 3, 4$, the associated graph is:



where

$$y_1(t) = 1\left[a \sum_{j=1}^{4} y_j(t - 1) - \theta \right],$$

$$y_j(t) = y_{j-1}(t - 1), \qquad j = 2, 3, 4.$$

The preceding remarks show that our analysis can be considered as a first step towards the study of the dynamical behaviour of nonsymmetric threshold networks, i.e., directed graphs where each node is a threshold cell.

A first problem to study should be the cyclic behaviour of arbitrary directed networks of order n consisting of one threshold cell and $n-1$ transfer cells.

Finally, the approach introduced here seems to be the appropriate tool for this study because it takes into account both the structure of the network and the iteration scheme in an algebraic fashion.

REFERENCES

[1] E. R. CAIANELLO, *Decision equations and reverberations*, Kybernetik, 3 (1966), pp. 33-40.

[2] E. R. CAIANELLO, DE LUCA AND L. RICCIARDI, *Reverberations and control of neural networks*, Kybernetik, 4 (1967), pp. 10-18.

[3] E. GOLES AND J. OLIVOS, *Comportement périodique des fonctions à seuil binaires et applications*, Disc. Appl. Math., 3 (1981), pp. 93-105.

[4] T. KITAGAWA, *Dynamical systems and operators associated with a single neuronic equation*, Math. Biosci., 18 (1973), pp. 27-71.

[5] W. MCCULLOCH AND W. PITTS, *A logical calculus of the ideas immanent in nervous activity*, Bull. Math. Biophys., 5 (1943), pp. 115-133.

[6] H. NAGAMI, T. KITAHASHI AND K. TANAKA, *Characterization of dynamical behavior associated with a single neural equation*, Math. Biosci., 32 (1976), pp. 221-237.

[7] S. POLJAK AND D. SURA, *On periodical behaviour in societies with symmetric influences*, Combinatorica (1983), to appear.