Ten years LNMB

Ph.D. research and graduate courses of
the Dutch Network of Operations Research

W.K. Klein Haneveld
O.J. Vrieze
L.C.M. Kallenberg
(editors)

1991 Mathematics Subject Classification: 90-XX (Operations Research)
ISBN 90 6196 475 X
NUGI-code: 811

Printed in the Netherlands

Preface

The Dutch Network of Operations Research (in Dutch LNMB) is an interuniversity organization, in which all universities in the Netherlands and the Centre for Mathematics and Computer Science at Amsterdam (CWI) participate. It was founded on July 1, 1987. The LNMB is a national graduate network. The PhD students in OR are expected to participate in the education program, which consists of courses, colloquia and workshops. An other function of the LNMB is to be a forum of all scientific OR-workers. There is a board in which each university is represented.

During the November meeting of the board, we discussed what to do with the tenth anniversary of our network. One of the suggestions was to compose a book with contributions of those, who were involved in the LNMB, particularly the former PhD-students and some of the members of the LNMB. The major objective was to present the research done by young OR researchers in the Netherlands. We are very glad that many of them responded enthusiastically.

The book contains over forty contributions and covers a broad spectrum of Operations Research. After an introduction on the origin of the network, written by the first director Wim Klein Haneveld, the first part is devoted to Combinatorial Optimization and Discrete Mathematics, with a review paper by Karen Aardal, Stan van Hoesel, Jan Karel Lenstra and Leen Stougie. The second part, on Stochastic Operations Research, starts with a retrospective view, composed by Henk Tijms. The third part concerns Game Theory. Stef Tijs and Koos Vrieze wrote a paper on the contribution in game theory of the Netherlands during the past decade. This book covers also some papers on OR-applications. These are collected in part four with a review by Rommert Dekker. In part five of this book some other articles are gathered, e.g. a short survey on semidefinite programming by Etienne de Klerk, Kees Roos and Tamas Terlaky. The book is concluded with a review of the activities of the LNMB during this first decade.

Our editorial tasks were much alleviated by the efforts of Eric Bakker. He did a marvellous job by unifying all kinds of TeX styles and by converting Word, Wordperfect and other files into TeX. We are very grateful to him. We also wish to thank CWI, who published this book and made it available at a reasonable price. Last, but not least, we wish to express our gratitude to the authors of this book.

Lodewijk Kallenberg
Wim Klein Haneveld
Koos Vrieze

May 1997

Table of Contents

On the origin and first decade of the LNMB 1
W. Klein Haneveld

Part I: Combinatorial Optimization and Discrete Mathematics 3

A Decade of Combinatorial Optimization 5
K. Aardal, S. van Hoesel, J.K. Lenstra and L.Stougie

LP-based solution methods for single-machine scheduling problems 15
J.M. van den Akker

Graphs, distances and eigenvalues 19
E.R. van Dam

Algorithms for graphs of small treewidth 23
B. de Fluiter

Planning shifts in a 36-hour cyclical roster 31
R. Freling, N. Piersma and A.P.M. Wagelmans

On the graph parameters of Colin de Verdière 37
H. van der Holst

Scheduling with target start times 45
J.A. Hoogeveen and S.L. van de Velde

A facet preserving extension of the travelling salesman polytope 55
J. de Kort and T. Volgenant

A polyhedral approach to grouping problems 63
M. Oosten

Determining Haemer's rank bound is NP-hard 69
R. Peeters

$1, \frac{3}{2}, \frac{13}{7}$ 77
F. Spieksma

Part II: Stochastic Operations Research 85

Stochastic Operations Research: A Retrospective View 87
H.C. Tijms

Performance modelling and analysis of an ATM-DQDB interworking unit 93
S. Aalto, J.L. van den Berg, E. Smeitink and J.T. Virtamo

Explicit solutions for queueing problems 105
I. Adan

On the transition matrix of Markov chains obtained via cyclic mappings 115
R.J. Boucherie

Queueing systems with periodic service 121
M.J.A. van Eenige

A general approach to computing loss probabilities in finite-buffer queues 131
F.N. Gouweleeuw

Analysis of a customer assignment model with decentralized control 135
A. Hordijk and A. Loeve

Analysis of a generalized shortest queue system by flexible bound models 151
G.J. van Houtum

Stochastic scheduling with event-based dynamic programming 161
G. Koole

Performance of queues with 'worst case input' 171
M.R.H. Mandjes

Periodic polling systems in heavy traffic 179
R.D. van der Mei

Heuristics for complex inventory systems 191
E. van der Sluis

Maintenance policies for complex systems 199
S.G. Vanneste

Part III: Game Theory 205

Game theory of the past decade, the contribution of the Netherlands 207
S.H. Tijs and O.J. Vrieze

On 'dynamics, equilibria, and values' 217
R. Joosten

On consistency of reward allocation rules in sequencing situations 223
J. Suijs, H. Hamers and S. Tijs

A survey on optimality and equilibria in stochastic games 233
F. Thuijsman

Stable sets in non-cooperative game theory 243
D. Vermeulen

Part IV: Applications 251

A review of applications of operations research in the Netherlands 253
R. Dekker

Integration of operational research and environmental management 261
J.M. Bloemhof-Ruwaart

Robust investments 269
D.-F. Broens

Robotic flowshop scheduling is strongly NP-complete 277
Y. Crama and J. van de Klundert

Asset liability management for pension funds 287
C.L. Dert

Minimizing leadtimes by optimizing batch sizes 301
P. Tielemans

Performance analysis of repairable systems 309
P. Wartenhorst

Part V: Other contributions 315

OR-consultants at CQM: make optimization work! 317
D. den Hertog and B. Jansen

A short survey on semidefinite programming 323
E. de Klerk, C. Roos and T. Terlaky

Global optimization: from pure adaptive search to simulated annealing 341
H.E. Romeijn

Generalized job shop Scheduling: complexity and local search 351
R.J.M. Vaessens

Convex approximations for stochastic programs with simple integer recourse 357
M.H. van der Vlerk

Part VI: Some facts 367

Review of the education activities of the LNMB 369
L.C.M. Kallenberg

On the origin and first decade of the LNMB

Wim Klein Haneveld
Chairman LNMB
E-mail: w.k.klein.haneveld@eco.rug.nl

Some day in 1988 I got up early, to find myself at seven o'clock in the morning in the Oude Boteringestraat, downtown Groningen. There I joined the rector magnificus and other officials of the University of Groningen. A bus took us to the Ministry of Education in Zoetermeer, where university officials from all over the Netherlands were invited to be informed about the minister's decision on the financing of a number of so-called 'aio-networks'. I was a little nervous, because one of the proposals was the LNMB program, the National Network of Mathematics of Operations Research in the Netherlands. Every now and then some delegates left, and others came in the conference room. When the LNMB was discussed, the waiting room became almost empty, because all universities participated in this initiative. Ten minutes later congratulations were collected, since the minister granted the complete starting subsidy we asked for! This happy day marked the end of a sometimes chaotic period of creating a new way of cooperation in the Dutch OR-community.

What is this all about? As is well-known, the discipline 'Operations Research' owes its name to the successful application of mathematical models for solving logistic planning of military operations in World War II. After the war, complex planning problems in industry, government, and agriculture were analyzed with mathematical models, too, often under the name 'Management Science'. By using the possibilities of the computer, already at an early stage, OR developed quickly. Countless models and methods for project planning, production planning and scheduling, maintenance, replacement, allocation, routing and transportation, physical distribution, inventory, investments, telecommunication, and congestion are the result. Since the sixties, all universities in the Netherlands including the Technical Universities and the Agricultural University founded chairs in OR, under a variety of names. OR is typically interdisciplinary, as can be seen from the fact, that it is included in curricula of mathematics, econometrics, engineering, and business economics, and its contribution ranges from mathematical abstraction to practical application.

In the early seventies, several OR groups existed in the Netherlands. One of the first people, who realized that more was needed, and who did something about it, was Gijs de Leve. Together with Jan Karel Lenstra he made a traveling salesman tour through the country. As a result we have, each year from 1976 on, an 'International Conference on the Mathematics of Operations Research', organized by the Centrum voor Wiskunde en Informatica. Foreign distinguished speakers report on promising research areas, and in due time, the Lunteren environment inspired the OR-people in the Netherlands to become an informal but lively OR-community.

In the early eighties the Dutch government introduced the so-called aio-system: Ph.D.-students were given a new name, 'assistent-in-opleiding' (aio) or 'onderzoeker-in-opleiding'

(oio), and a new status. In order to promote the development of corresponding graduate programs, some possibilities were created for initial funding. The existence of an OR-community appeared to be very instrumental. Although on every sub-area of OR, as stochastic OR, combinatorial optimization, mathematical programming, and game theory the research level in the Netherlands was internationally competitive, every separate group was rather small. However, all over the country we were large enough for a common graduate program. And that was precisely what was aimed at in the spring of 1987, resulting in the formal start of the LNMB, July 1987.

In the past ten years it appeared that the program is very successful. All Ph.D.-students in OR participate in common courses in Utrecht, and in workshops all over the country. They have the opportunity to learn from expert researchers on all important sub-areas. Of course, the thesis is extremely important. Nevertheless, the LNMB provides the young scientists an orientation on actual developments in the whole of the discipline, and this feature can hardly be underestimated. Moreover, one of the side effects is, that all over the country beginning OR-researchers know each other, and so the juniors are not working on isolated islands any more, but they are part of the OR-community. It is characteristic in this respect, that the international workshops of the LNMB have been integrated in the Lunteren conferences.

Up to now, the LNMB organized 38 courses, 16 workshops, 175 colloquia for 231 aio's/ oio's, and more than 80 dissertations have been finished, and many more are coming. In this book you will find overviews of the development of several areas in mathematics of OR, together with a number of research topics dealt with by alumni of the network. As can be seen, a wide range of subjects and approaches are presented. The alumni graduated at mathematical faculties, both at general universities and technical universities, and at economic faculties. In spite of the broad background they share a high level of modeling and mathematical expertise.

Foreign guests, giving a colloquium in Utrecht, or presenting a lecture in Lunteren, are amazed, and often a little jealous of the LNMB. Indeed, we have the advantage to be a small country in terms of traveling time. On the other hand, many counterexamples show, that it is far from obvious that individual research groups are willing to cooperate, even if their physical distance is small. I am proud that the OR-community in the Netherlands is able to deal with its common interests. This does not mean that no effort is needed to realize the goals. One needs people who feel it as a personal responsibility to make things work. In the early days Jan Karel Lenstra acted as an inspiring and efficient chairman. Much work was done by Suwarni Bambang Oetomo. She not only had to deal with rapidly changing workloads, but she also had to solve many practical problems, related to an organization in statu nascendi. After a period of improvisation, the network matured under the leadership of Lodewijk Kallenberg, assisted by Gonnie Ooms-Woudstra.

Today, the LNMB is an established organization. Beginning aio's usually do not realize that it ever started. Together with the Netherlands Society of OR (NGB) it is recognized as the national forum on OR. I hope and expect that this added value will continue in the future.

Part I

Combinatorial Optimization
and Discrete Mathematics

A Decade of Combinatorial Optimization

Karen Aardal
Utrecht University
E-mail: aardal@cs.ruu.nl

Stan van Hoesel
University of Maastricht
E-mail: s.vanhoesel@ke.unimaas.nl

Jan Karel Lenstra
Eindhoven University of Technology; CWI, Amsterdam
E-mail: jkl@win.tue.nl

Leen Stougie
University of Amsterdam
E-mail: leen@fee.uva.nl

Abstract

This paper offers a brief overview of the developments in combinatorial optimization during the past decade. We discuss improvements in polynomial-time algorithms for problems on graphs and networks, and review the methodological and computational progress in linear and integer optimization. Some of the more prominent software packages in these areas are mentioned. With respect to obtaining approximate solutions to NP-hard problems, we survey recent positive and negative results on polynomial-time approximability and summarize the advances in local search.

1 Introduction

Combinatorial optimization is involved with models and methods for optimization over discrete choices. It is rooted in the theory of linear programming, and has strong links with discrete mathematics, probability theory, algorithmic computer science, and complexity theory. Some problems in the area are relatively well understood and admit solution to optimality in polynomial time. Many others are NP-hard, and one is forced to go one of three ways. Either one chooses an enumerative method that is guaranteed to produce an optimal solution. Or one applies an approximation algorithm that runs in polynomial time. Or one resorts to some type of heuristic search technique, without any a priori guarantee in terms of solution quality or running time.

In the past decade we have seen significant progress on all these fronts. Network flow algorithms became more efficient, and so did algorithms for linear and convex optimization. For the hard problems, advances in polyhedral techniques extended the realm of true optimization methods. Performance bounds that can – or probably cannot – be met in polynomial time were tightened. And there has been a surge in the development of local

search approaches. Remarkable aspects were the use of randomization in the design and analysis of algorithms, and the attention paid to on-line planning models.

Two developments outside the area stimulated research in combinatorial optimization. First, the continued increase in computing power strengthened the need for efficient algorithms. The ability to handle bigger problems made the distinction between low and high order running times more pronounced and diminished the power of brute force. Second, at the application side, there has been an increasing confidence in the practical potential of optimization techniques. Large and difficult real-world problems that were out of reach ten years ago are now being solved. Notable examples occurred in airline crew scheduling, train timetabling, time-constrained vehicle routing, telecommunication network design, frequency allocation, VLSI layout synthesis, and statistical disclosure control.

Combinatorial optimization has established itself as a mature discipline of scientific interest and practical relevance. The selection of topics and references presented below has been governed by space constraints and personal bias. We apologize for all omissions.

2 Graphs and networks

Major improvements in running times of algorithms for specific graph related problems have been obtained during the last ten years. We highlight a few here. Quite a number of these results have been obtained via randomized algorithms, for which we refer to the book by Motwani and Raghavan [39].

Much research on designing faster algorithms for the maximum flow problem and the minimum cost flow problem was initiated by the work of Tardos [49], who found the first strongly polynomial algorithm for the minimum cost flow problem. Scaling of the input parameters and prefixing flows are the main ingredients of most of these new algorithms, but the design of efficient data structures has also had an important impact. For a network with n nodes and m arcs, the best known strongly polynomial algorithms for finding a maximum flow and a minimum cost flow have running times $O(nm \log n)$ and $O(m \log n(m + n \log n))$, respectively. A thorough treatment of these results is given by Ahuja et al. [6]. This book also describes improvements in solution times for problems such as shortest path and matching problems.

The minimum cut problem of finding a minimum weight set of arcs in a network whose removal would disconnect the network is dual to the maximum flow problem. Recently, new algorithms have been developed for this problem that do not exploit this duality. Nagamochi and Ibaraki [40], for instance, use edge contraction in their algorithm. Randomized edge contraction, introduced by Karger and Stein [32], leads to the fastest algorithm so far. An overview of these algorithms with a computational study is given by Chekuri et al. [17].

Interesting results have been obtained in determining polynomially solvable subclasses of generally NP-hard problems. Robertson and Seymour [45] proved an old conjecture of Wagner: for each set of graphs that is closed under taking minors, there exists a finite set of graphs that are forbidden to be minors of any graph in the set. This *obstruction set* can be enormously large, but its finiteness allows Robertson and Seymour to prove the existence of a polynomial-time algorithm for determining the tree-width of any graph in a class that is known to contain graphs of bounded tree-width only. The proof is non-constructive, and the algorithm may involve a large constant (of the order of the number of forbidden minors). Bodlaender [15] gives a good overview of the techniques involved together with some applications.

Next to these classical graph problems, there are several problems in planning and

scheduling that can be viewed as problems on graphs. One well-studied problem of this kind is the uncapacitated lot sizing problem. In 1958 Wagner and Whitin proposed an $O(T^2)$ dynamic programming algorithm, where T is the number of time periods in the planning horizon. It lasted more than thirty years before a better algorithm was found. In the early 1990's three groups simultaneously developed algorithms with running time $O(T \log T)$; see [5], [24], [52].

Important for the implementation of graph related algorithms is the availability of software packages. The most prominent software library is LEDA, A Library of Efficient Datatypes and Algorithms, developed by Melhorn and Näher [38]. It is implemented by a C++ class library, and incorporates many efficient data structures and algorithms. LEDA is available at `ftp://ftp.mpi-sb.mpg.de` in directory `pub/LEDA`.

3 Linear optimization

The main developments in linear optimization have sprouted from the work of Karmarkar [33], who started a wave of research on so-called *interior point methods*. Both theoretical and practical advances were accomplished over the past ten years, and by now some interior point methods are competitive with the celebrated simplex method. An interesting overview and discussion of the use of simplex and interior point methods can be found in the *ORSA Journal on Computing 6.*1 (1994). The book by Roos et al. [46] gives a comprehensive treatment of interior point methods for linear optimization. Interior point methods have also been developed for convex optimization problems. The application of interior point type methods to semidefinite optimization has led to results that have proved particularly useful in the design of approximation algorithms for certain combinatorial optimization problems; see Section 5.

A new line of research is the development of randomized algorithms for the search of an optimal basic feasible solution. The main open question here is if there exist randomized algorithms that solve linear optimization problems in strongly polynomial expected running time. Though this question has not been resolved yet, major steps have been taken. The fastest randomized algorithm is due to Kalai [31], and has expected running time $O(n^2m + b^{\sqrt{n \log n}} \log m)$, where n is the number of variables, m the number of constraints, and b a constant independent of the input. It is in essence a randomized simplex algorithm. For a review of research in this direction we refer to Chapter 9 of the book by Motwani and Raghavan [39].

With respect to deterministic simplex algorithms, many improvements in practical performance have been achieved. Many of these improvements have been implemented in the state-of-the-art software package CPLEX [21]. CPLEX also contains an interior point method.

To enhance user-friendliness of software for linear and integer optimization, modeling languages that allow for representation of variables and constraints in a set-based format are very useful. Leading computer packages for modeling are AMPL [25] and AIMMS [14].

4 Integer optimization

The most commonly used technique for solving (mixed) integer programs is still branch-and-bound. The quality of the available upper and lower bounds on the optimal value of

the considered instance is the decisive factor for success of this tree search technique. A lower bound on the optimal value (assuming a minimization problem) is obtained from a relaxation of the integer program. In the past ten to fifteen years attention has shifted from Lagrangian relaxation to linear programming relaxation, since the latter type of relaxation can be strengthened more easily by using cutting planes. Combining cutting planes and Lagrangian relaxation usually causes convergence problems. Moreover, good LP solvers, such as CPLEX, that allow for addition of rows are nowadays available.

The theory of cutting planes in the form of valid inequalities that define facets of the convex hull of feasible solutions to an instance, was mainly developed prior to this past decade. During the past ten years, however, an enormous amount of more problem specific results have been obtained. Moreover, surprisingly large instances have been solved using a mixture of cutting plane algorithms and branch-and-bound. For recent surveys we refer to Aardal and Van Hoesel [1], [2], and to Chapter 3 of [22]. Similar developments have been attained for column generation methods, which can be viewed as dual to cutting plane techniques. For a survey we refer to Barnhart et al. [11].

A new development of the last decade is the theoretical quality analysis of cutting planes. Negative results for some classes of cutting planes have been reported by Goemans [26]. He evaluated the worst-case improvement resulting from adding several of the known classes of facets for the traveling salesman polytope to the subtour polyhedron, i.e., the set of vectors satisfying the so-called subtour elimination constraints.

Another surprising theoretical result in polyhedral combinatorics is due to Lovász and Schrijver [37], who developed an algorithm for obtaining a sequence of tighter and tighter relaxations of integer 0-1 programs. The algorithm iterates the following steps. First, each constraint of the considered problem is multiplied by each variable x_j and its complement $1 - x_j$ $(j = 1, \ldots, n)$. The resulting quadratic program is then linearized by replacing the nonlinear terms $x_i x_j$ by new variables y_{ij}. This linear formulation is finally projected onto the space of the original variables. Lovász and Schrijver showed that this procedure needs to be repeated at most n times before the convex hull of feasible solutions is obtained. Balas et al. [10] showed that it is sufficient to multiply each constraint by a single variable x_j and its complement at a time.

For branch-and-bound algorithms powerful and quite flexible software packages have been developed. We mention MINTO [47] and ABACUS [50]. MINTO contains more tools such as preprocessing and generic valid inequalities, whereas ABACUS has the advantage that it is written in C++.

Apart from the further development of existing solution techniques, also two new techniques for integer optimization received much attention in the last decade. The first algorithm we mention, developed by H.W. Lenstra [35], is older than ten years, but served as an inspiration for further developments. Lenstra's algorithm was developed to show that the problem of determining whether the polyhedron $K = \{x \in \mathbb{R}^n : Ax \leq b\}$ contains an integer vector x, can be solved in polynomial time if n is fixed. First, the algorithm finds a transformation τ such that the polyhedron τK has a "spherical" appearance. If the basis of the lattice $\tau\mathbb{Z}^n$ has short and near-orthogonal vectors, then the membership problem can be solved recursively by branching on a number of parallel hyperplanes. The number of such hyperplanes can be proved to be bounded by a constant depending only on n. For any lattice such a basis exists and can be found in polynomial time starting from an arbitrary basis by using *basis reduction*; see Lenstra et al. [34]. Lovász and Scarf [36] designed a "generalized" basis reduction algorithm, which works directly on the polyhedron instead of using approximations such as Lenstra does. The advantage of their method is that less

information is lost, the disadvantage is that it uses considerably more computational steps. Cook et al. [20] implemented the Lovász-Scarf algorithm and solved some previously unsolved integer programming problems. Barvinok [12] generalized Lenstra's result and proved that the number of integral points in a polyhedron can be *counted* in polynomial time if the dimension is fixed.

Another new technique, based on the theory of *Gröbner bases*, was already known in computational algebraic geometry, and was introduced for solving integer optimization problems by Conti and Traverso [19]. It amounts to translating the integer programming problem into an algebraic membership problem. The Gröbner bases are used to guide the generalized division that decides the membership. Advances in applicability of these methods are due mainly to Thomas [51]. Their current practical power is restricted by the size of the Gröbner bases, which is large for most problems. Due to their structure such methods have advantages over other more conventional IP methods in solving stochastic integer programming problems; see Schultz et al. [48]. Computer packages for computing Gröbner bases are available, e.g., CoCoa [16] and MACAULAY [13].

5 Polynomial-time approximation

As an alternative to solving NP-hard combinatorial optimization problems to optimality, which may be very time consuming, a stream of research has concentrated on designing polynomial-time algorithms that aim at good approximations for such problems. A widely accepted quality measure of such approximations is the *performance guarantee*, i.e., an upper bound on the ratio between the approximate solution value and the optimal one. A comprehensive and up-to-date survey of the theory of approximation algorithms is provided in the book edited by Hochbaum [28].

Some of the major achievements in this field are based on a combination of relaxation and randomization. Goemans and Williamson (see [27] and Chapter 11 of [28]) designed approximation algorithms that solve appropriately chosen relaxations of mathematical programming formulations of the considered combinatorial problems, and then round the obtained solution in a randomized way. The rounding can be derandomized yielding deterministic approximation algorithms. In particular, Goemans and Williamson use semidefinite optimization relaxations to design algorithms with very good performance guarantees for the problem of finding a maximum cardinality cut in a graph and the problem of finding the maximum number of simultaneously satisfiable clauses in a Boolean expression with at most two literals per clause.

A remarkable result was obtained by Arora [8]. He developed a polynomial-time approximation scheme for the traveling salesman problem (TSP) in the Euclidean space. Here we notice that Christofides' algorithm of 1976 [18], with its performance guarantee of 3/2, is still the best polynomial approximation algorithm for the TSP whose distances are symmetric and satisfy the triangle inequality.

Apart from the above positive sounds on approximation, there has also been a breakthrough on the negative side, in the sense of non-approximability of optimal solutions of some problems. Papadimitriou and Yannakakis [42] defined a class of maximization problems for the purpose of distinguishing problems whose optimal solutions are hard to approximate within arbitrarily small ratio. This class called MAXSNP has a two-sided polynomial reduction defined on it under which it is closed. Given a Boolean expression in conjunctive normal form, the problem MAXSAT of finding a truth assignment to the variables that satisfies the maximum number of clauses is complete for this class.

Arora et al. [9] gave a strong justification for investigating these concepts. They showed that there cannot exist a polynomial-time approximation scheme for MAXSAT unless $P = NP$. The proof is based on an alternative definition of NP in terms of randomized certificate verification based on fingerprinting methodology. This important result implies that for any MAXSNP-complete problem there must be a threshold value strictly greater than 1 on the achievable polynomial-time performance guarantee. For an overview of specific results in this direction, we refer to Chapter 10 of [28].

In sequencing and scheduling, techniques based on linear programming and rounding led to surprising performance guarantees for the off-line and on-line minimization of total (weighted) completion time on a single machine and on parallel machines, and for the minimization of makespan on parallel machines subject to communication delays. An investigation of the complexity of finding very short schedules yielded lower bounds on the polynomial-time approximability of several scheduling problems, including the job shop scheduling problem. For specific results and references, we refer to Chapter 12 of [22].

The previous paragraphs concerned the *worst-case* approach to approximation. A complementary approach is *average-case* or *probabilistic* analysis, a research field that started more than twenty years ago. The main developments in this field during the last decade were based on discovering the possibility to exploit existing results from probability theory. *Empirical process* theory provided tools for the analysis of the optimal solution value of a series of number problems; see Piersma [43]. *Martingale* theory allowed for relatively elegant asymptotic characterizations of optimal solution values of several problems; see Rhee and Talagrand [44]. Finally we mention the rather complete probabilistic analysis of *bin-packing* algorithms, presented in Chapter 2 of [28].

Next to these developments for optimization problems, a breakthrough in approximation was accomplished for *counting* problems, again based on randomization. Counting combinatorial structures such as the number of Hamiltonian cycles in a graph is obviously harder than just deciding on the presence of the structure. Jerrum et al. [30] showed the equivalence between approximate counting and approximate sampling for a wide class of combinatorial structures. Building on work by Aldous [7], they use Markov chains to simulate random (uniform) sampling of the structures, and proved that these "mix rapidly". As a first result Jerrum and Sinclair [29] devised a fully polynomial randomized approximation scheme (FPRAS) for counting perfect matchings in dense graphs, whose vertices have degree at least half of the total number of vertices. The non-dense graph case is still open. Another prominent result in this direction is an FPRAS for computing volumes of convex bodies by Dyer et al. [23]. A series of subsequent papers have given schemes with increasingly better running times. For an overview we refer to Chapter 12 of [28].

6 Local search

For many years heuristic search approaches have been used throughout science and engineering. Their performance was generally considered to be satisfactory, partly based on experience, partly based on a belief in some physical or biological analogy, which was not always supported by familiarity with what has been achieved in mathematics.

Still, in the past decade local search has reinforced its position as a standard approach in combinatorial optimization. Problem size or lack of analytical insight may prohibit the application of true optimization algorithms. Polynomial-time approximation algorithms may give inferior solutions, and their performance bounds, if they can be obtained at all, may be meaningless in practice. Local search is a robust way to obtain good solutions to real

problems in reasonable time.

Simulated annealing has established itself as a relatively straightforward technique that performs very well when given enough time. Tabu search requires more tuning but often less running time. Genetic algorithms are not known to perform well in a pure form, even when problem solutions allow a natural string representation, but hybrid forms in which offspring are subjected to iterative improvement are promising. Neural networks have many applications, which, however, seem to fall outside the realm of optimization.

Many aspects of local search are discussed in the book edited by Aarts and Lenstra [3]. We see three main lines of advance. First, a theory of the computational complexity and performance analysis of local search is now emerging; see Chapters 2 and 3 of [3]. Second, neighborhoods embodying problem-specific knowledge and data structures supporting incremental computations are being used in rather sophisticated implementations. Third, some of the more successful search strategies are hybrids, which combine local search with a constructive method, with tree search or, again, with local search. The *shifting bottleneck procedure* for job shop scheduling of Adams et al. [4] is a constructive rule that reoptimizes partial schedules along the way. The shop scheduling algorithms of Nowicki and Smutnicki (see, e.g., [41] and Chapter 11 of [3]) apply tabu search and jump back to previously considered promising but rejected moves; many other combinations of local search and tree search have been proposed. Johnson's *iterated Lin-Kernighan* algorithm for the TSP (see Chapter 8 of [3]) is a nested form of local search, which applies 4-exchanges to local optima resulting from variable-depth search.

References

[1] K. Aardal, C.P.M. van Hoesel (1996). Polyhedral techniques in combinatorial optimization I: Theory. *Statist. Neerlandica* **50**, 3–26.

[2] K. Aardal, S. van Hoesel (1995). *Polyhedral Techniques in Combinatorial Optimization II: Computations*, Report UU-CS-1995-42, Utrecht University.

[3] E.H.L. Aarts, J.K. Lenstra (eds.) (1997). *Local Search in Combinatorial Optimization*, Wiley, Chichester.

[4] J. Adams, E. Balas, D. Zawack (1988). The shifting bottleneck procedure for job shop scheduling. *Management Sci.* **34**, 391–401.

[5] A. Aggarwal, J.K. Park (1993). Improved algorithms for economic lot size problems. *Oper. Res.* **41**, 549–571.

[6] R.K. Ahuja, T.L. Magnanti, J.B. Orlin (1993). *Network Flows; Theory, Algorithms, and Applications*, Prentice Hall, Englewood Cliffs.

[7] D.J. Aldous (1981). Random walks on finite groups and rapidly mixing Markov chains. *Séminaire de Probabilités XVII*, Lecture Notes in Mathematics 986, Springer, New York, 243–297.

[8] S. Arora (1996). Polynomial time approximation schemes for Euclidean PST and other geometric problems. *Proc. 37th Annual Symp. Foundations of Computer Science*, 2–11.

[9] S. Arora, C. Lund, R. Motwani, M. Sudan, M. Szegedy (1992). Proof verification and hardness of approximation problems. *Proc. 33rd Annual Symp. Foundations of Computer Science*, 14–23.

[10] E. Balas, S. Ceria, G. Cornuéjols (1993). A lift-and-project cutting plane algorithm for mixed 0-1 programs. *Math. Programming* **58**, 295–324.

[11] C. Barnhart, E.L. Johnson, G.L. Nemhauser, M.W.P. Savelsbergh, P.H. Vance (1994). Branch-and-price: column generation for solving huge integer programs. In: J.R. Birge, K.G. Murty (eds.). *Mathematical Programming: State of the Art 1994*, University of Michigan, Ann Arbor, 186–207.

[12] A.I. Barvinok (1994). A polynomial time algorithm for counting integral points. *Math. Oper. Res.* **19**, 769–779.

[13] D. Bayer, M. Stillman. *MACAULAY: A Computer Algebra System for Algebraic Geometry*, Available by anonymous ftp from `zariski.harvard.edu`.

[14] J. Bisschop, R. Entriken (1993). *AIMMS: The Modeling System*, Available at `http://www.paragon.nl/aimms220.html`.

[15] H.L. Bodlaender (1993). A tourist guide through treewidth. *Acta Cybernet.* **11**, 1–21.

[16] A. Capani, G. Niesi (1995). *CoCoa User's Manual, Release 3.0b, Edition 1995*, Department of Mathematics, University of Genova.

[17] C.S. Chekuri, A.V. Goldberg, D.R. Karger, M.S. Levine, C. Stein (1996). *Experimental study of minimum cut algorithms*, Technical report, NEC Research Institute, Princeton.

[18] N. Christofides (1976). *Worst-case analysis of a new heuristic for the travelling salesman problem*, Report 388, GSIA, Carnegie Mellon University.

[19] P. Conti, C. Traverso (1991). Buchberger algorithm and integer programming. *Proceedings of AAECC 9, New Orleans*, Lecture Notes in Computer Science 539, Springer, Berlin, 130–139.

[20] W. Cook, T. Rutherford, H.E. Scarf, D. Shallcross (1993). An implementation of the generalized basis reduction algorithm for integer programming. *ORSA J. Comput.* **5**, 206–212.

[21] CPLEX Optimization, Inc. (1994). *Using the CPLEX callable library, Version 3.0.*

[22] M. Dell'Amico, F. Maffioli, S. Martello (eds.) (1997). *Annotated Bibliographies in Combinatorial Optimization*, Wiley, Chichester.

[23] M.E. Dyer, A.M. Frieze, R. Kannan (1991). A random polynomial algorithm for approximating the volume of convex bodies. *J. Assoc. Comput. Mach.* **38**, 1–17.

[24] A. Federgruen, M. Tzur (1991). A simple forward algorithm to solve general dynamic lot sizing models with n periods in $O(n \log n)$ or $O(n)$ time. *Management Sci.* **37**, 909–925.

[25] R. Fourer, D.M. Gay, B.W. Kernighan (1993). *AMPL: A Modeling Language for Mathematical Programming*, Duxbury Press.

[26] M.X. Goemans (1995). Worst-case comparison of valid inequalities for the TSP. *Math. Programming* **69**, 335–349.

[27] M.X. Goemans, D.P. Williamson (1994). .878-Approximation algorithms for MAX CUT and MAX 2SAT. *Proc. 26th Annual ACM Symp. Theory of Computing*, 422–431.

[28] D.S. Hochbaum (ed.) (1996). *Approximation Algorithms for NP-Hard problems*, PWS Publishing Company, Boston.

[29] M.R. Jerrum, A. Sinclair (1988). Approximating the permanent. *SIAM J. Comput.* **18**, 1149–1178.

[30] M.R. Jerrum, L.G. Valiant, V.V. Vazirani (1986). Random generation of combinatorial structures from a uniform distribution. *Theoret. Comput. Sci.* **43**, 169–188.

[31] G. Kalai (1992). A subexponential randomized simplex algorithm. *Proc. 24th Annual ACM Symp. Theory of Computing*, 475–482.

[32] D.R. Karger, C. Stein (1993). An $O(n^2)$ algorithm for minimum cuts. *Proc. 25th Annual ACM Symp. Theory of Computing*, 757–765.

[33] A.V. Karmarkar (1984). A new polynomial-time algorithm for linear programming. *Combinatorica* **4**, 373–395.

[34] A.K. Lenstra, H.W. Lenstra, Jr., L. Lovász (1982). Factoring polynomials with rational coefficients. *Math. Ann.* **261**, 515–534

[35] H.W. Lenstra, Jr. (1983). Integer programming with a fixed number of variables. *Math. Oper. Res.* **8**, 538–548.

[36] L. Lovász, H.E. Scarf (1992). The generalized basis reduction algorithm. *Math. Oper. Res.* **17**, 751–764.

[37] L. Lovász, A. Schrijver (1991). Cones of matrices and set-functions and 0-1 optimization. *SIAM J. Optimization* **1**, 166–190.

[38] K. Mehlhorn, S. Näher (to appear). *The LEDA Platform of Combinatorial and Geometric Computing*, Cambridge University Press, Cambridge.

[39] R. Motwani, P. Raghavan (1995). *Randomized Algorithms*, Cambridge University Press, New York.

[40] H. Nagamochi, T. Ibaraki (1992). Computing edge-connectivity in multigraphs and capacitated graphs. *SIAM J. Discrete Math.* **5**, 54–66.

[41] E. Nowicki, C. Smutnicki (1996). A fast taboo search algorithm for the job shop problem. *Management Sci.* **42**, 797–813.

[42] C.H. Papadimitriou, M. Yannakakis (1991). Optimization, approximation, and complexity classes. *J. Comput. System Sci.* **28**, 425–440.

[43] N. Piersma (1993). *Combinatorial Optimization and Empirical Processes*, Tinbergen Institute Research Series 52, Amsterdam.

[44] W.T. Rhee, M. Talagrand (1989). Martingale inequalities, interpolation and NP-complete problems. *Math. Oper. Res.* **14**, 91–96.

[45] N. Robertson, P.D. Seymour (1996). Graph minors XV: giant steps. *J. Combin. Theory Ser.* **B 68**, 112–148.

[46] C. Roos, T. Terlaky, J.P. Vial (1997). *Theory and Algorithms for Linear Optimization: An Interior Point Approach*, Wiley, Chichester.

[47] M.W.P. Savelsbergh, G.C. Sigismondi, G.L. Nemhauser (1994). Functional description of MINTO, a Mixed INTeger Optimizer. *Oper. Res. Lett.* **15**, 47–58.

[48] R. Schultz, L. Stougie, M.H. van der Vlerk (1995). *Solving stochastic programs with integer recourse by enumeration: a framework using Gröbner basis reductions*, Discussion paper TI 95-216, Tinbergen Institute, Amsterdam.

[49] É. Tardos (1985). A strongly polynomial minimum cost circulation algorithm. *Combinatorica* **5**, 247–255.

[50] S. Thienel (1995). *ABACUS, A Branch-And-CUt System*, PhD thesis, Institut für Informatik, Universität zu Köln, Germany.

[51] R.R. Thomas (1995). A geometric Buchberger algorithm for integer programming. *Math. Oper. Res.* **20**, 864–884.

[52] A. Wagelmans, S. van Hoesel, A. Kolen. Economic lot sizing: an $O(n \log n)$ algorithm that runs in linear time in the Wagner-Whitin case. *Oper. Res.* **40**, S145–S156.

LP-based solution methods for single-machine scheduling problems *

J.M van den Akker
National Aerospace Laboratory NLR
P.O.Box 90502, 1006 BM Amsterdam
E-mail: vdakker@nlr.nl

Consider the following situation. A mechanic has to repair a number of cars. For each car we know the time at which it is brought to the garage, the time the mechanic needs to repair it, and the time at which its repair should be finished. The mechanic can only repair one car at the same time, and it seems that time is running out on him. We are looking for a schedule for the mechanic in which the number of cars that are not ready at the agreed time is minimal.

This is a classical example of a single-machine scheduling problem: the mechanic is the machine. Sequencing and scheduling is concerned with the allocation of jobs to machines with a restricted availability and capacity. A *schedule* specifies for each job by which machine and at what time it is processed. We are looking for a schedule with minimal cost, where the cost of a schedule is defined as some function of the job completion times. The variety of properties of jobs, machines, and objective functions leads to huge amount of different scheduling problems. For many of these problems, we do not know a fast algorithm that solves the problem at hand to optimality; this is even true for problems with only one machine, like the one described above.

We analyze solution methods for single-machine scheduling problems. We can formulate a single-machine scheduling problem as an integer linear programming problem, i.e., a problem with a given linear objective function that we have to minimize subject to a number of linear constraints, where the variables are allowed to attain integral values only. A simple example of such a problem is the following. Find the solution (x, y) that minimizes $x - 10y$ over the set of points (x, y) that satisfy $x \geq 0$, $5x - 4y \geq 0$, and $5x + 4y \leq 20$ (these are the linear constraints), where x and y are integral (these are the integrality constraints). The set of feasible solutions to this problem, i.e., the set containing all (x, y) satisfying the constraints mentioned above, is depicted in Figure 1. The set of all points (x, y) satisfying the linear constraints corresponds to the bold triangle and its interior. Because of the integrality constraints, the set of feasible solutions is equal to the set of dots in the interior and on the boundary of the triangle.

In the thesis, we discuss different ways to formulate a single-machine scheduling problem as an integer linear programming problem. The solution methods that we study are based on a *time-indexed formulation*. We assume that the planning period is divided into T time

*This is the summary of the PhD thesis with the same title, which is based on research conducted at Eindhoven University of Technology

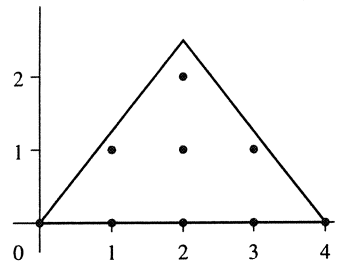


Figure 1: The set of feasible solutions.

periods and that the processing of a job always starts at the beginning of such a time period. For job j and period t, the variable x_{jt} signals if job j is started in period t, i.e., x_{jt} equals 1 if job j is started in period t and 0 otherwise.

We study the set of feasible solutions of the integer linear programming problem given by the time-indexed formulation. In general, solving an integer linear programming problem is hard. If we omit the integrality constraints, then we obtain a *linear programming problem*, which is called the *LP-relaxation*; a linear programming problem can be solved easily, for example by the simplex method.

Removing the integrality constraints extends the set of feasible solutions. In the example depicted in Figure 1, the set of feasible solutions of the LP-relaxation is the bold triangle and its interior, whereas the set of feasible solutions of the integer linear programming problem consists of the dots inside and at the boundary of this triangle. In most situations, the optimal solution of the LP-relaxation will not be integral, i.e., it will be fractional. In case we find a fractional solution, we know that the objective value of each integral solution is greater than or equal to the objective value of this solution. The optimal value of the LP-relaxation is hence a *lower bound* on the optimal value of the integer linear programming problem. To improve this lower bound, we add to the LP-relaxation linear constraints, which exclude fractional solutions of this relaxation but which are satisfied by all integral solutions. Therefore, such constraints are called *valid inequalities*. In the example depicted in Figure 1, $y \leq 2$ is a valid inequality. Figure 2 shows that this inequality cuts off a part of the triangle. For this reason such inequalities are also called *cutting planes*.

We are especially interested in valid inequalities that are necessary in the description of the *convex hull* of the set of integral solutions. Such valid inequalities are called *facets*. In Figure 2 the convex hull of the set of integral solutions is indicated by the dotted triangle. On the x-axis, the edge of this triangle coincides with one of the edges of the bold triangle. The facets are given by the edges of the dotted triangle. If all facets are added to the LP-relaxation, then the optimal solution is guaranteed to be integral. However, for a problem that cannot be solved easily, finding the complete set of facets is also hard.

We study facets for single-machine scheduling problems that are modelled by the time-indexed formulation, where we extended the set of feasible schedules to get a full-dimensional polytope. We consider facets with right-hand sides 1 and 2, i.e., facets of the form $\sum a_{jt}x_{jt} \leq 1$ and $\sum a_{jt}x_{jt} \leq 2$. We derive a number of structural properties of these facets. From these properties, it follows that all facets with right-hand side 1 are contained in one class of valid inequalities. Since all inequalities in this class are indeed facet-defining, all facets with right-

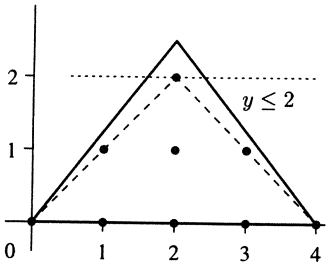


Figure 2: A valid inequality and the convex hull of the set of integral solutions.

hand side 1 are characterized in this way. The structural properties of facets with right-hand side 2 lead to three classes of valid inequalities containing all facets. Most valid inequalities in these classes were unknown. We show which of the valid inequalities in these three classes are facets. In this way, we have characterized all facets with right-hand side 2. We show that under mild conditions the characterized facets also define facets for the convex hull of the original set of feasible solutions.

The next step is to investigate how the characterized facets can be used to solve single-machine scheduling problems. As the number of characterized facets is rather large, it is impractical to add them all to the LP-relaxation simultaneously. For this reason, we proceed in the following way. First, we solve the LP-relaxation. If the solution is not integral, we look within the set of characterized facets for inequalities that are violated by the current solution. Then we add some of these violated inequalities, i.e., cutting planes, to the LP-relaxation. The current solution is no longer feasible for the resulting linear programming problem. We solve this linear programming problem; if the solution is fractional, we again add violated inequalities. This process is repeated, either until we have found an integral solution, or until we cannot find any more violated inequalities. The identification of inequalities that are violated by a given fractional solution is called *separation*. In the thesis, we derive separation algorithms for the characterized facets with right-hand sides 1 and 2. Although the addition of cutting planes does not always lead to finding integral solutions, the lower bound obtained from the LP-relaxation is strongly improved.

If we cannot find any more inequalities that are violated by our fractional solution, we apply *branch-and-bound*. In a branch-and-bound algorithm, the problem is divided into subproblems by partitioning the set of feasible solutions (this is the branching part). For each subproblem we determine a lower bound on the objective value of any solution of this subproblem (this is the bounding part). If a feasible integral solution with value less than or equal to this lower bound is known, then the subproblem can be skipped from further consideration, because it cannot have a solution that is better than the best known solution so far. In the branch-and-bound algorithm we use the lower bound obtained by the addition of cutting planes to the LP-relaxation. A branch-and-bound algorithm in which such kind of lower bounds are used is called a *branch-and-cut* algorithm. The performance of the branch-and-cut algorithm is influenced by the way in which the problem is partitioned into subproblems, by the order in which these subproblems are analyzed, and by the quality of the feasible integral solutions that are generated through a primal heuristic. Another important issue is which of the identified violated inequalities are actually added to the LP-relaxation.

Different possibilities lead to different variants of the branch-and-cut algorithm. We tested different variants and we present the computational results of these tests.

The computational results show that the time-indexed formulation provides very strong lower bounds. However, because of the large number of constraints and variables, the computation of these lower bounds requires large linear programs to be solved, which takes a lot of computation time. For this reason, we study a method which has especially been composed to solve linear programming problems with a large number of variables. This method proceeds as follows. First, we apply *Dantzig-Wolfe decomposition.* This results in a formulation with fewer constraints, but with many more variables. The large number of variables does not pose a problem, since it can be handled by using *column generation.* In a column generation algorithm we consider a restricted problem, in the sense that only a subset of the variables is included; the other variables are implicitly fixed at zero. After solving the restricted problem, we check if variables outside the restricted problem are needed to improve the current solution. This check is performed by solving the so-called *pricing problem.* If such variables are identified, we add them to the restricted problem, and the procedure is repeated. We show that for our formulation solving the pricing problem amounts to determining the shortest path in a network. This implies that the pricing problem can be solved efficiently, which is very important for the performance of the column generation algorithm. Our computational results show that especially for larger problem instances the LP-relaxation of the time-indexed formulation can be solved considerably faster by column generation. We also discuss the combination of column generation and the addition of cutting planes, and the combination of column generation and branch-and-bound. We give computational results based on a preliminary implementation of the combination of column generation and the addition of violated inequalities.

Graphs, distances and eigenvalues

Edwin R. van Dam
Department of Econometrics
Tilburg University
E-mail: Edwin.vanDam@kub.nl

1 Introduction

We have developed a tool with which we can derive bounds on subsets of vertices in a graph satisfying some distance property, in terms of the (Laplace) eigenvalues of the graph. The significance of such a bound is obvious when the related problem is NP-complete, and the determination of a good solution will be computationally hard, for instance, when we bound the sizes of two sets of vertices at some given distance, or in particular, when we bound the size of two equally large sets of vertices with no edges in between.

When we are dealing with a relatively easy problem, like the determination of the diameter of a graph, a bound in terms of (some of) the eigenvalues of the graph can still be useful. Of course, when given a graph, we should not compute its eigenvalues, and then derive the diameter bound, when it is much easier to find the diameter explicitely with a polynomial-time algorithm. However, sometimes we do not know the full structure of a graph, while we may have some information about its eigenvalues. This is for example the case with so-called Ramanujan graphs, graphs which are known to have good expanding properties (cf. [9]), and which therefore can be used to build good (and large) information networks (cf. [1]).

In this paper we consider undirected graphs. The Laplace eigenvalues of such a graph are the eigenvalues of the associated Laplace matrix Q, which is a square matrix with rows and columns labelled by the vertices of the graph, defined by $Q_{xx} = d_x$, and $Q_{xy} = -A_{xy}$ for $x \neq y$, where d_x is the vertex degree of x, and A_{xy} denotes the number of edges between x and y. The Laplace matrix is a positive semidefinite matrix.

The results in this paper are mainly from [3] and [4] (see also Chapter 5 of the author's thesis [2]).

2 The tool

Let P_m be the set of polynomials p with real coefficients of degree m such that $p(0) = 1$. Our main tool will be the following theorem.

Theorem 1 *[4] Let G be a connected graph on v vertices with $r+1$ distinct Laplace eigenvalues $0 = \theta_0 < \theta_1 < \ldots < \theta_r$. Let m be a nonnegative integer and let X and Y be sets of vertices, such that the distance between any vertex of X and any vertex of Y is at least $m+1$. Then*

$$\frac{|X||Y|}{(v-|X|)(v-|Y|)} \leq \min_{p \in P_m} \max_{i \neq 0} p^2(\theta_i).$$

First of all, this implies that the diameter of a graph is at most r, otherwise we would be able to find nonempty sets X and Y at distance $r+1$, and a polynomial $p \in P_r$ which is zero at the r nonzero eigenvalues, giving an upper bound zero, contradicting the theorem.

Using the theory of uniform approximations of continuous functions we were able to rewrite the upper bound as (cf. [3])

$$\min_{p \in P_m} \max_{i \neq 0} p^2(\theta_i) = \max_{T \subset \{1,\ldots,r\}, |T|=m+1} \left(\sum_{j \in T} \prod_{i \in T \setminus \{j\}} \frac{\theta_i}{|\theta_j - \theta_i|} \right)^{-2}.$$

3 Sets of vertices at given distance

A second application of the theorem now gives a bound on the number of vertices at distance r (hence at extremal distance) from an arbitrary vertex.

Theorem 2 *[3] Let G be a connected graph on v vertices with $r+1$ distinct Laplace eigenvalues $0 = \theta_0 < \theta_1 < \ldots < \theta_r$. Let x be an arbitrary vertex, and let k_r be the number of vertices at distance r from x. Then*

$$k_r \leq \frac{v}{1 + \frac{\gamma^2}{v-1}}, \text{ where } \gamma = \sum_{j \neq 0} \prod_{i \neq 0,j} \frac{\theta_i}{|\theta_j - \theta_i|}.$$

Of course we should note that computing the number of vertices at distance r can be done in polynomial time.

It is, however, not hard to show that deciding whether there exist two equally large sets of vertices of size κ with no edges in between (disconnected vertex sets) is an NP-complete problem (cf. [5, problem GT24]). From our tool we derive that

$$\kappa \leq \frac{1}{2} v (1 - \frac{\theta_1}{\theta_r}),$$

by using the polynomial $p(z) = 1 - \frac{2z}{\theta_1 + \theta_r}$. Haemers [6] used this method to derive a bound due to Helmberg, Mohar, Poljak and Rendl [8] on the bandwidth of a graph. Note that computing the bandwidth is also an NP-co

A similar problem is to find two sets of vertices of size κ_r which are at (extremal) distance r. Here we find that

$$\kappa_r \leq \frac{v}{1+\gamma}, \text{ where } \gamma = \sum_{j \neq 0} \prod_{i \neq 0,j} \frac{\theta_i}{|\theta_j - \theta_i|}.$$

A related problem is the problem of finding two sets of vertices with no edges in between (disconnected vertex sets) such that the product of the sizes of these sets is maximized. This problem has an application in information theory and is studied by Haemers [7]. By using Theorem 1 he finds that

$$\max_{X,Y \text{ disconnected}} \sqrt{|X||Y|} \leq \frac{1}{2} v (1 - \frac{\theta_1}{\theta_r}).$$

We should note that all bounds mentioned so far are attained by infinitely many graphs.

4 The diameter

To obtain a bound on the diameter $d(G)$ of a graph G we shall not solve the maximization problem in the upper bound of Theorem 1, but a relaxation of this problem. Instead of evaluating the polynomials at the discrete values $\theta_1, \ldots, \theta_r$, we evaluate them at the interval $[\theta_1, \theta_r]$, so that for the upper bound we find

$$\min_{p \in P_m} \max_{i \neq 0} p^2(\theta_i) \leq \min_{p \in P_m} \max_{z \in [\theta_1, \theta_r]} p^2(z).$$

The solution of the relaxation can be described in terms of Chebyshev polynomials (cf. [10]). The polynomial $C_m(z) = \frac{T_m(\frac{\theta_r+\theta_1-2z}{\theta_r-\theta_1})}{T_m(\frac{\theta_r+\theta_1}{\theta_r-\theta_1})}$ where $T_m(z) = \cosh(m \cosh^{-1}(z))$, solves the problem, thus giving the following diameter bound.

Theorem 3 *[4] Let G be a connected noncomplete graph on v vertices with smallest nonzero Laplace eigenvalue θ_1 and largest Laplace eigenvalue θ_r, then*

$$d(G) \leq \frac{\cosh^{-1}(v-1)}{\cosh^{-1}(\frac{\theta_r+\theta_1}{\theta_r-\theta_1})} + 1.$$

We shall apply this bound to Ramanujan graphs. These are regular graphs, say of degree k, for which $\max_{i:0<\theta_i<2k} |k - \theta_i| \leq 2\sqrt{k-1}$ (cf. [9]). Now it follows from Theorem 3 that for a nonbipartite Ramanujan graph G on v vertices we have

$$d(G) < \frac{2 \log 2(v-1)}{\log(k-1)} + 1$$

(and for bipartite Ramanujan graphs we obtain a similar bound after applying an improved diameter bound for bipartite graphs (cf. [4])), which means that a Ramanujan graph has a small diameter, since the upper bound is approximately twice a (trivial) lower bound for the diameter of any k-regular graph on v v

The diameter bound of Theorem 3 also has an interesting application in coding theory. Using the coset graph of a linear code, it gives a bound for the covering radius of the code in terms of its dual weights (cf. [4]).

References

[1] F. Bien, Constructions of telephone networks by group representations, *Notices A.M.S.* **36** (1989), 5-22.

[2] E.R. van Dam, *Graphs with few eigenvalues - an interplay between combinatorics and algebra*, CentER dissertation series 20, Thesis, Tilburg University, 1996.

[3] E.R. van Dam, Bounds on special subsets in graphs, eigenvalues and association schemes, *J. Alg. Comb.* (to appear).

[4] E.R. van Dam and W.H. Haemers, Eigenvalues and the diameter of graphs, *Linear Multilin. Alg.* **39** (1995), 33-44.

[5] M.R. Garey and D.S. Johnson, *Computers and intractibility: a guide to the theory of NP-completeness*, Freeman, San Francisco, 1979.

[6] W.H. Haemers, Interlacing eigenvalues and graphs, *Linear Algebra Appl.* **226-228** (1995), 593-616.

[7] W.H. Haemers, Disconnected vertex sets and equidistant code pairs, *Electronic J. Combinatorics* **4** (1997), R7 (at URL http://www.combinatorics.org).

[8] C. Helmberg, B. Mohar, S. Poljak and F. Rendl, A spectral approach to bandwidth and separator problems in graphs, *Linear Multilin. Alg.* **39** (1995), 73-90.

[9] A. Lubotzky, R. Phillips and P. Sarnak, Ramanujan graphs, *Combinatorica* **8** (1988), 261-277.

[10] T.J. Rivlin, *Chebyshev polynomials* (2nd ed.), Wiley, New York, 1990.

Algorithms for Graphs of Small Treewidth*

Babette de Fluiter
University of Utrecht
E-mail: babette@cs.ruu.nl

Abstract

In this paper I give an overview of my PhD thesis, entitled *Algorithms for Graphs of Small Treewidth* (ISBN 90-393-1528-0). This thesis was completed under supervision of Dr. Hans Bodlaender and Prof. Dr. Jan van Leeuwen in the Department of Computer Science at Utrecht University.

1 Introduction

Many real-life problems can be modeled as optimization or decision problems on graphs. Unfortunately, many graph problems that model real-life problems are NP-hard, meaning that there are (probably) no efficient algorithms which solve these problems. A way of overcoming this disadvantage is to discover a special structure in the graphs modeling the real-life problem which may help in finding a more efficient algorithm for the problem. For instance, the input graphs may have a special structure that assures that the problem at hand is easy to solve. Another possibility is that the problem can be decomposed into subproblems, and that the structure of the input graphs assures that some of these subproblems are easy to solve. This might help in finding a more efficient algorithm that computes an optimal solution for the complete problem, or in finding an efficient algorithm that computes a good approximation of the optimal solution.

One suitable structure is the tree-structure: it appears that many graph problems that are hard in general, are efficiently solvable on trees, often by applying dynamic programming on the tree. For most practical cases however, the class of trees is too limited. Therefore, we consider extensions of the class of trees which are more useful in practice, namely the classes of graphs of *treewidth* at most k and *pathwidth* at most k, for any positive integer k. Intuitively, the treewidth of a graph measures the resemblance of the graph to a tree: the smaller the treewidth, the larger the resemblance.

Definition 1 [10]. Let $G = (V, E)$ be a graph. A *tree decomposition* TD of G is a pair $(T, \mathcal{X})$, where $T = (I, F)$ is a tree, and $\mathcal{X} = \{X_i \mid i \in I\}$ is a family of subsets of V, one for each node of T, such that

- $\bigcup_{i \in I} X_i = V$,
- for every edge $\{v, w\} \in E$, there is an $i \in I$ with $v \in X_i$ and $w \in X_i$, and

*This research was supported by the Foundation for Computer Science (S.I.O.N) of the Netherlands Organization for Scientific Research (N.W.O.).

- for all $i, j, k \in I$, if j is on the path from i to k in T, then $X_i \cap X_k \subseteq X_j$.

The *width* of a tree decomposition $((I, F), \{X_i \mid i \in I\})$ is $\max_{i \in I} |X_i| - 1$. The treewidth of a graph G is the minimum width over all possible tree decompositions of G.

An example of a graph G of treewidth two and a tree decomposition TD of width two of the graph is given in Figure 1. A tree decomposition is depicted as a tree in which each node i contains the vertices of X_i.

Definition 2 [11]. A *path decomposition* of a graph is a tree decomposition with the extra restriction that the tree is a path. A graph has pathwidth at most k if there is a path decomposition of the graph of width at most k.

The graph depicted in Figure 1 has pathwidth three, and a path decomposition PD of width three of G is also depicted in Figure 1.

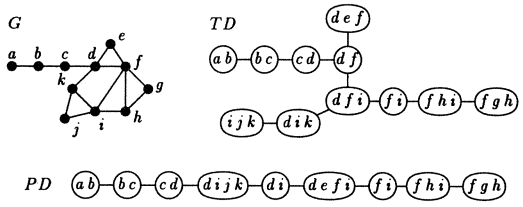


Figure 1: A graph G of treewidth two and pathwidth three, a tree decomposition TD of width two of G, and a path decomposition PD of width three of G.

Many (hard) problems can be solved efficiently on graphs of small treewidth, using the tree-like structure of the graphs. For instance, a large class of problems can be solved efficiently by applying dynamic programming on a tree decomposition of small width of the graph. These algorithms usually work on rooted tree decompositions of small width with $O(n)$ nodes. Examples of problems that can be solved efficiently on graphs of small treewidth by using the dynamic programming approach are MAX INDEPENDENT SET, TRAVELING SALESMAN, CHROMATIC NUMBER, and MIN DOMINATING SET. These problems can all be solved in $O(n)$ time sequentially and in $O(\log n)$ time in parallel with $O(n/\log n)$ processors (the algorithms are exponential in the treewidth of the graph). To solve problems this way, it is necessary to find a tree decomposition of small width of the given graph first. Fortunately, for each positive integer k, there is a linear time algorithm which, given a graph, finds a tree decomposition of width at most k of the graph, if one exists (this algorithm is again exponential in k) [3]. In parallel, the problem can be solved in $O(\log^2 n)$ time with $O(n/\log^2 n)$ processors on an EREW or CRCW PRAM [4].

It appears that many graph problems have practical instances in which the input graphs have small treewidth. Also, many (practical) graph problems require that the treewidth or pathwidth of the input graph is small.

Unfortunately, many algorithms solving problems on graphs of small treewidth are only efficient in theory: the running time of the algorithms is usually exponential in the treewidth of the graph. This means that if the input graph is only of moderate size, and the bound on the treewidth is six or more, then in the running time of the algorithm, the factor that

is exponential in the treewidth is likely to overtake the factor that is polynomial in the size of the graph. This holds e.g. for the algorithms for finding a tree or path decomposition of width at most k of a given graph, if one exists (k constant).

The goal of the thesis is to give efficient sequential and parallel algorithms for several problems on graphs of small treewidth or pathwidth. We consider both graph problems which require that the treewidth or pathwidth of the input graph is bounded by some constant, and graph problems which are hard on general graphs, but have more efficient solutions on graphs of small treewidth or pathwidth. The aim is to design algorithms which are not only theoretically efficient, but are also efficient in practical applications. The thesis comprises two subjects: *DNA physical mapping* and *reduction algorithms.*

2 DNA Physical Mapping

In the thesis, we consider two problems which originate from molecular biology and are known as *sequence reconstruction* problems that occur in DNA physical mapping. In both problems, the input consists of k copies of a DNA string that are fragmented, and for each pair of fragments, either it is known that they overlap, or it is known that they do not overlap, or nothing is known about their overlap. In one of the problems, we additionally have the information that all fragments have the same length. There is no explicit information on the order of the fragments in the DNA string, or on the copy from which each fragment originates. The problem is to recover the complete overlap information of the fragments, and with this, the order of the fragments in each copy of the DNA string.

The input of the problems is modeled as graph $G = (V, E)$ and an extra set of edges F: the vertices of the graph represent the fragments, and for each two vertices in V, there is an edge between u and v in E if we know that the corresponding fragments overlap, and there is an edge between u and v in F if the corresponding fragments possibly overlap, i.e. are not known not to overlap. The complete overlap information can again be represented by a graph. This graph must be an *interval graph* or a *unit-interval graph* in which the clique size is at most k.

Definition 3. A graph $G = (V, E)$ is an interval graph if there is a function Φ which maps each vertex $v \in V$ to an interval on the real line, such that for each $u, v \in V$ with $u \neq v$,

$$\Phi(u) \cap \Phi(v) \neq \emptyset \quad \Leftrightarrow \quad \{u, v\} \in E.$$

The function Φ is called an *interval mapping.*

An interval graph G is called a unit-interval graph graph there is an interval mapping for G in which all intervals have the same length.

The two sequence reconstruction problems can be modeled as follows.

INTERVALIZING SANDWICH GRAPHS (ISG)
Instance: A graph $G = (V, E_1)$, a set E_2 of edges with $E_1 \subseteq E_2$, a positive integer k.
Question: Is there an interval graph $G = (V, E)$ such that $E_1 \subseteq E \subseteq E_2$, and G has no cliques of size more than k?

UNIT-INTERVALIZING SANDWICH GRAPHS (UISG)
Instance: A graph $G = (V, E_1)$, a set E_2 of edges with $E_1 \subseteq E_2$, a positive integer k.
Question: Is there a unit-interval graph $G = (V, E)$ such that $E_1 \subseteq E \subseteq E_2$, and G has no cliques of size more than k?

For fixed k, the problems are denoted by k-ISG and k-UISG, respectively.

It is known that if, for any k, $G = (V, E_1)$ and E_2 form a yes-instance of k-ISG or k-UISG, then G has pathwidth at most $k-1$ [6].

In the thesis, we consider k-ISG and k-UISG. We resolve the complexity of k-ISG for all fixed integers $k \geq 2$: we give a linear time algorithm for 2-ISG, a quadratic algorithm for 3-ISG, and we show that k-ISG is NP-complete if $k \geq 4$. Furthermore, we give an $O(n+m)$ time algorithm for 3-UISG (where $m = |E_2|$). There is an algorithm for k-UISG which uses $O(n^{k-1})$ [9]. Our algorithm improves on this result for the case that $k = 3$.

The algorithms for 3-ISG and 3-UISG heavily rely on the fact that yes-instances have pathwidth at most two: a complete characterization of graphs of pathwidth at most two is given first. After that, this characterization is used to give the algorithms for 3-ISG and 3-UISG. The algorithms first check whether the input graph has pathwidth at most two, and if so, they use the structure of the graph to solve 3-ISG and 3-UISG, respectively.

3 Reduction Algorithms

A reduction algorithm is an algorithm which can be used to solve decision or optimization problems of which the input is a graph. A reduction algorithm is based on a finite set of *reduction rules* and a finite set of graphs. Each reduction rule describes a way to modify a graph locally.

An example of a reduction rule is given in Figure 2: a reduction rule consists of a pair of graphs, each with a set of distinguished vertices, called *terminals*. Both graphs have the same set of terminals. A reduction is an application of a reduction rule on a graph G: if a reduction rule (H_1, H_2) is applied to a graph G, then a subgraph G_1 isomorphic to H_1 is taking in G, such that only the terminal vertices may have edges to vertices outside of G_1. Then G_1 is replaced by a subgraph isomorphic to G_2, such that corresponding terminals are mapped to the same vertices (see Figure 2 for an example).

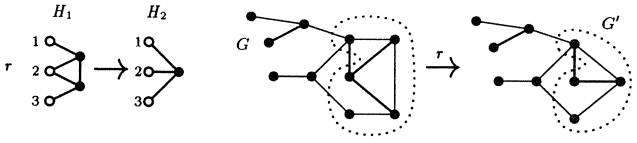


Figure 2: An example of a reduction rule $r = (H_1, H_2)$, and an application of r to a graph G, resulting in graph G'.

The idea of a reduction algorithm is to solve a decision problem by repeatedly applying reduction rules on the input graph until no more rule can be applied. If the resulting graph is in the finite set of graphs, then the algorithm returns **true**, otherwise it returns **false**. Hence the set of reduction rules and the finite set of graphs are problem specific.

In a sequential algorithm, all reductions are performed subsequently, but in a parallel reduction algorithm, non-interfering reductions can be performed at the same time.

It turns out that for many decision and optimization problems, it is possible to generate a set of problem specific reduction rules, and with this set, the problem can be solved efficiently on graphs of small treewidth, both sequentially and in parallel. The sequential algorithms take $O(n)$ time. The parallel algorithms take $O(\log n \log^* n)$ time with $O(n/(\log n \log^* n)$

processors on an EREW PRAM, or $O(\log n)$ time with $O(n/\log n)$ processors on a CRCW PRAM ($\log^* n$ denotes the amount of times we have to replace n by the value of $\log n$ in order to get a value that is at most one. For all practical values of n, $\log^* n \leq 5$.). The sets of problem specific reduction rules can be generated from a description of the problem in *monadic second order logic* (note that this generated set only depends on the problem, but may be rather large).

An advantage of reduction algorithms is that they are easy to implement: the difficulty of a problem is hidden in the design of the problem specific set of reduction rules, and not in the reduction algorithm itself. Another advantage of reduction algorithms over other algorithms on graphs of small treewidth is that a reduction algorithm works directly on the input graph, and hence no tree decomposition of small width of the graph is needed. As the running times of the algorithms for finding a tree decomposition of small width of a graph are not efficient in practice, this makes reduction algorithms potentially more practical (if the set of reduction rules is not too large).

In the thesis, we present the basic theory on reduction algorithms and we show that reduction algorithms can be used to solve large classes of decision and optimization problems on graphs of bounded treewidth, thus giving a comprehensive overview of results presented in [1, 2, 4].

One drawback of reduction algorithms is that they only solve decision and optimization problems. For decision problems, the algorithms only return 'yes' or 'no', but they do not return a solution for the problem if the answer is 'yes'. Similarly, for optimization problems, only the optimal value is returned, but no optimal solution of the problem is returned. In the thesis, we extend the theory of reduction algorithms to *constructive* reduction algorithms, which also return an (optimal) solution for the problem at hand, if one exists.

The constructive reduction algorithms consist of two phases. In the first phase, an ordinary reduction algorithm is applied. If the reduced graph is not a yes-instance, then false is returned. Otherwise, the second phase is started. In the second phase, first a solution is constructed for the small graph. After that, the reductions that are applied in the first phase are undone one by one, in reversed order. Each time a reduction is undone, the solution of the current graph is reconstructed into a solution of the new graph. This eventually terminates with the original input graph and a solution for this graph. In case of an optimization problem, this solution is optimal.

In the thesis, we show that the theory of constructive reduction algorithms can be applied to a large class of constructive decision and optimization problems on graphs of bounded treewidth, and the resulting algorithms run in the same time as the ordinary reduction algorithms. These results again show how the problem specific set of reduction rules can be generated from the problem description, and how the reconstruction algorithm in phase 2 can be done.

There are a number of problems on graphs of bounded treewidth for which the technique of constructive reduction algorithms can not be applied directly, i.e. we know no algorithm to generate a set of reduction rules with a reconstruction algorithm, from a problem description. For two of these problems, we show in the thesis that they can be solved with the constructive reduction technique anyhow. This results in new, parallel algorithms which are more efficient than previous algorithms for these problems. The first problem is concerned with *series-parallel graphs.*

Definition 4. A series-parallel graph is a triple (G, s, t), where G is a multigraph, and s and t are distinct vertices of G, for which one of the following conditions holds

- G consists of one edge between s and t,

- (G, s, t) can be obtained by a *series composition* of two series-parallel graphs (G_1, s_1, t_1) and (G_2, s_2, t_2), i.e. G is obtained by taking the disjoint union of G_1 and G_2, and then identifying t_1 with s_2 and letting $s = s_1$ and $t = t_2$.
- (G, s, t) can be obtained by a *parallel composition* of two series-parallel graphs (G_1, s_1, t_1) and (G_2, s_2, t_2), i.e. G is obtained by taking the disjoint union of G_1 and G_2, and then identifying s_1 with s_2 and t_1 with t_2, and letting $s = s_1 = s_2$ and $t = t_1 = t_2$.

A series-parallel graph can be decomposed into series and parallel compositions. An *sp-tree* is a tree which reflects such a decomposition.

The problem we consider is the problem of checking whether a given triple (G, s, t) is a series-parallel graph, and if so, constructing an sp-tree for (G, s, t). We show that, with the technique of constructive reduction algorithms, this problem can be solved in $O(\log m \log^* m)$ time with $O(m/(\log m \log^* m))$ processors on an EREW PRAM, and in $O(\log m)$ time with $O(m/\log m)$ processors on a CRCW PRAM (where m denotes the number of edges of the graph). To this end, we explicitly give a set of 18 reduction rules, and we show that, with this set, series-parallel graphs can be recognized, and in the second phase of the algorithm, an sp-tree can be reconstructed when the reductions are undone. The algorithm improves in efficiency on the parallel algorithms of [5, 7, 8].

The second problem is the problem of finding a tree decomposition of width at most two of a graph, if one exists. This problem is closely related to the problem of recognizing series-parallel graphs, since any series-parallel graph has treewidth at most two. We again use the technique of constructive reduction algorithms: we extend the set of reduction rules for series-parallel graphs with five extra rules, and we show that this set can be used to recognize graphs of treewidth at most two. We also show how, in the second phase of the constructive reduction algorithm, a tree decomposition of width at most two of the graph can be maintained. This algorithm runs in $O(\log n \log^*)$ time with $O(n/(\log n \log^* n))$ processors on an EREW PRAM, and in $O(\log n)$ time with $O(n/\log n)$ processors on a CRCW PRAM. It improves in efficiency on the algorithm of [4], which uses $O(\log^2 n)$ time with $O(n/\log^2 n)$ operations, both on an EREW and a CRCW PRAM.

References

[1] S. Arnborg, B. Courcelle, A. Proskurowski, and D. Seese. An algebraic theory of graph reduction. *J. ACM* **40**, 1134–1164, 1993.

[2] H. L. Bodlaender. On reduction algorithms for graphs with small treewidth. In J. van Leeuwen, editor, *Proceedings of the 19th WG*, volume 790 of *Lecture Notes in Computer Science*, pages 45–56. Springer-Verlag, Berlin, 1994.

[3] H. L. Bodlaender. A linear time algorithm for finding tree-decompositions of small treewidth. *SIAM J. Comput.* **25**, 1305–1317, 1996.

[4] H. L. Bodlaender and T. Hagerup. Parallel algorithms with optimal speedup for bounded treewidth. In Z. Fülöp and F. Gécseg, editors, *Proceedings of the 22nd ICALP*, volume 944 of *Lecture Notes in Computer Science*, pages 268–279. Springer-Verlag, Berlin, 1995.

[5] D. Eppstein. Parallel recognition of series parallel graphs. *Information and Computation* **98**, 41–55, 1992.

[6] M. R. Fellows, M. T. Hallett, and H. T. Wareham. DNA physical mapping: Three ways difficult (extended abstract). In T. Lengauer, editor, *Proceedings of the 1st ESA*, volume 726 of *Lecture Notes in Computer Science*, pages 157–168. Springer-Verlag, Berlin, 1993.

[7] X. He. Efficient parallel algorithms for series-parallel graphs. *J. Algorithms* **12**, 409–430, 1991.

[8] X. He and Y. Yesha. Parallel recognition and decomposition of two terminal series parallel graphs. *Information and Computation* **75**, 15–38, 1987.

[9] H. Kaplan and R. Shamir. Pathwidth, bandwidth and completion problems to proper interval graphs with small cliques. *SIAM J. Comput.* **25**, 540–561, 1996.

[10] N. Robertson and P. D. Seymour. Graph minors. I. Excluding a forest. *J. Comb. Theory Series B* **35**, 39–61, 1983.

[11] N. Robertson and P. D. Seymour. Graph minors. II. Algorithmic aspects of tree-width. *J. Algorithms* **7**, 309–322, 1986.

Planning shifts in a 36-hour cyclical roster

Richard Freling
Nanda Piersma
Albert P.M. Wagelmans
Econometric Institute
Erasmus University
E-mail: piersma@few.eur.nl

1 Introduction

In the public sector and certain industries where operations are ongoing 24 hours a day, employees work in shifts, usually of 8 hours in length with a 30 minute break. The shifts, assigned to an employee, are currently based on a 40-hour working week. The (40-hour) rosters are often handmade and the result of years of planning experience. When the workload was cut down to 38-hour, people continued to work 40 hour/week, and extra free time was assigned to each employee (the so called ATV or ADV). The planners continued to work with the existing 40-hour rosters.

Lately, a considerable number of corporations in business and the public sector in the Netherlands agreed to start a 36-hour working week. At the same time that the further cut back in working hours was negotiated, national labor laws were updated and became more severe with respect to shift length and shift sequencing. Many existing 40-hour roster are in conflict with the new laws. Besides, when working 36 hours, every employee will get too much time off in a 40-hour roster and too many shifts will be unfilled.

The planners are thus faced with the problem to design new, more complex rosters. The complexity is increased due to stricter labor laws and because, over the years, there is an increasing need for rosters that consider social aspects (such as day care, car pooling and working part-time). This has resulted into an awareness of OR techniques to support the rostering process and a need for decision support systems that will help the planners with their task.

In this paper we will describe such a complex workforce rostering problem for a detention center in Amsterdam. Although the mathematical models and techniques used are not new, it will give the reader insight into the state of the art of planning systems that are now used in practice. For security reasons we cannot give results, or describe every detail of the application.

2 Problem formulation

The workforce at the detention centre can be considered to be a homogeneous group such that every person can perform every task. The workforce is fixed, there are no temporary employees allowed because the work demands a proper training and every employee must get

a background security check. The employees should in principle be assigned to one particular unit, but can be assigned to other units when needed.

The current 40-hour roster is cyclic, such that everyone has the same roster over time. The roster consists of N weeks and is made for N employees. Employee 1 is assigned to week schedule 1 in week 1, to week schedule 2 in week 2 and so on. In week N+1 the employee is again assigned to week schedule 1. The N week schedules together assure that all tasks on each day are done. The new roster must assure that every person works approximately 36 hours a week and it should also be cyclic. The planners asked for a roster that plans an employee off-duty during one weekend and on-duty during the next.

In the 40-hour roster, the tasks are combined into usual shift types, like a morning shift, an evening shift etc. The shift lengths are not fixed as long as there is a 30 minute overlap between successive shifts for a transfer of the shift. Each shift type is filled multiple times. For the detention center a critical new rule is that an employee has at least 11 hours rest between two shifts. This means that after a late shift, ending at say 11 pm, the employee cannot be assigned to a morning shift before 10 am. In the old roster it was common practice to combine a late shift with an early morning shift.

In the standard approach to shift scheduling the first step is to determine the required manpower at each hour of the day, and to define shifts that will accomodate the demand. However, the workload of each unit depends on the workforce assigned to the centre by the government rather than on the tasks. When more personnel is assigned to a unit there can be more activities leading to an increase in the tasks for the workforce. The shift lengths and shift types are thus determined by the workforce. It was decided to maintain the shift types of the 40-hour roster and to adjust the start and end time of the shifts to assure a 36-hour working week. The planners where advised to consider a systematic procedure to determine shifts from a given formation of personnel.

The second crucial step in the planning proces is to define a two-week working schedule. This schedule starts on a free Sunday, assigns exactly two shifts during the middle weekend and ends with a free Saturday. Successive two-week schedules thus have the required property that an employee is off-duty one weekend and on-duty the next. The labor laws concerning the succession and sequencing of shifts within the two-week schedules are recorded in a computer program. This computer program can check if a given two-week schedule satisfies the labor laws. It can also generate all feasible two-week schedules. Each schedule contains a combination of shifts that assure a 36-hour working week and that satisfies the labor laws. For our application all the feasible schedules, approximately 40.000) were generated and stored in a binary file.

When combining these schedules into rosters only a few more labor laws needed to be checked (like an uninterrupted rest of 60 hours within an 9 day sequence). Most labor laws are satisfied in the feasible two-week schedules. The roster problem then becomes to find an optimal combination of the (two-week) schedules into a cyclic roster for all personnel in a working unit. The units are not completely independent (they share for instance the task of isolation cells duty), but for the planning proces a simple preprocessing of the common tasks will suffice to make the units independent.

The objective criteria for an optimal roster are not evident. Since the detention centre has a government assigned workforce, it is not neccessary to minimize the number of employees to perform all tasks. The rosters are solely evaluated by social aspects, such as the spread of the off days, the number of early shifts during a week, et cetera. The planners thus need an interactive roster generating system, that can distinguish between popular and less popular schedules. A decision support system should thus find one or more feasible schedules under

social restrictions.

In the literature on shift scheduling, the roster problem is often formulated as a Generalized Set Partitioning problem:

$$\min \sum_{j=1}^{n} c_j x_j + \sum_{i=1}^{m} d_i s_i$$

subject to

$$\sum_{j=1}^{n} a_{ij} x_j + s_i = b_i \qquad i = 1, \ldots, m$$

$$x_j, s_i \geq 0 \text{ integer } j = 1, \ldots, n; i = 1, \ldots, m.$$

with m tasks, n schedules and

$$\begin{aligned}
c_j &= \text{costs for schedule} j \\
d_i &= \text{cost for choosing dummy schedule } i \\
a_{ij} &= \begin{cases} 1 & \text{if schedule } j \text{ covers shift } i \\ 2 & \text{if schedule } j \text{ covers shift } i \text{ twice (once every week)} \\ 0 & \text{otherwise.} \end{cases} \\
b_i &= \text{number of schedules that must include shift } i \\
x_j &= \text{number of times that schedule } j \text{ is selected} \\
s_i &= \text{number of times that dummy schedule } i \text{ is selected.}
\end{aligned}$$

The dummy schedule i consists of shift i only. The inclusion of the dummy schedules assures a feasible solution, albeit with very high cost. The model assumes that a selection out of n schedules is determined that covers the number of times that every specific shift is needed. When all cost c_j are equal the model searches for a feasible solution only. A scoring system can be introduced to distinguish between the schedules. Schedules with non popular aspects, for instance with one day on-duty between two days off-duty, get a low score.

3 Solution methods for Cyclic Shift Scheduling

Mathematical models for workforce scheduling have been developed for many applications like nurse scheduling [14, 15], crew scheduling for airlines [13] and mass public transport [2, 3, 8], telephone operator scheduling [11].

While most of the early approaches to workforce scheduling are heuristic, with the increasing computational power, exact planning algorithms can be applied that better solve larger and more complicated planning problems. Exact models for shift scheduling are based on assignment problems [1] or on set covering/ set partitioning problems [4, 6, 10]. In this note we have showed that our application also falls within the framework of set covering and we will follow the well known approach to solve the LP relaxation and search for a feasible solution in the neighborhood of the LP solution.

The Generalized Set Partitioning problem could be routinely solved when the size is reasonable. However, the number of columns (schedules) can be up to 40.000 (or more!) and the model is too large to be solved by standard methods. Crainic and Rousseau [4] suggested a column generation method for the airline crew scheduling problem, where a subset of columns (schedules) was considered. This method became popular in the early

90's when Desrochers and Soumis [5, 6] applied the same technique to the Urban Transit Crew Scheduling Problem.

The basic idea of the method is to start with a small subset of schedules, further referred to as Basic Set, and to solve the LP relaxation of the restricted GSP problem. Based on reduced cost manipulations columns (schedules) are added to the Basic Set. The LP relaxation of the restricted GSP is repeatedly solved until optimality of the LP relaxation of the GSP problem itself has been established.

Earlier column reduction methods were of a subjective nature, based on economic or social criteria. In particular, a set of all feasible columns is generated, and then reduced through rejection of specific columns or column features. For the detention centre all feasible columns are also generated, but instead of reducing the set of columns, a subset of columns is selected and then extended.

Apart from the issue of finding a good Basic Set, the quality of the LP relaxation as a lower bound for the solution value of the GSP problem and the issue of finding a good feasible integer solution are important.

3.1 The LP relaxation

A feasible solution is only found when the Basic Set contains enough schedules to cover every shift. Also, there should be enough schedules such that "overcovering", that is unnecessary multiple covering of a single shift is avoided. The generation of feasible schedules can be enumerative or, when we do not want to generate all feasible schedules, it can be based on a priority principle of the following type

$$P_{ij} = \frac{b_{kj}}{n_{ij}} \text{ with } k = i \bmod 7 \text{ if } i > 7$$

and P_{ij} is a performance measurement for task j on day i, n_{ij} counts the number of times that task j on day i belongs to a generated schedule, and b_{kj} is the number of times that task j on day k should be covered. In the process of generating schedules for the Basic Set the priorities are adjusted. New schedules are chosen based on the highest priority scores. Of course the labor laws are checked in the generation of the basic schedules.

The Basic Set is extended with schedules that are added based on the ideas of the column generation method described by Desrochers and Soumis. When the solution method stops it returns an optimal solution of the LP relaxation of the GSP problem. The Basic Set has been extended with a number of columns (schedules). The LP relaxation provides us with a lower bound to the solution value of the (integer) GSP problem.

From the literature ([9, 12]) it is well known that the LP relaxation yields a good lower bound for Set Covering and Set Partitioning problems. It seems sensible to apply the LP relaxation for the determination of a good Basic Set and for a good lower bound. Lagrangean relaxation can also be used to derive a lower bound for the solution value of the GSP problem. However, a well known consequence of the strong duality theorem of linear programming is that Lagrangean relaxation yields (at best) the same bound.

3.2 A Feasible Integer Solution

The GSP model with a fixed Basic Set, such as is determined by the LP relaxation, can be solved using standard solvers such as CPLEX. For the detention centre, there was a need for a decision support system that was independent of a costly solver.

Wedelin [17] suggests a very simple iterative search algorithm based on a Lagrangean dual approach with cost pertubation for 0-1 integer linear programming problems to generate a good feasible 0-1 solution. The Basic Set of the optimal solution of the LP relaxation is used with the generation of a feasible 0-1 solution. This method has been succesfully applied in the Carmen system for airline crew scheduling and is used by major airlines in Europe.

The GSP problem is given with decision variables that are integer, and not only restricted to 0 and 1. The solution of the 0-1 GSP is of course a solution for the general GSP. The method can easily be adjusted to allow a specific schedule more than once in the feasible integer solution by adding copies of the selected schedules to the Basic Set and generating a new 0-1 solution.

For the detention centre, the units have the same schedules and size. All feasible schedules are generated, a total of 40.000 schedules, on a Pentium 100 Mhz. PC in 500 seconds. The schedules are stored in a binary coded file and 150 randomly selected schedules are added to an initial Basic Set. The file was preprocessed several times after consulting the planners, who qualified some schedule structures as unacceptable. Currently, the resulting rosters are being evaluated with the planners. The size of the set of feasible schedules will diminish with every roster that is shown to the planners since unacceptable (two-week) schedules are observed rather than unacceptable rosters.

4 Conclusions and further research

With so many corporations shifting to the 36-hour labor week the need for decision support systems is enormous. The method has also succesfully been applied at the GG&GD Rotterdam [16] and at a dutch security firm (with ORTEC consultants, [7]). For all the applications we noticed that the complexity of the labor laws and social requirements make it almost impossible to work without computer support. Even checking the validity of the a given roster is a job not eagerly done by planners, since it consist of checks over long periods and of many different rules.

The decision support systems need to be specially developed for each profession, since labor laws and union treaties are quite diverse. For each application we started with a computer program that checks the feasibility of generated rosters and schedules. The basic approach using shift scheduling based on column generation seems to be very suitable.

Further research is currently done in two directions. First with a increasing number of people working part-time, the rosters need to be more flexible. Scheduling two or more part-timers in a single full-time job forces them to divide the off-duty days among each other. Also, the part-time hours need to be a fixed portion of a full-time job (like 0.5, or 0.75) while in practice there is a wide range in part-time working hours. Preferably, the part-timers will take shifts from the existing full-time roster and combine them into several full-time jobs. Secondly, the new labor laws also complicate exchanging shifts between employees and filling shifts for employees that are sick or on leave. Apart from developing new rosters that allow maximum flexibility in shift exchange and back up shifts, there is also a great need for decision support systems that help decide who to call for back up when an employee is sick or on leave.

References

[1] M. Fischetti A. Caprara and P. Toth. Models and algorithms for the crew rostering

problem. Technical report, DEIS University of Bologna, 1996.

[2] L. Bianco, M. Bielli, A. Mingozzi, S. Ricciardelli, and M. Spadoni. A heuristic procedure for the crew rostering problem. *European Journal of Operational Research*, 58:272–283, 1992.

[3] P. Carraresi and G. Gallo. A multilevel bottleneck assignment approach to the bus drivers' rostering problem. *European Journal of Operational Research*, 16:163–173, 1984.

[4] T.G. Crainic and J.M. Rousseau. The column generation principle and the airline crew scheduling problem. *INFOR*, 25:136–151, 1987.

[5] M. Desrochers and F. Soumis. A column generation approach to the urban transit crew scheduling problem. *Transportation Science*, 23:1–13, 1989.

[6] M. Desrosiers, Y. Dumas, M. Desrochers, F. Soumis, B. Sanso, and P. Trudeau. A break through in airline crew scheduling. Technical report, GERARD, Montreal, 1991.

[7] M. Van Duyn. Kwantitatieve planningsondersteuning bij de Nederlandse Veiligheidsdienst B.V., Master Thesis, Erasmus University, Rotterdam, The Netherlands, 1997.

[8] A. Wren (editor). *Computer scheduling of public transport.* North Holland, 1981.

[9] M.L. Fisher and P. Kedia. Optimal solution of set covering/partitioning problems using dual heuristics. *Management Science*, 36:674–688, 1990.

[10] R. Freling. The column generation pricing problem for crew scheduling. Technical report, Econometric Institute, Erasmus University Rotterdam, 1996.

[11] W.B. Henderson and W.L. Berry. Heuristic methods for telephone operator shift scheduling: an experimental analysis. *Management Science*, 22:1372–1380, 1976.

[12] N. Piersma. *Combinatorial Optimization and Empirical Processes.* PhD thesis, Tinbergen Institute, Amsterdam, The Netherlands, 1993.

[13] D.M. Ryan. The solution of massive generalized set partitioning problems in air-crew rostering. *Journal of the Operational Research Society*, 43:459–467, 1992.

[14] S. Perrott Siferd. Workforce staffing and scheduling: hospital nursing specific models. *European Journal of Operational Research*, 60:233–246, 1992.

[15] L.D. Smith. The application of an interactive algorithm to develop cyclical rotational schedules for nursing personnel. *INFOR*, 14:53–70, 1976.

[16] M. Vink. Kwantitatieve planningsondersteuning bij GG & GD in Rotterdam, Master Thesis, Erasmus University, Rotterdam, The Netherlands, 1996.

[17] D. Wedelin. The design of a 0-1 integer optimizer and its application in the Carmen system. *European Journal of Operational Research*, 87:722–730, 1995.

On the graph parameters of Colin de Verdière*

Hein van der Holst
Program in Applied and Computational Mathematics
Princeton University
E-mail: hein@math.Princeton.EDU

1 Introduction

Let $M = m_{i,j}$ be a real-valued symmetric matrix with nonpositive off-diagonal entries and with arbitrary diagonal entries. Recall that M is called *reducible* if the index set of M can be partitioned into two sets I and J such that $m_{i,j} = 0$ for all $i \in I$ and $j \in J$. A matrix M is called *irreducible* if M is not reducible. If M is irreducible then as a consequence of the Perron-Frobenius theorem (by looking to the matrix $cI - M$ for large c) we have that the smallest eigenvalue λ of M is simple (it has multiplicity 1) and the corresponding eigenvector belonging to λ can be chosen to have all entries positive. Let us state this consequence in a more graph theoretical framework. To the matrix M we can associate a graph $G(M)$ with vertex set the index set of M and with between two vertices i and j an edge if $m_{i,j} \neq 0$. Then irreducible means that the graph is connected. If the graph is not connected then we can find a matrix M such that the largest eigenvalue has multiplicity at least 2. So information of the multiplicity of the smallest eigenvalue of matrices M with $G(M) = G$ gives information about the graph G, namely connectedness of the graph.

More information of the graph can be obtained by not only looking to the smallest eigenvalue but by looking also to the smallest but one. By adding an appropiate diagonal matrix we may assume that the smallest but one eigenvalue is equal to 0.

Let G be a graph with vertex set $\{1, \ldots, n\}$ and let $\mathcal{O}_G$ denote the set of all real-valued symmetric matrices $M = (m_{i,j})$ with

i $m_{i,j} < 0$ if ij is an edge, $m_{i,j} = 0$ if ij is not an edge and $i \neq j$, and

ii M has 1 negative eigenvalue.

So $m_{i,i} \in \mathbb{R}$. We do not assume that G is connected.

What information about the graph can we get if we know that corank(M) $\leq$ t for each $M \in \mathcal{O}_G$? Although the full answer of this question is not known there are some results. In fact we will see that there is an intriguing connection between the maximum corank attained by a matrix $M \in \mathcal{O}_G$ and the topological structure of the graph G. What follows is extracted from [8].

*This research was supported by the Netherlands Organization for Scientific Research (N.W.O.)

2 The parameter $\mu(G)$

For any vector x, let supp(x) denote the support of x (i.e., the set $\{i|x_i \neq 0\}$). Moreover we denote the *positive support* by $\text{supp}_+(x) := \{i|x_i > 0\}$ and the *negative support* by $\text{supp}_-(x) := \{i|x_i < 0\}$.

If $x \in \mathbb{R}^n$ and $I \subseteq V$, then x_I denotes the subvector of x induced by the indices in I. Similarly, if M is an $n \times n$ matrix and $I, J \subseteq V$, then $M_{I\times J}$ denotes the submatrix of M induced by row indices in I and column indices in J.

Theorem 2.1 *Let G be a connected graph and let $M \in \mathcal{O}_G$. Let $x \in \ker(M)$ and let I and J be two components of $G[\text{supp}_+(x)]$. Then there is a $y \in \ker(M)$ with $\text{supp}_+(y) = I$ and $\text{supp}_-(y) = J$, such that y_I and y_J are scalar multiples of x_I and x_J respectively.*

Proof. Let $K := \text{supp}_-(x)$. Since $m_{i,j} = 0$ if $i \in I$, $j \in J$, we have:

$$\begin{aligned} M_{I\times I}x_I + M_{I\times K}x_K &= 0, \\ M_{J\times J}x_J + M_{J\times K}x_K &= 0. \end{aligned} \tag{1}$$

Let z be an eigenvector of M with negative eigenvalue. By the Perron-Frobenius theorem we may assume $z > 0$. Let

$$\lambda := \frac{z_I^T x_I}{z_J^T x_J}. \tag{2}$$

Define $y \in \mathbb{R}^n$ by: $y_i := x_i$ if $i \in I$, $y_i := -\lambda x_i$ if $i \in J$, and $x_i := 0$ if $i \notin I \cup J$. By (2), $z^T y = z_I^T x_I - \lambda z_J^T x_J = 0$. Moreover, one has (since $m_{i,j} = 0$ if $i \in I$ and $j \in J$):

$$\begin{aligned} y^T My = y_I^T M_{I\times I} y_I + y_J^T M_{J\times J} y_J = \\ x_I^T M_{I\times I} x_I + \lambda^2 x_J^T M_{J\times J} x_J = \\ - x_I^T M_{I\times K} x_K - \lambda^2 x_J^T M_{J\times K} x_K \leq 0 \end{aligned} \tag{3}$$

(using (1)), since $M_{I\times K}$ and $M_{J\times K}$ are nonpositive, and since $x_I > 0$, $x_J > 0$ and $x_K < 0$.

Now $z^T y = 0$ and $y^T My \leq 0$ imply that $My = 0$ (as M is symmetric and has exactly one negative eigenvalue, with eigenvector z). Therefore, $y \in \ker(M)$.

People familiar with differential geometry will see that this theorem is analogous to the Courant nodal theorem [7].

We say that a vector $x \in \ker(M)$ *has minimal support* if x is nonzero and if for each nonzero vector $y \in \ker(M)$ with $\text{supp}(y) \subseteq \text{supp}(x)$ one has $\text{supp}(y) = \text{supp}(x)$. We have the following consequence of Theorem 2.1.

Corollary 2.1a *Let G be a connected graph and let $M \in \mathcal{O}_G$. Let $x \in \ker(M)$ have minimal support. Then $G[\text{supp}_+(x)]$ and $G[\text{supp}_-(x)]$ are connected.*

A graph G is called *planar* if the graph can be drawn in the plane such that no two edges cross. A graph G is called *outerplanar* if the suspension on G is planar, that is, G is planar and all vertices of G are incident with one face.

Theorem 2.2 *Let G be a path. Then each $M \in \mathcal{O}_G$ has* $\text{corank}(M) \leq 1$.

Theorem 2.3 *Let G be a 2-connected outerplanar graph. Then each $M \in \mathcal{M}$ has* $\text{corank}(M) \leq 2$.

Theorem 2.4 *Let G be a 3-connected planar graph. Then for each $M \in \mathcal{O}_G$,* corank(M) ≤ 3.

Proof. We assume that G is embedded into the plane. Let v_1 be a vertex of G. Let v_2 and v_3 be vertices of G which are adjacent to v_1 and such that v_1, v_2, v_3 are incident with the same face F of G. Assume that there exists a matrix $M \in \mathcal{O}_G$ with corank(M) > 3. Let $x \in \ker(M)$ be a nonzero vector with $x_{v_i} = 0$ for $i = 1, 2, 3$. We may assume that x has minimal support.

Since G is 3-connected there exists 3 pairwise disjoint paths P_1, P_2, P_3, where each P_i starts in a vertex $w_i \notin$ supp(x) adjacent to at least one vertex in supp(x), and ends in v_i. Each vertex $v \notin$ supp(x) adjacent to some vertex in $\text{supp}_+(x)$ is also adjacent to some vertex in $\text{supp}_-(x)$ and conversely. So each w_i is adjacent to at least one vertex in $\text{supp}_+(x)$ and at least one vertex in $\text{supp}_-(x)$.

By Corollary 2.1a, $\text{supp}_+(x)$ and $\text{supp}_-(x)$ can be contracted to one vertex each. Delete all vertices of G not contained in supp(x) or in any P_i and contract each P_i to one vertex. Add a vertex in the face F and edges from v_1, v_2, v_3 to this new vertex. The resulting graphs is still planar. But this graphs contains a $K_{3,3}$-minor, a contradiction, hence we have corank(M) ≤ 3.

A graph is called *flat* if the graph can be embedded into 3-space such that for each circuit C of the graph there exists a homeomorph of an open disc Δ with boundary the circuit C, such that Δ is disjoint from G. For flat graphs we have the following theorem due to L. Lovasz and A. Schrijver[12].

Theorem 2.5 *Let G be a 4-connected flat graph. Then for each $M \in \mathcal{O}_G$,* corank(M) ≤ 4.

Some remarks are needed here. First Theorem 2.3 is not true if we remove the condition that G is a 2-connected graph. In fact corank(M) can be arbitrary high as the following example shows. Take $G := K_{1,n}$, the bipartite graph with one vertex of degree n and n vertices of degree 1. Let $M := (m_{i,j})$ be the matrix with $m_{i,j} = -1$ if ij is an edge, $m_{i,i} = 1$ if i is the vertex of degree n, and $m_{i,i} = 0$ if i is a vertex of degree 1. Then $M \in \mathcal{O}_G$, as M has only one negative eigenvalue of mulitplicity 1. However corank(M) = n − 1. The same applies to Theorem 2.4. For this, look to the graph $G := K_{2,n}$.

Secondly, are the reverse statements of Theorem 2.3 and Theorem 2.4 true? More precisely, if G is not outerplanar, does there exist a matrix $M \in \mathcal{O}_G$ with corank(M) > 2, and, if G is not planar, does there exist a matrix $M \in \mathcal{O}_G$ with corank(M) > 3?

Let us first look to the second remark. In [5, 6, 4], Y. Colin de Verdière defined a property, which he called the *Strong Arnol'd Hypothesis*[1], for operators. To state this property we must introduce some basic concepts of differential geometry applied to matrices.

Let $\mathcal{S}_{\setminus,\|}$ denote the submanifold of all real-valued symmetric $n \times n$ matrices with corank k, and let $T_M\mathcal{S}_{\setminus,\|}$ denote the tangent space of $\mathcal{S}_{\setminus,\|}$ at matrix M. The normal space of $\mathcal{S}_{\setminus,\|}$ at M is the space of all real-valued symmetric $n \times n$ matrices X with $MX = 0$; we denote this space by $N_M\mathcal{S}_{\setminus,\|}$. Note that we use the inner product defined by $A \cdot B = \text{Tr(AB)}$.

Let $T_M\mathcal{O}_G$ denote the tangent space of $\mathcal{O}_G$ at M. So $T_M\mathcal{O}_G$ is the space of all real-valued symmetric $n \times n$ matrices $K = (k_{i,j})$ with $k_{i,j} = 0$ if ij is not an edge and $i \neq j$. The normal space of $\mathcal{O}_G$ at M is denote by $N_M\mathcal{O}_G$, and is the space of all real-valued symmetric $n \times n$ matrices with $K = (k_{i,j})$ with $k_{i,j} = 0$ if $i = j$ and i and j are adjacent.

A matrix $M \in \mathcal{O}_G$ is said to fulfill the Strong Arnol'd Hypothesis if the linear span of $T_M\mathcal{S}_{\setminus,\|}$ and $T_M\mathcal{O}_G$ is the whole space of all real-valued symmetric $n \times n$ matrices. This is what mathematicians working in differential topology would call transversality, and is some kind of general position property, slightly pertubating the submanifolds does not change the

intersection of the two submanifold dramatically. This in contrast to the case where the two submanifold do not intersect transversally in which case it is always possible to find a small pertubation which changes the intersection of the two submanifolds dramatically.

Instead of looking to the maximum corank of any matrix $M \in \mathcal{O}_G$, we now look to the maximum corank of any matrix $M \in \mathcal{O}_G$ with the additional condition that M fulfills the Strong Arnol'd Hypothesis. This number is denoted by $\mu(G)$ and was discovered by Y. Colin de Verdière[5, 6]. So for any graph G, $\mu(G) \leq t$ if and only if corank(M) $\leq$ t for any matrix $M \in \mathcal{O}_G$ fulfilling the Strong Arnol'd Hypothesis.

It is not difficult to check whether a matrix does fulfill the Strong Arnol'd Hypothesis. The following criterion is given in [5, 6].

Proposition 2.6 *A matrix $M \in \mathcal{O}_G$ fulfills the Strong Arnol'd Hypothesis if and only if for each symmetric $n \times n$ matrix A there is a matrix $B \in T_M\mathcal{O}_G$ such that $x^TAx = x^TBx$ for each $x \in \ker(M)$.*

Here is another criterion [11].

Proposition 2.7 *A matrix $M \in \mathcal{O}_G$ fulfills the Strong Arnol'd Hypothesis if and only if there is no nonzero symmetric matrix $X = (x_{i,j})$, with $x_{i,i} = 0$ and $x_{i,j} = 0$ if i and j are adjacent in G, such that $MX = 0$.*

The matrix $M = (m_{i,j}) \in \mathcal{O}_{\mathcal{K}_{\infty,\setminus}}$ with $m_{i,j} = -1$ if ij is an edge, $m_{i,i} = 1$ if i is the vertex of degree n, and $m_{i,i} = 0$ if i is a vertex of degree one, does not fulfill the Strong Arnol'd Hypothesis if $n > 3$, as also follows from the following proposition.

Proposition 2.8 *Let G be a graph. Let $M \in \mathcal{O}_G$ fulfill the Strong Arnol'd Hypothesis, and let $x \in \ker(M)$. If $G[\mathrm{supp}_+(x)]$ has more than one component, then $G[\mathrm{supp}_+(x)]$ has only two components and $G[\mathrm{supp}_-(x)]$ has only one component.*

The fact that M does not fulfill the Strong Arnol'd Hypothesis also follows from the following important property of $\mu(G)$. Proofs of it can be found in [5, 6, 8, 10]. First we need some definitions.

Let e be an edge of $G = (V, E)$. Then the graph obtained by *deleting* e is the graph $G' := (V, E \setminus \{e\})$. If e is not a loop, then the graph obtained by *contracting* e is the graph obtained by deleting e and identifying the ends of e. A *minor* of a graph arises by a series of deletions and contractions of edges of G and deletions of isolated vertices. A minor of a graph G is called a *proper minor* if the minor is not equal to G. A class $\mathcal{C}$ of graphs is closed under taking minors and isomorphism if for each $G \in \mathcal{C}$ also all its minors belong to $\mathcal{C}$. A graph G is called a *forbidden minor* for $\mathcal{C}$ if G does not belong to $\mathcal{C}$. A graph G is called a *minimally forbidden minor* for $\mathcal{C}$ if G does not belong to $\mathcal{C}$ but each proper minor of G does belong to $\mathcal{C}$.

By the well-quasi ordering theorem of Robertson and Seymour[14], for each class $\mathcal{C}$ of graphs closed under taking minors, there exists a finite collection $\mathcal{F}$ of minimally forbidden minors for a class $\mathcal{C}$ closed under taking minors and isomorphisms.

The important property of $\mu(G)$ is

Theorem 2.9 *If G' is a minor of G then $\mu(G') \leq \mu(G)$.*

So the class of graphs G with $\mu(G) \leq t$ is closed under taking minors. Since, by definition, the class of graphs G with $\mu(G) \leq t$ is closed under taking isomorphisms, there is a finite collection $\mathcal{F}_{\sqcup}$ of minimally forbidden minors for the class of graphs with $\mu(G) > t$.

Note that, since $K_{1,n}$ is a minor of a 2-connected outerplanar graph, $\mu(K_{1,n}) \leq 2$. Thus, the matrix $M := (m_{i,j})$ with $m_{i,j} = -1$ if ij is an edge, $m_{i,i} = 1$ if i is the vertex of degree n, and $m_{i,i} = 0$ if i is a vertex of degree 1, does not fulfill the Strong Arnol'd Hypothesis.

The minimally forbidden minors for a graph being a path are $K_{1,3}$ and K_3. The minimally forbidden minors for the class of outerplanar graphs are $K_{2,3}$ and K_4. By the Wagner/Kuratowski theorem, the minimally forbidden minor for the class of planar graphs are $K_{3,3}$ and K_5.

Theorem 2.10 *$\mu(K_{1,3}) = 2$, $\mu(K_3) = 2$, $\mu(K_{2,3}) = 3$, $\mu(K_4) = 3$, $\mu(K_{3,3}) = 4$ and $\mu(K_5) = 4$.*

Since maximally outerplanar graphs (these are graphs such that addition of any edge would make the graph non-simple or not outerplanar) are 2-connected, and, since maximally planar graphs are 3-connected, we have:

Theorem 2.11 *A graph is outerplanar if and only if $\mu(G) \leq 2$.*

and

Theorem 2.12 *A graph is planar if and only if $\mu(G) \leq 3$.*

In [13], Robertson, Seymour and Thomas showed that there are seven minimally forbidden minors for the class of flat graphs. The set of these seven graphs is called the Petersen family (one of the graphs of the Petersen family is the Petersen graph).

For flat graphs we have:

Theorem 2.13 *A graph is flat if and only if $\mu(G) \leq 4$.*

It is however not true that maximally flat graphs are 4-connected. However, in [11], Van der Holst, Lovász and Schrijver showed the invariance of the Colin de Verdière parameter under clique sums, which shows that a minimal counterexample to Theorem 2.13 must be 4-connected.

A graph H is a *subgraph* of G if $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$. Let G_1 and G_2 be subgraphs of G. If $VG = VG_1 \cup VG_2$, $EG = EG_1 \cup EG_2$, and $S := VG_1 \cap VG_2$ induces a clique in G_1 and G_2, then G is called a *clique sum* of G_1 and G_2.

Let $t := \max\{\mu(G_1), \mu(G_2)\}$. For any $U \subseteq VG$, let $N(U)$ denote the set of vertices in $VG \setminus U$ that are adjacent to at least one vertex in U. Let $K_{t+3} \setminus \Delta$ denote the graph obtained from K_{t+3} by deleting the edges of a triangle in K_{t+3}. (A triangle is a subgraph isomorphic to K_3.) Then:

Theorem 2.14 *If $\mu(G) > t$, then $\mu(G) = t+1$ and we can contract two or three components of $G - S$ so that the contracted vertices together with S form a $K_{t+3} \setminus \Delta$.*

Let v be a vertex of degree 3. A graph G' is obtained from G by applying a $Y\Delta$-operation on v if G' is obtained from G by deleting vertex v and connecting the neighbours of v by edges. A graph G' is obtained from G by applying a ΔY-operation if G' is obtained from G by deleting the edges of a triangle and adding a new vertex to the graph and connecting this new vertex to all vertices of the triangle.

From Theorem 2.14 we get the following corollary:

Corollary 2.14a *Let v be a vertex of degree 3 in G. Let G' be the graph obtained from G by applying a $Y\Delta$-operation on v. Then:*

(i) If the graph, obtained from G by deleting v and its neighbours, is connected and $\mu(G) \geq 4$ then $\mu(G') \geq \mu(G)$.

(ii) If $\mu(G) \geq 5$ then $\mu(G') \geq \mu(G)$.

Another proof not using Theorem 2.14 is given by Bacher and Colin de Verdière in [2]. There they also showed:

Theorem 2.15 *Let G' be obtained from G by a ΔY-operation. Then $\mu(G') \geq \mu(G)$.*

Corollary 2.14a and Theorem 2.15 imply that the set of graphs G obtained from K_6 by applying ΔY- and $Y\Delta$-operations, all have $\mu(G) = 5$. In [13], Robertson, Seymour and Thomas showed that the minimally forbidden minors for the class of flat graphs are the set of graphs obtained from K_6 by applying ΔY- and $Y\Delta$-operations. So graphs G with $\mu(G) \leq 4$ are flat graphs. What remains open is the question whether flat graphs G have $\mu(G) \leq 4$.

3 The parameter $\lambda(G)$

Let G be a graph and let $M \in \mathcal{O}_G$ fulfill the Strong Arnol'd Hypothesis. Let $x \in \ker(M)$. Recall that by Proposition 2.8, $G[\text{supp}_+(x)]$ can have at most two components, and, if $G[\text{supp}_+(x)]$ has two components, then $G[\text{supp}_-(x)]$ has only one component. Define a new graph parameter $\lambda(G)$ as follows: $\lambda(G)$ is the largest $d \in \mathbb{N}$ for which there exists a d-dimensional subspace X of $\mathbb{R}^V$ such that:

(4) for each nonzero $x \in X$, $G[\text{supp}_+(x)]$ is a nonempty connected graph.

We will see that although $\lambda(G)$ is not always equal to $\mu(G)$, $\lambda(G)$ seems to be very close to $\mu(G)$ and in fact $\lambda(G)$ can be used to compute $\mu(G)$.

Here is an equivalent characterization of $\lambda(G)$. A subset H of $\mathbb{R}^d$ is called a *halfspace* if $H = \{x \in \mathbb{R}^d \mid c^T x > 0\}$ for some nonzero $c \in \mathbb{R}^d$. A function $\phi : V \longrightarrow \mathbb{R}^d$ is called a *valid representation* if

(5) for each halfspace H of $\mathbb{R}^d$, the set $\phi^{-1}(H)$ is nonempty and induces a connected subgraph of G.

Then $\lambda(G)$ is equal to the largest $d \in \mathbb{N}$ such that there exists a valid representation $\phi : V \longrightarrow \mathbb{R}^d$.

Theorem 3.1 *If G' is a minor of G then $\lambda(G') \leq \lambda(G)$.*

In [9], Van der Holst, Laurent and Schrijver showed:

Theorem 3.2 $\lambda(G) \leq 1$ *if and only if G is a forest.*

Theorem 3.3 $\lambda(G) \leq 2$ *if and only if G has no K_4-minor.*

Theorem 3.4 $\lambda(G) \leq 3$ *if and only if G has no K_5- and no V_8-minor.*

Since K_5 and V_8 are forbidden minors for the class of planar graphs (as $K_{3,3}$ is a minor of V_8), Theorem 3.4 tells us that $\lambda(G) \leq 3$ if G is a planar graph. Similarly, Theorem 3.3 tells us that $\lambda(G) \leq 2$ if G is an outerplanar graph.

Lovász and Schrijver[12] showed:

Theorem 3.5 *If G is a flat graph then $\lambda(G) \leq 4$.*

The proof uses a theorem on antipodal links, which can be regarded as an extension of Borsuk's antipodality theorem.

A classification of the graphs G with $\lambda(G) \leq 4$ is not known to the author.

4 Another graph parameter

Let $\mathcal{M}_G$ be the set of all real-valued symmetric matrices $M = (m_{i,j})$ with $m_{i,j} = 0$ if ij is not an edge and $i \neq j$. The Perron-Frobenius theorem does not hold for these kind of matrices; i.e. it is not always true that the multiplicity of the smallest eigenvalue is 1, as $M = (m_{i,j})$ with $m_{i,j} = 1$ for each i, j shows that the multiplicity of the smallest eigenvalue of a matrix of $\mathcal{M}_{K_\setminus}$ can be as large as $n-1$. But the question here is to characterize for each $t \in \mathbb{N}$ those graphs G for which the multiplicity of the smallest eigenvalue is at most t. By adding an appropiate diagonal matrix to M, the question becomes to characterize for each $t \in \mathbb{N}$ those graphs G for which the corank of each positive semi-definite matrix $M \in \mathcal{M}_G$ is at most t. (A *positive semi-definite* matrix M is a matrix with $x^T M x \geq 0$ for all vectors x.)

Theorem 4.1 *Let G be a tree. Then each positive semi-definite matrix $M \in \mathcal{M}_G$ has corank at most 1.*

A proof of this theorem is given in [4]. See also [8] for a proof.

Theorem 4.2 *Let G be a 2-connected graph which is the dual of an outerplanar graph. Then each positive semi-definite matrix $M \in \mathcal{M}_G$ has corank at most 2.*

See [8] for a proof.

A *k-clique tree* is a graph of the form $K_k \times T$ with T a tree. So a 1-clique tree is just a tree. Theorem 4.2 can be stated as follows.

Theorem 4.3 *Let G be a 2-connected graphs which is a minor of a 2-clique tree. Then each positive semi-definite matrix $M \in \mathcal{M}_G$ has corank at most 2.*

Also for matrices $M \in \mathcal{M}_G$ there is some kind of Courant nodal theorem.

Theorem 4.4 *Let $M \in \mathcal{M}_G$ be positive semi-definite. Let $x \in \ker(M)$ have minimal support. Then $G[\mathrm{supp(x)}]$ is connected.*

Without the condition that G is 2-connected, Theorem 4.2 is not true. Also for matrices $M \in \mathcal{M}_G$ we can define the notion of Strong Arnol'd Hypothesis. In [3], Colin de Verdière introduced the parameter $\nu(G)$ which he defined as the largest corank of any matrix $M \in \mathcal{M}_G$ fulfilling the Strong Arnol'd Hypothesis. For $\nu(G)$ we have:

Theorem 4.5 *$\nu(G) \leq 1$ if and only if G is a forest, that is, if G is a minor of a 1-clique tree.*

Theorem 4.6 *$\nu(G) \leq 2$ if and only if G is a minor of a 2-clique tree.*

For $\nu(G) \leq 3$, it is not true that G is a minor of a 3-clique tree, as V_8 is not a minor of a 3-clique tree.

References

[1] V.I. Arnol'd. Modes and quasimodes. *J. Funct. Anal.*, 6:94–101, 1972.

[2] R. Bacher and Y. Colin de Verdière. Multiplicités des valeurs propres et transformations étoile-triangle des graphes. preprint.

[3] Y. Colin de Verdière. Discrete magnetic Schrödinger operators and tree-width of graph. preprint.

[4] Y. Colin de Verdière. Sur une hypothèse de transversalité d'Arnold. *Comment. Math. Helv.*, 63:184–193, 1988.

[5] Y. Colin de Verdière. Sur un nouvel invariant des graphes et un critère de planarité. *J. Comb. Theory, Ser. B.*, 50:11–21, 1990.

[6] Y. Colin de Verdière. On a new graph invariant and a criterion of planarity. In N. Robertson and P. Seymour, editors, *Graph Structure Theory*, volume 147 of *Contemporary Mathematics*, pages 137–147. American Mathematical Society, Providence, Rhode Island, 1993.

[7] R. Courant and D. Hilbert. *Methods of Mathematical Physics*, volume 1. Interscience, New York, 1953.

[8] H van der Holst. *Topological and Spectral Graph Characterizations.* PhD thesis, University of Amsterdam, 1996.

[9] H. van der Holst, M. Laurent, and A. Schrijver. On a minor-monotone graph invariant. *J. Comb. Theory, Ser. B.*, 65:291–304, 1995.

[10] H. van der Holst, L. Lovász, and A. Schrijver. The Colin de Verdière graph parameter. preprint.

[11] H. van der Holst, L. Lovász, and A. Schrijver. On the invariance of Colin de Verdière's graph parameter under clique sums. *Linear Algebra Appl.*, 226-228:509–517, 1995.

[12] L. Lovász and A. Schrijver. The Colin de Verdière number of linklessly embeddable graphs. preprint.

[13] N. Robertson, P. Seymour, and R. Thomas. Sachs' linkless embedding conjecture. *J. Comb. Theory, Ser. B.*, 64:185–227, 1995.

[14] N. Robertson and P.D. Seymour. Graph minors. XV. Wagner's conjecture. preprint, 1988.

Scheduling with target start times

J.A. Hoogeveen
Department of Mathematics and Computing Science
Eindhoven University of Technology
E-mail: slam@win.tue.nl

S.L. van de Velde
Department of Mechanical Engineering
University of Twente
E-mail: s.l.vandevelde@wb.utwente.nl

Abstract

We address the single-machine problem of scheduling n independent jobs subject to target start times. Target start times are essentially release times that may be violated at a certain cost. The goal is to minimize an objective function that is composed of total completion time and maximum promptness, which measures the observance of these target start times. We show that in case of a linear objective function the problem is solvable in $O(n^4)$ time if preemption is allowed or if total completion time outweighs maximum promptness.

1 Introduction

A production company has to deal with the traditional conflict between internal and external efficiency of the production. *Internal efficiency* is the efficient use of the scarce resources. It results in a cost reduction and hence in possibly more competitive prices or higher profits. *External efficiency* is achieved by meeting the conditions superimposed by external relations. Clients, for instance, insist on product quality, short delivery times, and in-time delivery, among other things. Compromising product quality is playing with fire, but many a company tries to get away with late deliveries. After all, a good due-date performance may be achieved only in case of putting work out, overwork, frequent setups, or high setup costs. Unfortunately, many companies do not realize that a better planning may accomplish the same. This type of external efficiency, between the company and its clients, is actually *downstream*; it is the extent by which the company successfully copes with the requirements on the *demand* side.

We also distinguish *upstream* external efficiency. This is the extent by which the company successfully copes with the conditions on the *supply* side. A company, for instance, negotiates on the prices and delivery times of raw material. In order to achieve a higher internal efficiency, but especially a better due date performance, it may be worthwhile to pay a higher price to get the raw material sooner.

There exist several single-machine scheduling models of the trade-off between internal and downstream external efficiency. Van Wassenhove and Gelders (1980), for instance,

consider a model for making the trade-off between work-in-process inventories and due date performance; see also Hoogeveen and Van de Velde (1995). Schutten, Van de Velde, and Zijm (1996) consider a batching problem for balancing out utilizing machine capacity against due date performance. Single-machine problems seem to be oversimplified models, but the study of these models makes sense, if we think of a company as a single-machine shop, or if there is a single bottleneck. What is more, single-machine models serve as building-blocks for solving complex scheduling problems.

In this paper, we study a single-machine scheduling model for striking a rational balance between internal and upstream external efficiency. Our model specification is as follows. A set of n independent jobs has to be scheduled on a single machine that is continuously available from time zero onwards and that can process at most one job at a time. Each job J_j ($j = 1, \ldots, n$) requires processing during a positive time p_j and has a target start time s_j. Without loss of generality, we assume that the processing times and target start times are integral. A *schedule* σ specifies for each job when it is executed while observing the machine availability constraints; hence, a schedule σ defines for each job J_j its start time $S_j(\sigma)$ and its completion time $C_j(\sigma)$. The *promptness* $P_j(\sigma)$ of job J_j is defined as $P_j(\sigma) = s_j - S_j(\sigma)$, and the maximum promptness is defined as $P_{\max}(\sigma) = \max_{1 \leq j \leq n} P_j(\sigma)$. We note that the maximum promptness $P_{\max}(\sigma)$ equals the *maximum earliness* $E_{\max}(\sigma) = \max_{1 \leq j \leq n}(d_j - C_j(\sigma))$ if each J_j has a due date d_j for which $s_j = d_j - p_j$ and if interruption of job processing is not allowed.

The problem we consider is to schedule the jobs so as to minimize total completion time $\sum_{j=1}^{n} C_j$ and maximum promptness $P_{\max}$ simultaneously. Total completion time $\sum_{j=1}^{n} C_j$ is a measure of the work-in-process inventories as well as the average leadtime. Hence, it is a performance measure for internal efficiency as well as downstream external efficiency.

Maximum promptness measures the observance of target start times. If it is positive, then it signals an inefficiency: at least one job is scheduled to start before its target start time. Generally, this is possible only if we are willing to pay a penalty. In case the target start times are derived from the delivery times of raw material, then this penalty is actually the price of a speedier delivery. In case the target start times are derived from the completion times of the parts in the preceding production stage, then this penalty may be an overwork bonus to expedite the production. If the maximum promptness is negative, then it signals a slack, which implies that we may increase the deadlines that are used in the preceding production stage.

It is important to realize that the target start times are actually *release times* that may be violated at a certain cost. In this sense, our problem comes close to the well-studied single-machine problem of minimizing total completion time subject to release times; see for instance Lenstra, Rinnooy Kan, and Brucker (1977) and Ahmadi and Bagchi (1990).

We now give a formal specification of our objective function. We associate with each schedule σ a point $(\sum_{j=1}^{n} C_j(\sigma), P_{\max}(\sigma))$ in $\Re^2$ and a value $F(\sum_{j=1}^{n} C_j(\sigma), P_{\max}(\sigma))$. The function $F : \Omega \rightarrow \Re$, where Ω denotes the set of all feasible schedules, is a given composite objective function that is nondecreasing in either of its arguments; this implies that for any two schedules σ and π with $\sum_{j=1}^{n} C_j(\sigma) \leq \sum_{j=1}^{n} C_j(\pi)$ and $P_{\max}(\sigma) \leq P_{\max}(\pi)$ we have that $F(\sum_{j=1}^{n} C_j(\sigma), P_{\max}(\sigma)) \leq F(\sum_{j=1}^{n} C_j(\pi), P_{\max}(\pi))$. Our problem is then formulated as

$$\min\{F(\sum_{j=1}^{n} C_j(\sigma), P_{\max}(\sigma)) \mid \sigma \in \Omega\}.$$

Extending the three-field notation scheme of Graham, Lawler, Lenstra, and Rinnooy Kan (1979), we denote this problem by $1||F(\sum_{j=1}^{n} C_j, P_{\max})$. The special case in which the function F is linear is denoted by $1||\alpha_1 \sum_{j=1}^{n} C_j + \alpha_2 P_{\max}$, where $\alpha_1 \geq 0$ and $\alpha_2 \geq 0$.

In comparison to single-criterion problems, there are few papers on multicriteria scheduling problems. We refer to Dileepan and Sen (1988) and Hoogeveen (1992) for an overview of problems, polynomial algorithms, and complexity results.

This paper is organized as follows. In Section 2, we make some general observations and outline a generic strategy for solving the $1||F(\sum_{j=1}^{n} C_j, P_{\max})$ problem. We also point out that $1||F(\sum_{j=1}^{n} C_j, P_{\max})$ as well as its preemptive version $1|pmtn|F(\sum_{j=1}^{n} C_j, P_{\max})$, in which jobs may be interrupted and resumed later on, are $\mathcal{NP}$-hard in the strong sense. In Section 3, we consider the linear variant $1|pmtn|\alpha_1 \sum_{j=1}^{n} C_j + \alpha_2 P_{\max}$. Our main results are that $1|pmtn|\alpha_1 \sum_{j=1}^{n} C_j + \alpha_2 P_{\max}$ and, in the case that $\alpha_1 \geq \alpha_2$, also $1||\alpha_1 \sum_{j=1}^{n} C_j + \alpha_2 P_{\max}$ are solvable in $O(n^4)$ time.

2 General observations

The fundamental question is whether the $1||F(\sum_{j=1}^{n} C_j, P_{\max})$ problem is solvable in polynomial time for any given function F that is nondecreasing in its arguments. The first observation we make is that this is so, if we can identify all the so-called *Pareto optimal* schedules in polynomial time.

Definition 1 *A schedule $\sigma \in \Omega$ is Pareto optimal with respect to the objective functions $(\sum_{j=1}^{n} C_j, P_{\max})$ if there exists no feasible schedule π with either $\sum_{j=1}^{n} C_j(\pi) \leq \sum_{j=1}^{n} C_j(\sigma)$ and $P_{\max}(\pi) < P_{\max}(\sigma)$, or $\sum_{j=1}^{n} C_j(\pi) < \sum_{j=1}^{n} C_j(\sigma)$ and $P_{\max}(\pi) \leq P_{\max}(\sigma)$.*

Once the *Pareto optimal set*, that is, the set of all schedules that are Pareto optimal with respect to the functions $(\sum_{j=1}^{n} C_j, P_{\max})$, has been determined, the $1||F(\sum_{j=1}^{n} C_j, P_{\max})$ problem can be solved for any function F by computing the cost of each Pareto optimal point and taking the minimum. Hence, if each Pareto optimal schedule can be found in polynomial time and the number of Pareto optimal schedules is polynomially bounded, then the problem is solvable in polynomial time.

We start with analyzing the two single-criterion problems that are embedded within $1||F(\sum_{j=1}^{n} C_j, P_{\max})$, that is, $1||P_{\max}$ and $1||\sum_{j=1}^{n} C_j$. The $1||P_{\max}$ problem is clearly meaningless, since we can improve upon each solution by inserting extra idle time at the beginning of the schedule. Hence, we impose the restriction that machine idle time before the processing of any job is prohibited, that is, all jobs are to be scheduled in the interval $[0, \sum_{j=1}^{n} p_j]$. It is easily checked that in case of a given overall deadline $D > \sum_{j=1}^{n} p_j$ the optimal schedule is obtained by inserting $D - \sum_{j=1}^{n} p_j$ units of idle time before the start of the first job. In the three-field notation scheme, the no machine idle time constraint is denoted by the acronym *nmit* in the second field. The $1|nmit|P_{\max}$ problem is solved by sequencing the jobs in order of *non-decreasing target start times* s_j. The $1||\sum_{j=1}^{n} C_j$ problem is solved by sequencing the jobs in order of non-decreasing processing times p_j (Smith, 1956). Let now $MTST$ be an optimal schedule for the $1|nmit|P_{\max}$ problem in which ties are settled to minimize total completion time; $MTST$ is the abbreviation of minimum target start time. In addition, let SPT be an optimal schedule for the $1||\sum_{j=1}^{n} C_j$ problem, in which ties are settled to minimize maximum promptness; SPT is the abbreviation of shortest processing time. It then follows that $P^*_{\max} \leq P_{\max}(\sigma) \leq P_{\max}(SPT)$ and $\sum_{j=1}^{n} C^*_j \leq \sum_{j=1}^{n} C_j(\sigma) \leq \sum_{j=1}^{n} C_j(MTST)$ for any Pareto optimal schedule σ, where $P_{\max}$ and $\sum_{j=1}^{n} C^*_j$ denote the outcome of the respective single-criterion problems.

Consider any Pareto optimal schedule σ; let $(P_{\max}(\sigma), \sum_{j=1}^{n} C_j(\sigma))$ be the corresponding Pareto optimal point. By definition, σ solves the problems $1|P_{\max} \leq P_{\max}(\sigma)|\sum_{j=1}^{n} C_j$ and

$1|\sum_{j=1}^{n} C_j \leq \sum_{j=1}^{n} C_j(\sigma)|P_{\max}$; the notation $P_{\max} \leq P_{\max}(\sigma)$ in the second field means that we impose $P_{\max} \leq P_{\max}(\sigma)$ as an extra constraint that each feasible schedule has to satisfy. Hence, if we know some $P_{\max}$ value P that may correspond to a Pareto optimal point, then we can determine the corresponding schedule σ and $\sum_{j=1}^{n} C_j$ value by solving $1|P_{\max} \leq P|\sum_{j=1}^{n} C_j$. Since any given value $P_{\max}$ induces for each job J_j a release date $r_j = \max\{0, s_j - P_{\max}\}$, we have to solve a problem of the form $1|r_j|\sum_{j=1}^{n} C_j$. A generic strategy for solving the bicriteria problem is then to solve this type of problem for all $P_{\max}$ values that may correspond to a Pareto optimal point and evaluate the function F for all the resulting combinations $(P_{\max}, \sum_{j=1}^{n} C_j)$. Lenstra, Rinnooy Kan, and Brucker (1977), however, show that the $1|r_j|\sum_{j=1}^{n} C_j$ problem is $\mathcal{NP}$-hard in the strong sense.

We therefore make the additional assumption that preemption of jobs is allowed, that is, the execution of any job may be interrupted and resumed later on. This assumption implies a crucial relaxation of the original problem; it has both positive and negative aspects. To start with the positive part: we can apply the generic approach now, since the $1|pmtn, r_j|\sum_{j=1}^{n} C_j$ problem is solvable in $O(n \log n)$ time by Baker's algorithm (Baker, 1974): *always keep the machine assigned to the available job with minimum remaining processing time.* Note that this algorithm always generates a schedule without machine idle time if $P_{\max} \geq P^*_{\max}$. The disadvantage is that we lose the equivalence that existed between the maximum promptness criterion and the maximum earliness criterion in case $s_j = d_j - p_j$. This is so, since a given value $E_{\max}$ induces an earliest completion time for each job, not a release date.

Another crucial issue with respect to the applicability of the generic approach concerns the number of Pareto optimal points. Unfortunately, this number can grow arbitrarily large in general, since each value $P_{\max} \leq P_{\max}(SPT)$ corresponds to a Pareto optimal point, as we are allowed to preempt at any point in time, not just at the integral points. Seemingly, this is another disadvantage of allowing preemption, but this problem complicates the nonpreemptive version as well, since idle time can be inserted in any amount. The above implies that we can obtain a series of 2^n consecutive Pareto optimal points with $P_{\max}$ values that are multiples of 2^{-n}. Using the result by Schrijver (see Hoogeveen, 1996) that the problem of minimizing an arbitrary function $F(x, y)$ that is nondecreasing in both arguments over 2^n consecutive integral y values is $\mathcal{NP}$-hard in the strong sense, we conclude that $1|pmtn|F(\sum_{j=1}^{n} C_j, P_{\max})$ and $1||F(\sum_{j=1}^{n} C_j, P_{\max})$ are $\mathcal{NP}$-hard in the strong sense.

3 The linear variant $1|pmtn|\alpha_1 \sum_{j=1}^{n} C_j + \alpha_2 P_{\max}$

To deal with this infinite number of Pareto optimal points, we assume from now on that the composite objective function is linear; we can then limit ourselves to the subset of the set of Pareto optimal schedules that contains an optimal solution to the $1|pmtn|\alpha_1 \sum_{j=1}^{n} C_j + \alpha_2 P_{\max}$ problem for *any* $\alpha_1 \geq 0$ and $\alpha_2 \geq 0$. We define this set as the set of *extreme* schedules.

Definition 2 *A schedule $\sigma \in \Omega$ is extreme with respect to $(\sum_{j=1}^{n} C_j, P_{\max})$ if it corresponds to a vertex of the lower envelope of the Pareto optimal set for $(\sum_{j=1}^{n} C_j, P_{\max})$.*

If the extreme set can be found in polynomial time and if its cardinality is polynomially bounded, then the $1||\alpha_1 \sum_{j=1}^{n} C_j + \alpha_2 P_{\max}$ problem is solved in polynomial time by computing the cost of each extreme point and taking the minimum.

We start by analyzing the special case in which machine idle time before the processing of any job is prohibited; we later show that any instance of the general problem can be dealt with by reformulating it as an instance of the problem with no machine idle time allowed.

3.1 No machine idle time allowed

Recall that if machine idle time is not allowed, then all jobs are processed in the interval $[0, \sum_{j=1}^{n} p_j]$. Hence, we only have to consider $P_{\max}$ values in the interval $[P^*_{\max}, P_{\max}(SPT)]$, and for each $P_{\max}$ value P in this interval, Baker's algorithm provides an optimal schedule for the corresponding $1|r_j, pmtn|\sum_{j=1}^{n} C_j$ problem that does not contain idle time; let $\sigma(P)$ denote this schedule and let $(P, \sum_{j=1}^{n} C_j(\sigma(P)))$ denote the point in $\Re^2$ corresponding to it.

The problem is of course to distinguish between an extreme schedule and an ordinary Pareto optimal schedule. By definition, the schedule $\sigma(P_{\max})$ is extreme if increasing $P_{\max}$ by some $\epsilon > 0$ yields a smaller decrease in $\sum_{j=1}^{n} C_j$ than a decrease of $P_{\max}$ by the same amount ϵ would cost.

To illustrate the impact of an increase of $P_{\max}$, consider the following two-job example with $p_1 = 10$, $p_2 = 5$, $s_1 = 0$, and $s_2 = 10$. We have that $P^*_{\max} = 0$ and the corresponding $\sum_{j=1}^{n} C_j$ value amounts to 25. If we increase $P_{\max}$, nothing happens until it becomes advantageous to preempt job 1; this is the case for $P_{\max} = 5$. Then, until $P_{\max} = 10$, we gain ϵ on $\sum_{j=1}^{n} C_j$ by increasing $P_{\max}$ by ϵ; the value $P_{\max} = 10$ allows the SPT schedule.

From this example, we conclude that a schedule can only be extreme if a *complete interchange* has occurred in $\sigma(P)$, where an interchange is defined to be a complete interchange if there are two jobs J_i and J_j such that J_i is started before J_j in $\sigma(P - \epsilon)$, whereas J_j is started before J_i in $\sigma(P)$.

Lemma 1 *If $P > P^*_{\max}$, then the point $(P, \sum_{j=1}^{n} C_j(\sigma(P)))$ can be extreme only if a complete interchange has occurred in $\sigma(P)$.* □

The next step is to determine the $P_{\max}$ values P such that their corresponding points $(P, \sum_{j=1}^{n} C_j(\sigma(P)))$ satisfy this necessary condition. Given a pair of jobs J_i and J_j with $p_i > p_j$ and J_i started before J_j in $\sigma(P)$, we have to increase the upper bound on $P_{\max}$ such that J_j can start at time $S_i(\sigma(P))$. This will lead to a complete interchange of J_i and J_j in $\sigma(P^1)$, unless J_i itself is started at an earlier time in the schedule $\sigma(P^1)$, where $P^1 = s_j - S_i(\sigma(P))$ is the value of the upper bound on $P_{\max}$ that makes J_j available at time $S_i(\sigma(P))$. It is not possible to determine beforehand whether J_i gets started earlier when the upper bound on $P_{\max}$ is increased from P to P^1 J_i, except for one situation: J_i is executed between the start and completion time of a preemptive job J_k. In that case, increasing the upper bound on $P_{\max}$ will first lead to a uniform shift forward of J_i and J_j at the expense of J_k; the complete interchange of J_i and J_j cannot take place before a complete interchange has taken place between J_k and both J_i and J_j.

Algorithm I exploits these observations to generate each point $(P, \sum_{j=1}^{n} C_j(\sigma(P_{\max}))$ for which a complete interchange in $\sigma(P)$ may take place. The variable a_j $(j = 1, \ldots, n)$ signifies the increase of the current $P_{\max}$ value necessary to let a complete interchange involving J_j and some successor take place.

Algorithm I

Step 0. Let $P = P^*_{\max}$.

Step 1. Let $T \leftarrow 0$ and $a_j \leftarrow \infty$ for $j = 1, \ldots, n$; determine $\sigma(P)$ through Baker's rule.

Step 2. Let J_k be the job that starts at time T in $\sigma(P)$. Consider the following two cases:
(a) J_k is a preempted job. Then a_k is equal to the length of this portion of J_k. Set $T \leftarrow C_k(\sigma(P))$.

(b) J_k is not a preempted job. Then $a_k \leftarrow \min\{s_j - P - S_k(\sigma(P)) \mid J_j \in \mathcal{V}\}$, where $\mathcal{V}$ denotes

the set of jobs J_j for which $s_j - P > S_k(\sigma(P))$ and $p_j > p_k$. Set $T \leftarrow C_k(\sigma(P))$.

Step 3. If $T < \sum_{j=1}^{n} p_j$, then go to Step 2.

Step 4. Put $P \leftarrow \min_{1 \le j \le n} a_j + P$.

Step 5. If $P = P_{\max}(SPT)$, then stop; otherwise go to Step 1.

Theorem 1 *Algorithm I generates all $P_{\max}$ values P for which a complete interchange has taken place in the corresponding schedule $\sigma(P)$.*

Proof. Suppose that a complete interchange of the jobs J_i and J_j with $p_i > p_j$ took place in the schedule $\sigma(P)$, where P was not detected by Algorithm I. Hence, $S_i(\sigma(P_{\max}))$ must have been ignored in Step 2, which could have happened only in Step 2(a): J_i is started between the start and completion time of some preempted job J_k. This, however, conflicts with the earlier observation that the interchange of J_i and J_j has to wait until J_k has been interchanged with both J_i and J_j. □

As remarked before, the algorithm may generate too many $P_{\max}$ values P: in some of the schedules $\sigma(P)$ not a complete interchange has taken place. This is due to Step 2b. There we implicitly assumed that the part of the schedule before J_k, which was defined as the job to be interchanged, would remain scheduled before J_k, that is, that J_k itself would not be started earlier. This is not necessarily the case, however, since an increase of the upper bound on $P_{\max}$ may cause J_k to move forward at the expense of some job J_l with $p_l > p_k$, where the increase of the upper bound is not large enough to allow a complete interchange; J_k will preempt J_l then. Nevertheless, we now prove that the number of values $P_{\max}$ generated by Algorithm I is polynomially bounded, thereby establishing that $1|pmtn, nmit|\alpha_1 \sum_{j=1}^{n} C_j + \alpha_2 P_{\max}$ is polynomially solvable. We define for a given schedule σ the indicator function $\delta_{ij}(\sigma)$ as

$$\delta_{ij}(\sigma) = \begin{cases} 2, & \text{if } C_i(\sigma) \le S_j(\sigma) \text{ and } p_i > p_j, \\ 0, & \text{otherwise.} \end{cases}$$

We further define $\Delta_j(\sigma)$ as $\sum_{i=1}^{n} \delta_{ij}(\sigma)$ plus the number of preemptions of J_j, and we let $\Delta(\sigma) = \sum_{j=1}^{n} \Delta_j(\sigma)$.

Theorem 2 *Let P^1 be the $P_{\max}$ value that is found by Algorithm I when applied to $\sigma(P)$, where P is any $P_{\max}$ value determined by Algorithm I. We then have that $\Delta(\sigma(P^1)) < \Delta(\sigma(P))$.*

Proof. As explained above, one of the following three things has happened in $\sigma(P^1)$ in comparison to $\sigma(P)$:

(i) a preemption has been removed (Step 2a);

(ii) two jobs not in SPT-order have been interchanged (successful Step 2b);

(iii) a new preemption has been created (unsuccessful Step 2b).

All three cases have a negative effect on the value of Δ, as is easily checked (in the third case we do create an extra preemption (effect $+1$), but this pair of jobs is no longer in the wrong order (effect -2)). Hence, we only have to show that there are no moves possible that have an overall positive effect on the value of Δ. The candidates for such a move are a switch of two jobs from SPT order to LPT order and the addition of an extra preemption. We first investigate the effect of the 'wrong' switch.

Suppose that there are two jobs J_i and J_j with $p_i > p_j$ such that J_i succeeds J_j in $\sigma(P)$, whereas the order is reversed in $\sigma(P^1)$. Since Baker's algorithm prefers J_j to J_i if both

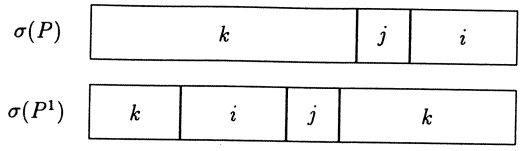


Figure 1: 'WRONG' SWITCH

jobs are available, J_i starts earlier in $\sigma(P^1)$ than J_j in $\sigma(P)$, which means that the execution of (a part of) some job J_k is postponed until J_i is completed. See Figure 1 for an illustration.

It is easily checked that we have $\Delta(\sigma(P)) = 4$ and $\Delta(\sigma(P^1)) = 3$. All we have to do is to show is that the situation depicted in Figure 1 is worst possible for this configuration. It is sufficient to prove that J_j is available at time $C_i(\sigma(P^1))$, that is, $s_j - P^1 \leq C_i(\sigma(P^1)) = s_i - P^1 + p_i$; if so, Baker's algorithm will prefer it to J_k, since the remainder of J_k has length at least equal to p_i. Hence, we have to show that $s_j \leq s_i + p_i$. As J_i did not preempt J_k in $\sigma(P)$, we must have $s_i - P + p_i \geq C_k(\sigma(P)) \geq s_j - P$, where the last inequality follows from the availability of J_j at time $C_k(\sigma(P))$. Since the smaller job is available as soon as the larger job involved in the wrong switch is completed, the increase of δ_{ij} is compensated for by the decrease of δ_{ki}. Moreover, job J_k cannot trigger a set of nested wrong switches, where we mean with a set of nested wrong switches that $\sigma(P)$ and $\sigma(P^1)$ contain the subschedules J_k, J_j, J_i, J_h and J_h, J_i, J_j, J_k with $p_j < p_i < p_h < p_k$.

Now consider the situation that the number of preemptions of a job J_k increases. Hence, there must be a job J_i with $p_i < p_k$ that succeeds J_k in $\sigma(P)$ but not in $\sigma(P^1)$, which move decreases the Δ function by one. □

Corollary 1 *If preemption is allowed, then the number of extreme schedules with respect to* $(P_{\max}, \sum_{j=1}^{n} C_j)$ *is bounded by* $n(n-1)+1$.

Proof. We have that $\Delta(\sigma) \leq n(n-1)$ for any schedule σ. Application of Theorem 2 yields the desired result. □

It is easy to construct an instance for which Algorithm I determines $O(n^2)$ different $P_{\max}$ values. We have not found an example with $O(n^2)$ extreme points yet.

Corollary 2 *The* $1|pmtn, nmit|\alpha_1 \sum_{j=1}^{n} C_j + \alpha_2 P_{\max}$ *problem is solvable in* $O(n^4)$ *time.* □

Theorem 3 *If* $\alpha_1 = \alpha_2$, *then there exists a nonpreemptive optimal schedule for the* $1|pmtn, nmit|\alpha_1 \sum_{j=1}^{n} C_j + \alpha_2 P_{\max}$ *problem. If* $\alpha_1 > \alpha_2$, *then any optimal schedule for the* $1|pmtn, nmit|\alpha_1 \sum_{j=1}^{n} C_j + \alpha_2 P_{\max}$ *problem is nonpreemptive.*

Proof. Suppose that the optimal schedule contains a preempted job. Start at time 0 and find the first preempted job J_i immediately scheduled before some nonpreempted job J_j. Consider the schedule obtained by interchanging job J_j and this portion of job J_i. If the length of the portion of job J_i is ϵ, then P_j is increased by ϵ, while C_j is decreased by ϵ. As $\alpha_1 = \alpha_2$, the interchange does not increase the objective value. The argument can be repeated until a nonpreemptive schedule remains. In case $\alpha_1 > \alpha_2$ such an interchange would decrease the objective value, contradicting the optimality of the initial schedule. □

3.2 The general case

We now drop the no machine idle time constraint. Obviously, if total completion time outweighs maximum promptness, then the insertion of machine idle time before the processing of any job makes no sense. Hence, we have the following.

Corollary 3 *If $\alpha_1 \geq \alpha_2$, then $1||\alpha_1 \sum_{j=1}^{n} C_j + \alpha_2 P_{\max}$ is solvable in $O(n^4)$ time.* □

If $\alpha_1 < \alpha_2$, then the insertion of idle time may decrease the value of the objective function. We now show that we can solve the $1|pmtn|\alpha_1 \sum_{j=1}^{n} C_j + \alpha_2 P_{\max}$ problem by using Algorithm I, which was initially designed for solving $1|nmit, pmtn|\alpha_1 \sum_{j=1}^{n} C_j + \alpha_2 P_{\max}$.

Suppose that α_1 and α_2 are given. Define $q = \alpha_2/\alpha_1$. If $q > n$, then it is always advantageous to decrease $P_{\max}$, which implies that the execution of the first job will be delayed for ever and ever. To prevent unbounded solutions, we therefore assume that $q \leq n$. A straightforward computation then shows that in any optimal schedule at least $\lfloor n - q + 1 \rfloor$ jobs are scheduled before the first incidence of idle time. The smallest value $P_{\max}(q)$ for maximum promptness that leads to such a schedule is readily obtained. Moreover, no optimal schedule with $P_{\max} \geq P_{\max}^*$ contains idle time. Therefore, we need to consider the case $P_{\max}(q) \leq P_{\max} \leq P_{\max}^*$ only.

Consider any instance $\mathcal{I}$ of $1|pmtn|\alpha_1 \sum_{j=1}^{n} C_j + \alpha_2 P_{\max}$; let $\sigma(P_{\max})$ denote any optimal schedule for $\mathcal{I}$ of $1|r_j, pmtn|\sum C_j$ for any $P_{\max}$ with $P_{\max}(q) \leq P_{\max} \leq P_{\max}^*$ and $r_j = \max\{0, s_j - P_{\max}\}$.

We create a very large job J_0 that is available from time 0 onwards to saturate $\sigma(P_{\max})$ by filling in J_0 in the periods of idle time. In fact, J_0 is so large that Baker's rule prefers each job in $\mathcal{I}$ to it; the choices $s_0 = P_{\max}(q)$ and $p_0 = P_{\max}^* - P_{\max}(q) + \max_{1 \leq j \leq n} p_j + 1$ ensure such a saturation for any $P_{\max}(q) \leq P_{\max} \leq P_{\max}^*$. Let $\mathcal{I}'$ denote the instance $\mathcal{I}$ to which J_0 is added. Due to the choice of p_0 and s_0, we have that no optimal schedule for the instance I' of $1|nmit, pmtn|\alpha_1 \sum_{j=1}^{n} + \alpha_2 P_{\max}$ contains machine idle time, and moreover, that by simply removing J_0 and leaving the rest of the schedule intact we obtain an optimal schedule for the original instance $\mathcal{I}$ of $1|pmtn|\alpha_1 \sum_{j=1}^{n} + \alpha_2 P_{\max}$. After all, we have that $C_0 = \sum_{j=0}^{n} p_j$ and that $P_0 < P_{\max}$ for any value of $P_{\max}$. Hence, instead of solving $1|pmtn|\alpha_1 \sum_{j=1}^{n} C_j + \alpha_2 P_{\max}$ for $\mathcal{I}$, we solve $1|nmit, pmtn|\alpha_1 \sum_{j=0}^{n} C_j + \alpha_2 P_{\max}$ for $\mathcal{I}'$. This approach provides us with the extreme points for $(\sum_{j=1}^{n} C_j, P_{\max})$ with $P_{\max}(q) \leq P_{\max} \leq P_{\max}^*$. If q is unknown, then we obtain all bounded extreme points by running the above procedure with $q = n$; this choice of q corresponds to the smallest value $P_{\max}(q)$ that may correspond to a bounded extreme point.

As the number of extreme points is at most equal to $n(n+1)+1$ (we have $n+1$ jobs now), and as each $P_{\max}$ value that corresponds to an extreme point is determined by Algorithm I, the $1|pmtn|\alpha_1 \sum_{j=1}^{n} C_j + \alpha_2 P_{\max}$ problem is solved in $O(n^4)$ time.

Finally, we wish to mention two important special cases of our problem. These are the case that promptness is assumed to be nonnegative, that is, $P_j = \max\{s_j - S_j, 0\}$, and the case that there is a given externally determined upper bound on $P_{\max}$. Either case can be dealt with by simply adjusting the objective function, and our algorithm can be used to solve the problem after the boundary points have been determined.

References

[1] R. Ahmadi, U. Bagchi (1990). Lower bounds for single-machine scheduling problems. *Naval Res. Log. Quart.* 37, 967-979.

[2] K.R. BAKER (1974). *Introduction to Sequencing and Scheduling*, Wiley, New York.

[3] P. DILEEPAN AND T. SEN (1988). Bicriterion static scheduling research for a single machine. *Omega 16*, 53-59.

[4] M.R. GAREY AND D.S. JOHNSON (1979). *Computers and Intractability: a Guide to the Theory of NP-Completeness*. Freeman, San Francisco.

[5] R.L. GRAHAM, E.L. LAWLER, J.K. LENSTRA, AND A.H.G. RINNOOY KAN (1979). Optimization and approximation in deterministic sequencing and scheduling: a survey. *Annals of Discrete Mathematics 5*, 287-326.

[6] J.A. HOOGEVEEN (1992). *Single-machine bicriteria scheduling*, PhD Thesis, CWI, Amsterdam.

[7] J.A. HOOGEVEEN (1996). Minimizing maximum promptness and maximum lateness on a single machine. *Mathematics of Operations Research 21*, 100-114.

[8] J.A. HOOGEVEEN AND S.L. VAN DE VELDE (1995). Minimizing total completion and maximum cost simultaneously is solvable in polynomial time, *Operations Research Letters 17*, 205-208.

[9] J.K. LENSTRA, A.H.G. RINNOOY KAN, AND P. BRUCKER (1977). Complexity of machine scheduling problems. *Annals of Discrete Mathematics 1*, 343-362.

[10] M. SCHUTTEN, S.L. VAN DE VELDE, AND W.H.M. ZIJM (1996). Single-machine scheduling with release dates, due date, and family setup times, *Management Science* 42, 1165-1174.

[11] W.E. SMITH (1956). Various optimizers for single-stage production. *Naval Research Logistics Quarterly 1*, 59-66.

[12] L.N. VAN WASSENHOVE AND F. GELDERS (1980). Solving a bicriterion scheduling problem. *European Journal of Operational Research 4*, 42-48.

A facet preserving extension of the Traveling Salesman polytope

Jeroen de Kort* and Ton Volgenant**
Faculty of Economics and Econometrics
Institute of Actuarial Science and Econometrics
University of Amsterdam
E-mail: jeroen.de.kort@pa-consulting.com tonv@fee.uva.nl

Abstract

We consider an extension of the Traveling Salesman Problem (TSP), for which 2 edge-disjoint Hamiltonian cycles of minimum total length are required. The problem is denoted as the Peripatetic Salesman Problem (PSP). The associated polytope can be seen as the union of two Traveling Salesman polytopes, with an additional edge-disjointness constraint for each edge in the graph under consideration. We give necessary and sufficient conditions under which a facet-inducing inequality for the TSP polytope can be lifted to a facet-inducing inequality for the PSP polytope. As almost all facet-inducing inequalities for the TSP polytope that are known to date satisfy these conditions, a large family of facet-inducing inequalities for the PSP polytope is determined at once. Furthermore, the dimension of the polytope is derived as well as the facet-inducing property of the edge-disjointness constraints.

1 Introduction

Consider the problem how to determine K edge-disjoint Hamiltonian cycles of minimum total length. The problem was first mentioned by Krarup (1975) who gave it the name *K-Peripatetic Salesman Problem* (K-PSP). It is not surprising that already for K = 2 the K-PSP is *NP*-hard.

Applications of the K-PSP arise in the area of data communication networks, where one wants to increase the reliability of the network while minimizing total costs. Whereas a common application of the Traveling Salesman Problem is to connect the components on a chip by a minimum costs cycle, the K-PSP connects these components by K edge-disjoint cycles, thereby reducing the potential damage in case of a link failure.

Furthermore consider a working floor in a facility were several manned or unmanned vehicles have to pickup and deliver goods at prespecified points, while at the same time the total traveled distance has to be minimized. If the routes of all vehicles start and finish at a fixed point and if pairwise disjointness of the routes is required, for instance to prevent collisions, then the problem can be modeled as a K-PSP.

We will focus on the 2-PSP, simply denoted as the PSP. A branch and bound algorithm

* Current address: PA Consulting Group, Beneluxlaan 905, 3526 KK Utrecht
** Corresponding author

for it is given by De Kort (1993), and solvable cases are given by De Brey and Volgenant (1996). Clearly, any solution to the PSP can be seen as two TSP solutions coupled by edge-disjointness constraints. We will exploit this relationship and the extensive research on the TSP polytope to derive interesting results for the PSP polytope. So we can consider the analysis to come as a special case of the more general situation: when we couple integer polyhedra, what can we say about the polyhedra that arise. Balakrishnan, Magnanti and Mirchandani (1996) consider this question in the context of so called overlay optimization problems to obtain worst case bounds of heuristics for these problem types.

In this article we will use the following *facet identification or separation* problem:

Given a vector $\mathbf{x}$ and a polytope Q, then either determine that $\mathbf{x} \in Q$ or find a facet-inducing inequality $\alpha\mathbf{x} \leq \alpha_0$ of Q such that $\alpha\mathbf{x} > \alpha_0$.

Several authors (Grötschel et al. (1985), Padberg and Rao (1981)) have independently shown that – apart from some technical details – we can optimize in polynomial time if and only if we can solve the separation problem in polynomial time for any vector $\mathbf{x}$ of appropriate dimension. Due to the *NP*-hardness of the TSP and PSP it is very unlikely that the separation problem can be solved in polynomial time for any real valued $\mathbf{x}$. With respect to the TSP however, a few classes of facet-inducing inequalities (trivial, subtour elimination, comb and clique tree inequalities) suffice to develop a branch and bound or branch and cut procedure able to solve large TSP instances, see e.g., Grötschel and Holland (1991) or Padberg and Rinaldi (1991). Much larger instances can be solved than by a branch and bound algorithm based on the 1-tree relaxation, see Held and Karp (1970).

In Section 1.1 some notation is introduced and Section 1.2 provides a brief overview of relations between the TSP and the PSP polytope. The relations are helpful to prove the main result in Section 2 consisting of two theorems that state necessary and sufficient conditions under which a facet-inducing inequality for the TSP can be lifted to a facet-inducing inequality for the PSP polytope. Almost all facet-inducing inequalities known to date for the TSP polytope satisfy the specified conditions so that a large family of facet-inducing inequalities for the PSP polytope is determined at once. In Section 3 we prove that the edge-disjointness constraints are facet-inducing for the PSP-polytope as well. Concluding remarks are given in Section 4.

1.1 Notation

The notation to be used is similar to that of several other authors see, e.g., Grötschel and Padberg (1985). Matrices and sets are denoted by capitals (A, B), vectors by bold characters ($\mathbf{a}$, $\mathbf{b}$, α, β) and scalars by conventional characters (a, b).

For the following properties we assume (A,$\mathbf{b}$) to be a minimum equation system for Q.

(1) $\dim(Q) + \text{rank}(A, \mathbf{b}) = \dim(Q) + \text{rank}(A) = m$.

(2) An inequality $\alpha\mathbf{x} \leq \alpha_0$, with $\alpha \in \mathbb{R}^m \backslash \{\mathbf{0}\}$ is valid with respect to Q if

$$Q \subseteq \{\mathbf{x} \in \mathbb{R}^m \mid \alpha\mathbf{x} \leq \alpha_0\}.$$

A valid inequality $\alpha\mathbf{x} \leq \alpha_0$ defines a *face* F of Q if $F = Q \cap \{\mathbf{x} \in \mathbb{R}^m \mid \alpha\mathbf{x} = \alpha_0\} \neq \emptyset$, and the face is *proper* if in addition $F \neq Q$. In practice one says that $\alpha\mathbf{x} \leq \alpha_0$ induces a proper face of Q whenever there exist two vectors $\mathbf{x}^1, \mathbf{x}^2 \in Q$ such that $\alpha\mathbf{x}^1 < \alpha_0$ and $\alpha\mathbf{x}^2 = \alpha_0$.

(3) A valid inequality $\alpha\mathbf{x} \leq \alpha_0$, $\alpha \in \mathbb{R}^m \backslash \{\mathbf{0}\}$, is said to be *facet-inducing* if and only if

(i) $\alpha\mathbf{x} \leq \alpha_0$ defines a proper face and there exist $\dim(Q)$ affine independent vectors in Q that satisfy $\alpha\mathbf{x} = \alpha_0$.

Or, as for instance indicated by Nemhauser and Wolsey (1988), if and only if

(iia) $\alpha\mathbf{x} \leq \alpha_0$ defines a proper face of Q and

(iib) for any inequality $\beta\mathbf{x} \leq \beta_0$ that is valid for Q and for which the following *Inclusion Property* holds:

$$\{\mathbf{x} \in Q \mid \alpha\mathbf{x} = \alpha_0\} \subseteq \{\mathbf{x} \in Q \mid \beta\mathbf{x} = \beta_0$$

there exist a vector λ of appropriate dimension and a scalar $\mu > 0$ such that

$$\beta = \lambda A + \mu\alpha \quad \text{and} \quad \beta_0 = \lambda\mathbf{b} + \mu\alpha_0.$$

Proving the facet-inducing property of an inequality using (i) is called the *direct method*, while the *indirect method* is based on (ii). In our terminology the phrase '$\alpha\mathbf{x} \leq \alpha_0$ defines a face or facet' is equivalent to 'the pair (α, α_0) defines a face or facet respectively'.

In modeling the TSP and the PSP we denote an edge between vertex i and j by $e = [i, j]$; $G = (V, E, \mathbf{c})$ represents an undirected graph, where V, with $|V| = n$, is the set of the vertices in G. The set E, with $|E| = m$, denotes the edges in G and $\mathbf{c} = (c_e)_{1\times m}$ the weights assigned to all edges. Only complete graphs will be considered, i.e., $m = 1/2n(n-1)$.

With respect to the TSP we define the incidence vector of a solution as the vector $\mathbf{y}$ with elements y_e such that $y_e = 1$ if edge e is in the solution and 0 otherwise. The set $\mathfrak{S}$ denotes the collection of the incidence vectors $\mathbf{y}$ representing a TSP solution in G.

With respect to the PSP, the two Hamiltonian cycles that represent the solution are denoted by H_k, $k \in \{1, 2\}$. For $n < 5$ the solution space is empty; for $n = 5$ only the trivial solution exists containing all edges, although there are 12 combinations of two Hamiltonian cycles for this set of edges. We denote by $\mathbf{x}(\mathbf{k})$ the incidence vector of H_k for $k = \{1, 2\}$. Then a PSP solution consists of a pair of $\{0, 1\}^m$ vectors, each corresponding with one Hamiltonian cycle. We assume $n \geq 6$. The set $\mathfrak{P}$ denotes the collection of incidence vectors $\mathbf{x} = (\mathbf{x}(\mathbf{1}), \mathbf{x}(\mathbf{2}))$ representing a PSP solution in G. Note that a single vector of dimension m is not suited to model the PSP, since it is not sufficient to point out the edges in a PSP solution. Given this set of edges in the PSP solution, that can be seen as a 4-regular graph we are faced with the determination of two Hamiltonian cycles, an *NP*-complete problem, see De Kort (1992).

1.2 Relations between the TSP and PSP polytope

If $\text{Conv}\{S\}$ denotes the convex hull of a set $S \subset \mathbb{R}^{2m}$ then the *symmetric Peripatetic Salesman polytope* is defined as $Q_P^n = \text{Conv}\{\mathbf{x} \in \{0, 1\}^{2m} \mid \mathbf{x} \in \mathfrak{P}\}$ so that the PSP is equivalent to

$$\min\{\mathbf{c}\mathbf{x}(\mathbf{1}) + \mathbf{c}\mathbf{x}(\mathbf{2}) \mid \mathbf{x} \in Q_P^n\}.$$

The degree constraints of the PSP, denoted as $A\mathbf{x}(\mathbf{k}) = \mathbf{2}$ for $k = 1, 2$, are nonredundant as the matrix

$$\begin{pmatrix} A & 0 \\ 0 & A \end{pmatrix}$$

has rank $2n$. This result enables to derive an upper bound for $\dim(Q_P^n)$:

$$Q_P^n \subseteq \{\mathbf{x} = (\mathbf{x}(1), \mathbf{x}(2)) \in \mathbb{R}^{2m} \mid A\mathbf{x}(\mathbf{k}) = \mathbf{2} \quad \text{for} \quad k = 1, 2\},$$

so that $\dim(Q_P^n) \leq 2m - 2n = n(n-3)$. De Kort (1992) showed how to construct $2(m-n)+1$ PSP solutions with the property that the corresponding incidence vectors can be used to obtain a nonsingular matrix with $2(m-n)+1$ rows and columns. This result implies that $\dim(Q_P^n) = 2(m-n)$ and that the degree constraints form a minimum equation system for Q_P^n.

For any valid inequality $\alpha(\mathbf{1})\mathbf{x}(\mathbf{1}) + \alpha(\mathbf{2})\mathbf{x}(2) \leq \alpha_0$ of Q_P^n, it can be assumed that $\alpha(k)$ is non-negative for $k \in \{1, 2\}$. (If not we lift the elements of $\alpha(\mathbf{k})$, and α_0 while maintaining the inequality.)

The *symmetric Traveling Salesman polytope* is defined as $Q_T^n = \text{Conv}\{\mathbf{y} \in \{0,1\}^m \mid \mathbf{y} \in \mathfrak{S}\}$ so that the TSP is equivalent to

$$\min\{\mathbf{cy} \mid \mathbf{y} \in Q_T^n\}.$$

Grötschel and Padberg (1979a) derived the dimension of Q_T^n : $\dim(Q_T^n) = 1/2n(n-3)$, for $n \geq 3$. As a result the degree constraints, represented as $A\mathbf{y} = \mathbf{2}$, are a minimum equation system for Q_T^n.

The following properties show the relation between Q_P^n and Q_T^n. Assume that n is large enough and that $(\alpha(\mathbf{1}), \alpha(\mathbf{2})) \geq 0$.

- If $\mathbf{x} = (\mathbf{x}(\mathbf{1}), \mathbf{x}(\mathbf{2})) \in Q_P^n$ then $\mathbf{y} = \mathbf{x}(\mathbf{k}) \in Q_T^n$ for $k \in \{1, 2\}$.
- If $\alpha(\mathbf{1})\mathbf{x}(\mathbf{1}) + \alpha(\mathbf{2})\mathbf{x}(\mathbf{2}) \leq \alpha_0$, is a valid inequality for Q_P^n then for $k \in \{1, 2\}$ there exists an $\alpha_0^k \leq \alpha_0$ such that $\alpha(\mathbf{k})\mathbf{y} \leq \alpha_0^k$ is a valid inequality for Q_T^n.
- If (α, α_0) induces a proper face or a facet of Q_T^n then $((\alpha, \mathbf{0}), \alpha_0)$ induces a proper face of Q_P^n.

An interesting question related to the latter property is: under which conditions can a facet-inducing inequality for Q_T^n be lifted to yield a facet-inducing inequality for Q_P^n? This question is the main topic of Section 2.

2 TSP facets for the PSP polytope

Two theorems will be given, each of which states necessary and sufficient conditions under which the pair $((\alpha, \mathbf{0}), \alpha_0)$ is facet-inducing for Q_P^n, given that (α, α_0) is facet-inducing for Q_T^n. The second theorem is best suited for practical purposes since the given conditions are easier to check.

Theorem 2.1

Suppose that the pair (α, α_0) is facet-inducing for Q_T^n; let $n \geq 6$.

For $k \in \{1, 2\}$, $\alpha\mathbf{x}(\mathbf{k}) \leq \alpha_0$ is facet-inducing for Q_P^n if and only if

- $\alpha\mathbf{x}(\mathbf{k}) \leq \alpha_0$ induces a proper face of Q_P^n,
- for any inequality $\beta\mathbf{x} \leq \beta_0$, $\beta = (\beta(\mathbf{1}), \beta(\mathbf{2}))$, that is valid for Q_P^n and for which the Inclusion Property holds:

$$\{\mathbf{x} \in Q_P^n \mid \alpha\mathbf{x}(\mathbf{k}) = \alpha_0\} \subseteq \{\mathbf{x} \in Q_P^n \mid \beta\mathbf{x} = \beta_0\},$$

there exist for $h = 3 - k$, a vector $(\lambda(\mathbf{k}), \lambda(\mathbf{h})) \in \mathbb{R}^{2n}$ and a positive scalar μ such that

$$\begin{aligned}(\beta(\mathbf{k}), \beta(\mathbf{h})) &= (\lambda(\mathbf{k}), \lambda(\mathbf{h})) \begin{pmatrix} A & 0 \\ 0 & A \end{pmatrix} + \mu(\alpha, \mathbf{0}) \\ \beta_0 &= \lambda(\mathbf{k})\mathbf{2} + \lambda(\mathbf{h})\mathbf{2} + \mu\alpha_0.\end{aligned} \tag{2.1}$$

Proof Follows from the indirect proof method (5 ii) stated in Section 1.1. □

Distinction has to be made between facet-inducing inequalities that satisfy and those that violate the given conditions. The facet-inducing property of (α, α_0) for Q_T^n implicates that $\alpha\mathbf{x}(\mathbf{k}) \leq \alpha_0$, $k \in \{1, 2\}$, induces a proper face of Q_P^n. The following theorem deals with the second condition.

Theorem 2.2

Suppose (α, α_0) is facet-inducing for Q_T^n. If $n \geq 7$ and $k \in \{1, 2\}$ then $\alpha\mathbf{x}(\mathbf{k}) \leq \alpha_0$ is facet-inducing for Q_P^n if and only if there does not exist a vector $\gamma \geq \mathbf{0}$, $\gamma \neq \mathbf{0}$ such that for $h = 3 - k$, $\alpha\mathbf{x}(\mathbf{k}) + \gamma\mathbf{x}(\mathbf{h}) \leq \alpha_0$ is a valid inequality for Q_P^n.

Proof Only the case $k = 1$ will be handled, as for the case $k = 2$ the proof is similar.

Let $\beta(\mathbf{1})\mathbf{x}(\mathbf{1}) + \beta(\mathbf{2})\mathbf{x}(\mathbf{2}) \leq \beta_0$ represent a valid inequality for Q_P^n. First it will be shown that if

$$\{\mathbf{x} \in Q_P^n \mid \alpha\mathbf{x}(\mathbf{1}) = \alpha_0\} \subseteq \{\mathbf{x} \in Q_P^n \mid \beta\mathbf{x} = \beta_0\},$$

a vector $\lambda(\mathbf{1}) \in \mathbb{R}^n$ exists as well as a $\mu \in \mathbb{R}^+$ and a $\beta_0^1 \in \mathbb{R}$ such that

$$(\beta(\mathbf{1}), \beta_0^1) = (\lambda(\mathbf{1})A + \mu\alpha, \lambda(\mathbf{1})\mathbf{2} + \mu\alpha_0). \tag{2.2}$$

Suppose on the contrary that for $c_1 \neq c_2$

(a) there exists a $\mathbf{y}^1 \in Q_T^n$ with $\alpha\mathbf{y}^1 = \alpha_0$, $\beta(\mathbf{1})\mathbf{y}^1 = c_1$, so that for all $\mathbf{z}^1 \in Q_T^n$, edge-disjoint from $\mathbf{y}^1$: $\beta(\mathbf{2})\mathbf{z}^1 = \beta_0 - c_1$ and

(b) there exists a $\mathbf{y}^2 \in Q_T^n$ with $\alpha\mathbf{y}^2 = \alpha_0$, $\beta(\mathbf{1})\mathbf{y}^2 = c_2$, so that for all $\mathbf{z}^2 \in Q_T^n$, edge-disjoint from $\mathbf{y}^2$: $\beta(\mathbf{2})\mathbf{z}^2 = \beta_0 - c_2$.

It can be shown (see De Kort 1992) that $\beta(\mathbf{2})$ can be written in some special form under assumption (a) and a different form under assumption (b), which leads to a contradiction. Then it follows that $\beta(\mathbf{1})$ is a combination of the degree constraints and α, which proves (2.2).

Note that the latter result can directly be obtained when a vector $\tilde{\mathbf{z}} \in Q_T^n$, edge-disjoint from $\mathbf{y}^1$ and $\mathbf{y}^2$, exists: in this case $\beta_0 - c_1 = \beta(\mathbf{2})\tilde{\mathbf{z}} \neq \beta_0 - c_2$ when assumptions (a) and (b) would both be valid.

Now suppose that only for $\hat{\gamma} \geq \mathbf{0}$, $\hat{\gamma} \neq \mathbf{0}$ a vector $\lambda(\mathbf{2}) \in \mathbb{R}^n$ exists with

$$(\beta(\mathbf{2}), \beta_0 - \beta_0^1) = (\lambda(\mathbf{2})A + \hat{\gamma}, \lambda(\mathbf{2})\mathbf{2}).$$

Theorem 2.1 indicates that this is equivalent to the assumption that $\alpha\mathbf{x}(\mathbf{1}) \leq \alpha_0$ is not facet-inducing for Q_P^n. From the validity of $\beta\mathbf{x} \leq \beta_0$ and the definition of $(\beta(\mathbf{1}), \beta(\mathbf{2}))$ it follows that $\mu\alpha\mathbf{x}(\mathbf{1}) + \hat{\gamma}\mathbf{x}(\mathbf{2}) \leq \mu\alpha_0$ is valid for Q_P^n. Defining $\gamma = \hat{\gamma}/\mu$ establishes the if-part of the proof.

To prove the theorem the other way around, assume that there is a vector $\gamma \geq \mathbf{0}$, $\gamma \neq \mathbf{0}$ and γ not a multiple of the degree constraints, such that for all $\mathbf{x} \in Q_P^n$: $\alpha\mathbf{x}(\mathbf{1}) + \gamma\mathbf{x}(\mathbf{2}) \leq \alpha_0$, then $\alpha\mathbf{x}(\mathbf{1}) = \alpha_0$ implicates $\gamma\mathbf{x}(\mathbf{2}) = 0$. Thus for all $\mathbf{x} = (\mathbf{x}(\mathbf{1}), \mathbf{x}(\mathbf{2})) \in Q_P^n$ the equation $\alpha\mathbf{x}(\mathbf{1}) = \alpha_0$ implicates:

$$(\lambda(\mathbf{1})A + \alpha)\mathbf{x}(\mathbf{1}) + (\lambda(\mathbf{2})A + \gamma)\mathbf{x}(\mathbf{2}) = \alpha_0 + (\lambda(\mathbf{1}) + \lambda(\mathbf{2}))\mathbf{2}.$$

Choosing $(\beta(\mathbf{1}), \beta_0^1) = (\lambda(\mathbf{1})A + \alpha, \lambda(\mathbf{1})\mathbf{2} + \alpha_0)$ and $(\beta(\mathbf{2}), \beta_0^2) = (\lambda(\mathbf{2})A + \gamma, \lambda(\mathbf{2})\mathbf{2})$ violates (2.1), so that $\alpha\mathbf{x}(\mathbf{1}) \leq \alpha_0$ is not facet-defining for Q_P^n. □

Theorem 2.2 holds for $n \geq 7$, as for $n = 6$ the number of edges in E is insufficient to guarantee edge-disjointness when executing the 2-opt or 3-opt operations as described. The facet-inducing property of a given class with respect to Q_T^6 can be checked by complete enumeration.

Remark 2.1 Clearly not every facet-inducing inequality of Q_T^n can be lifted to yield a facet-inducing inequality for Q_P^n. For $n \geq 6$, the constraint $y_e \leq 1$ is facet-inducing for Q_T^n. Nevertheless, $x_e(k) \leq 1$ does not define a facet for Q_P^n: choose in Theorem 2.1 $\beta\mathbf{x} \leq \beta_0$ as the edge-disjointness constraints $x_e(1) + x_e(2) \leq 1$. This inequality is valid for Q_P^n and satisfies the Inclusion Property but violates (2.1). Equivalently we can define in Theorem 2.2 (α, α_0) such that $\alpha\mathbf{x}(\mathbf{1}) \leq \alpha_0$ represents one or more constraints $x_e(k) \leq 1$. Then choosing $\gamma = \alpha$ proves that $x_e(k) \leq 1$ is not facet-defining for Q_P^n. It will be shown in Section 3 that the edge-disjointness constraints are facet-inducing for Q_P^n.

Remark 2.2 Since the proofs of the Theorem 2.1 and 2.2 are given in general terms, we expect them to be applicable to other combinatorial optimization problems as well. For instance additional insight might be gained into the relation between the Minimal Spanning Tree polytope and the Edge-disjoint Spanning Trees polytope along this way. Descriptions of these polytopes are obtained by combining matroid and polyhedral theory.

We conclude this section with results that follow by applying Theorem 2.2.

Corollary 2.3 For $k \in \{1, 2\}$ the inequalities given in (1) - (5) beneath are facet-inducing for Q_P^n. We give the reference where the analogue (detailed) result for Q_T^n can be found.

(1) The non-negativity constraints for $n \geq 6$ (Grötschel and Padberg, 1979a).
(2) The subtour elimination constraints $n \geq 6$ (Grötschel and Padberg, 1979b).
(3) The comb inequalities $n \geq 6$ (Grötschel and Padberg, 1979b).
(4) The clique tree inequalities $n \geq 11$ (Grötschel and Pulleyblank, 1986).
(5) The simple crown inequalities $n \geq 8$ (Naddef and Rinaldi, 1992).

Proof Consider a simple crown inequality, to be denoted as $-\sum_e c_e x_e(k) \leq -c_0$; let $k \in \{1, 2\}$ and define $h = 3 - k$. For any $\mathbf{x}(\mathbf{k}) \in Q_T^n$ with $-\sum_e c_e x_e(k) = -c_0$ there exists an $\mathbf{x}(\mathbf{h}) \in Q_T^n$ such that $(\mathbf{x}(\mathbf{k}), \mathbf{x}(\mathbf{h})) \in Q_P^n$ and such that there is an edge $\tilde{e}$ with $x_{\tilde{e}}(k) = 0$, $x_{\tilde{e}}(h) = 1$. Choosing $\gamma_{\tilde{e}} > 0$ and $\gamma_e = 0$ otherwise, gives $-\sum_e c_e x_e(k) + \sum_e \gamma_e x_e(h) > c_0$.

The existence of a vector $\mathbf{x}(\mathbf{k})$ with $-\sum_e c_e x_e(k) = -c_0$ is guaranteed by the facet-inducing property of the simple crown inequality with respect to Q_T^n. Then the facet-inducing property for Q_P^n follows from Theorem 2.2.

The proofs for the classes (1) - (4) are similar. With respect to (1) - (3) and $n = 6$, complete enumeration is suitable to prove the facet-inducing property of the concerning inequalities. □

Exploiting a lifting theorem, Naddef and Rinaldi (1992) have shown that for Q_T^n a simple crown inequality can be generalized to an *extended crown inequality* which is facet-inducing for Q_T^n. Cornuéjols et al. (1985) introduced the path inequalities in studying the Graphical Traveling Salesman polytope. Naddef and Rinaldi (1988) utilized these inequalities to derive new facets for the TSP polytope. Boyd and Hartman have shown independently in unpublished work that the chain inequalities, introduced by Padberg and Hong (1980), are facet-inducing for Q_T^n. Whether these facets for Q_T^n can be lifted to yield facets for Q_P^n requires further research.

3 Additional facets for the PSP polytope

In this section we will use Lemma 3.1 to give a sketch of the proof that the edge-disjointness constraints are facet-inducing for Q_P^n:

Lemma 3.1 (Harary, 1969, p. 89)
Let $G = (V, E)$ be the complete graph on n vertices and let r denote some integer.

(a) If $n = 2r + 1$, then r edge-disjoint Hamiltonian cycles $T(1), \ldots, T(r)$ exist in G, such that $E = \bigcup_{i=1}^{r} T(i)$

(b) If $n = 2r$, then there exist $r-1$ edge-disjoint Hamiltonian cycles $T(1), \ldots, T(r-1)$ and one perfect 1-matching M, edge-disjoint from any cycle such that $E = \bigcup_{i=1}^{r-1} T(i) \cup M$.

Theorem 3.2 For $n \geq 6$ the edge-disjointness constraints $x_e(1) + x_e(2) \leq 1$ induce facets of Q_P^n.

The proof of the theorem is rather technical. A sketch of the proof is provided here. The interested reader is referred to De Kort (1992) for further details.
First note that $\dim(Q_P^n) \leq n(n-3)$, as stated in Section 1.2. We define $d_n = n(n-3)$ and construct d_n PSP solutions that satisfy an arbitrary edge-disjointness constraint with equality and such that the associated incidence vectors form a $d_n \times d_n$ submatrix of full rank. Then it follows from property 3(i) in Section 1.1 that $x_e(1) + x_e(2) \leq 1$ is facet-inducing for Q_P^n.
To obtain the PSP solutions with the required properties we use Lemma 3.1 to construct r edge-disjoint cycles on $n-2$ vertices if n is odd and $r-1$ cycles and a perfect matching on $n-2$ vertices if n is even. Next these cycles are extended to PSP solutions in such a way that the corresponding matrix of incidence vectors has rank d_n.

4 Concluding remarks

The presented material suffices to develop a branch and cut algorithm of the type that has been proven to be very successful for the Traveling Salesman Problem - see De Kort (1992). A typical branch and cut procedure for the PSP combines a fast LP solver with a facet identification procedure and a branching scheme. The initial LP problem consists of the degree constraints together with non-negativity and edge-disjointness constraints. Enumerating the branch and cut search tree a large set of facet-inducing inequalities is obtained which can be added to the LP problem.

Further research can be done in determining additional facets for the PSP polytope that are of the form $\alpha(\mathbf{1})\mathbf{x}(\mathbf{1}) + \alpha(\mathbf{2})\mathbf{x}(\mathbf{2}) \leq \alpha_0$, $\alpha(\mathbf{1}), \alpha(\mathbf{2}) \neq \mathbf{0}$, rather then of the form $\alpha\mathbf{x}(\mathbf{k}) \leq \alpha_0$, $k \in \{1, 2\}$.

5 References

Balakrishnan, A., T.L. Magnanti and P. Mirchandani, "Heuristics, LPs, and trees on trees: Network Design Analyses ", *Operations Research* **44** (1996) 478–496.

Cornuéjols, G., J. Fonlupt and D. Naddef, "The Traveling Salesman Problem on a graph and some related integer polyhedra", *Mathematical Programming* **33** (1985) 1–27.

De Brey M.J.D. and A. Volgenant, "Well-solved cases of the 2-Peripatetic Salesman Problem", *Optimization* (1996), to appear.

De Kort, J.B.J.M., "Edge-disjointness in Combinatorial Optimization - Problems and Algorithms", PhD-thesis University of Amsterdam, Amsterdam (1992).

De Kort, J.B.J.M., "A branch and bound algorithm for symmetric 2-peripatetic salesman problems", *European Journal of Operational Research* **70** (1993) 229–243.

Grötschel, M. and O. Holland, "Solution of large-scale symmetric Travelling Salesman Problems", *Mathematical Programming* **51** (1991) 141–202.

Grötschel, M. and M. Padberg, "On the symmetric Traveling Salesman Problem I: "Inequalities", *Mathematical Programming* **16** (1979a) 265–280.

Grötschel, M. and M. Padberg, "On the symmetric Traveling Salesman Problem II: Lifting theorems and facets", *Mathematical Programming* **16** (1979b) 281–302.

Grötschel, M. and M. Padberg, "Polyhedral theory", In: E.L. Lawler, J.K. Lenstra, A.H.G. Rinnooy Kan and D.B. Shmoys (eds.) *The Traveling Salesman Problem, a guided tour of combinatorial optimization*, Wiley & Sons, Chichester, (1985) 250–305.

Grötschel, M. and W.R. Pulleyblank, "Clique Tree Inequalities and the symmetric Travelling Salesman Problem", *Mathematics of Operations Research* **11** (1986) 537–569.

Harary, F., *Graph Theory*, Addison-Wesley, Reading, Massachusetts (1969).

Held, M. and R.M. Karp, "The Traveling Salesman Problem and minimum spanning trees", *Operations Research* **18** (1970) 1138–1162.

Krarup, J., "The Peripatetic Salesman Problem and some related unsolved problems", In: B. Roy (ed.) *Combinatorial Programming, Methods and Applications*, Reidel, Dordrecht, The Netherlands (1975) 173–178.

Naddef, D. and G. Rinaldi, "The symmetric Traveling Salesman Polytope: New facets from the graphical relaxation", *Report R248* IASI-CNR (December 1988) Rome.

Naddef, D. and G. Rinaldi, "The Crown Inequalities for the symmetric Traveling Salesman polytope", *Mathematics of Operations Research* **17** (1992) 308–326.

Nemhauser, G.L. and L.A. Wolsey, Integer and combinatorial optimization, *Wiley & Sons* (1988) New York.

Padberg, M. and S. Hong, "On the symmetric Traveling Salesman Problem: A computational study", *Mathematical Programming Study* **12** (1980) 78–107.

Padberg, M. and M.R. Rao, "The Russian method for linear inequalities III: Bounded integer programming", *Working Paper 81-39* (1981) New York University, New York.

Padberg, M. and G. Rinaldi, "A branch-and-cut algorithm for the resolution of large-scale symmetric Traveling Salesman Problems", *SIAM Review* **33** (1991) 60–100.

A Polyhedral Approach to Grouping Problems

Maarten Oosten
Department of Mathematics
Maastricht University
E-mail: oosten@math.unimaas.nl

1 Introduction

When I started the quest for a PhD–thesis at Maastricht University in June 1990, I joined a group of researchers focusing on combinatorial aspects of problems arising in flexible manufacturing systems. Their efforts culminated in a series of publications, and eventually theses [Spieksma, (1992)], [Oerlemans, (1992)], [Klundert, (1996)], on different topics concerning scheduling and set–up problems in the highly automated environment of flexible manufacturing systems. In general, for production systems like these, it is extremely difficult to schedule the processing of every part in the right time on the right machine with the right tools. Therefore, it could be advantageous to split up the production process into smaller and relatively independent subprocesses, just to make the tooling, loading, and scheduling problems more tractable. So, I started studying the possibilities of forming production cells. The choice of applying polyhedral techniques was based on the observation that these techniques are efficient for other partitioning problems. Furthermore, they were in line with ongoing research within our group.

However, the central topic of my thesis was not cell formation, but the relations between polyhedra. The reason is that the formation of cells is not just one grouping problem, but it entails a wide variety of slightly different problems. For example, sometimes we have to take into account additional constraints; limits on the number of groups, the number of machines in one group, the number of parts in one group; pairs of machines that have to be in the same group; the possibility to buy additional machines; or alternative ways to process a part. It is easy to extend this list. Clearly, the value of deriving theoretical results for a specific problem is dependent on the possibility to transform those results into useful information about variants of the original problem. As a consequence the emphasis of the study shifted from the formation of production cells towards relating polyhedra. Some interesting results were derived on the strength of inequalities, the projection of polyhedra, and generalized lifting procedures (see [Oosten, (1996)]).

In the remainder of this contribution, I will try to give an impression of the nature of my thesis. First, in the next section, we will discuss the cell formation problem. In Section 3 a mathematical programming formulation for the cell formation problem is presented. This model can be used to derive strong bounds on the optimal value of the problem. Such bounds could prove useful, e.g. to assess the quality of heuristic approaches, or to develop an exact algorithm for the problem. The model was the starting point for the polyhedral approach. As an example of relating polyhedra to each other, in the final section we will use generalized

lifting techniques to extend the basic model to a variant of the cell formation problem, that is getting a lot of attention lately: concurrent engineering.

2 The Cell Formation Problem

Due to increasing international competition and fastidiousness of customers, manufacturers are forced from mass production to the production of a larger product mix. To meet these new requirements, it is very important to have the ability to produce many small volume batches consisting of complex parts in a short production period. This leads to an increased complexity of the management task, increased investments in inventory and a decreased efficiency of mass production systems. To maintain high efficiency levels, it is an accepted strategy to adopt a group technology philosophy, and to organize a large portion of the manufacturing system into cells [Schonberger, (1982), Hyer and Wemmerlöv, (1989)]. A group technology cell consists of a number of machines located close to each other (a machine group) and geared for the manufacturing of a number of similar parts (a part family). To boost efficiency, tooling, loading and scheduling decisions within each cell should be made (almost) independently of the other cells. This requires that machine groups and part families be identified on basis of their interrelations, so that a minimum of interference occurs between the cells. Intercell relations can be dealt with in various ways. Transporting a part during its production process complicates the scheduling and controlling of the cells to a considerable extent. Instead of intercell movements, some other options can be considered like extending the number of machines of a certain type (machine duplication), allowing parts to be rerouted, reviewing the design of the bottleneck parts, or even transferring the production of a part to another production line (part subcontracting). For each of these options, heuristics or exact algorithms have been designed. Most of these approaches, however, assume that a decomposition of the production system into cells has already been determined. Here, we will exclusively deal with the problem of forming such cells.

The data for the cell formation problem is commonly assumed to be summarized in the *machine-part incidence matrix* $A = [a_{ij}]$, where $a_{ij} = 1$ if part j has to be processed on machine i and $a_{ij} = 0$ otherwise. More generally, the elements of A could also represent the processing time required by each part on each machine, or some other numerical data; the models to be discussed here also apply to this situation, with only slight modifications. A 'natural' machine-part grouping is expected to emerge when the rows and columns of the incidence matrix are reordered in a proper way. In that case the diagonal blocks should be relatively 'filled' compared to the other blocks, as illustrated in table below. In this example, deleting the element $(1,2)$ and adding the element $(5,6)$ would result in a perfect block diagonal structure.

	part →	1	2	3	4	5	6
mach	1	1	1	0	1	0	0
↓	2	0	1	1	0	0	1
	3	1	0	0	1	0	0
	4	0	0	0	0	1	0
	5	0	1	1	0	0	0

	1	4	2	3	6	5
3	1	1	0	0	0	0
1	1	1	1	0	0	0
2	0	0	1	1	1	0
5	0	0	1	1	0	0
4	0	0	0	0	0	1

Table 1 Example of near block diagonalization by permutation of rows and columns

The ones outside the diagonal blocks ('exceptions') represent intercell relations and should be avoided as much as possible. On the other hand, it is also undesirable to have zeroes in

the diagonal blocks ('voids') or in other words to have a part and a machine in the same cell when they are not directly related. Informally stated, the cell formation problem consists in minimizing some combination of the number of exceptions and of voids. We now proceed to define the problem more formally.

A matrix $X = [x_{ij}]$ is called *block diagonal* if there exists a partition $R_1, \ldots, R_k, R_{k+1}$ of its row-set and a partition $C_1, \ldots, C_k, C_{k+1}$ of its column-set such that $x_{ij} \neq 0$ if and only if, for some $1 \leq l \leq k$, $i \in R_l$ and $j \in C_l$ (notice that, to be very precise, we should say that X is block diagonal up to permutations of its rows and columns).

When X is a block diagonal machine-part incidence matrix, we interpret the partition of its rows and columns as describing the formation of k cells, where the l-th cell consists of the machines in R_l and the parts in C_l ($l = 1, \ldots, k$). The machines in R_{k+1} and the parts in C_{k+1} are not included in any cell; in many applications R_{k+1} and C_{k+1} may be assumed empty.

If A and X are machine-part incidence matrices of the same dimension and X is block diagonal, then we say that element (i,j) is an *exception* of A (with respect to X) if $a_{ij} = 1$ and $x_{ij} = 0$; we call (i,j) a *void* of A (with respect to X) if $a_{ij} = 0$ and $x_{ij} = 1$.

The cell formation problem can be abstracted into the following *block diagonalization problem*:

> given an incidence matrix A and a function $f(.,.)$, find a block diagonal incidence matrix X of same dimension as A which minimizes $f(A,X)$.

The function $f(.,.)$ is meant to give a measure of the distance, or dissimilarity, between A and X. It may for instance compute some weighted combination of the number of exceptions and voids.

To round off this discussion, it should be noticed that the block diagonalization model has many potential applications outside of the group technology framework discussed above.

3 A New Mathematical Programming Model

Consider again the formulation of the block diagonalization problem: the problem consists in determining a 'close' approximation of the $M \times P$ machine-part incidence matrix A by a $0-1$ block diagonal matrix X. In view of this formulation, it is very natural to choose as decision variables the elements of X, with their obvious interpretation:

$$x_{ij} \begin{cases} = 1 & \text{if machine } i \text{ and part } j \text{ are in the same cell,} \\ & \text{for } i \in \{1, \ldots, M\} \text{ and } j \in \{1, \ldots, P\} \\ = 0 & \text{otherwise.} \end{cases}$$

Proposition 3.1 *The block diagonal incidence matrices correspond exactly to the integer solutions of the system (1)-(2) :*

$$x_{ij} + x_{hj} + x_{ik} - x_{hk} \leq 2 \quad \text{for } i,h \in \{1, \ldots, M\} \text{ and } j,k \in \{1, \ldots, P\} \tag{1}$$

$$0 \leq x_{ij} \leq 1 \qquad i \in \{1, \ldots, M\}, j \in \{1, \ldots, P\} \tag{2}$$

Proof: Omitted. □

Some limited computational experience seems to indicate that this system provides a useful description of the set of all block diagonal matrices. This impression has been confirmed by a variety of interesting theoretical results (see [Oosten, (1996)]).

In studying the model, it is useful to think of an alternative formulation of the block diagonalization in graph theoretic terms; we use the graph theoretic terminology of [Bondy and Murty, (1976)]). Every $M \times P$ binary matrix X can be viewed as the adjacency matrix of a bipartite graph $G(X)$ with vertex set $\{1,\ldots,M\} \cup \{1,\ldots,P\}$; there is an edge $\{i,j\}$ between vertices $i \in \{1,\ldots,M\}$ and $j \in \{1,\ldots,P\}$ if and only if $x_{ij} = 1$. Recall that a bipartite graph on the vertex set $V_1 \cup V_2$ is called complete bipartite if it contains all possible edges between V_1 and V_2, i.e. if its edge set is $V_1 \times V_2$. It is easy to see that a matrix X is block diagonal if and only if each connected component of $G(X)$ is a complete bipartite graph. So, the block diagonalization problem can be interpreted as the problem of approximating a given bipartite graph G by a disjoint union of complete bipartite graphs on the same vertex set as G.

As an illustration, Figure 1 shows the graph associated with the incidence matrix displayed in Table 1. Notice that deleting the edge $(1,2)$ from this graph and adding the edge $(5,6)$ would produce three complete bipartite components, associated with the three diagonal blocks of the corresponding matrix.

The inequalities (1) have an obvious interpretation in graph theoretic terms: if all components of $G(X)$ are complete bipartite, and if three edges $\{i,j\}$, $\{i,k\}$ and $\{h,j\}$ are present in $G(X)$, then $\{h,k\}$ must also be an edge. This interpretation suggests to refer to (1) as the *square inequalities*.

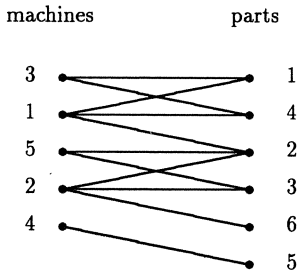


Figure 1 Example of a graph representation of machine–part relations

4 Concurrent Engineering

Suppose now that we have to adapt our model, because we are given an extra opportunity: for some part there is an alternative production plan, requiring a different set of machines. This is not an unusual scenario. The problem of having to choose between different production plans for a part is known in literature as *concurrent engineering*, see for a more thorough discussion of this topic [Ham et al, (1985)].

To start with a simple case, assume there are only two machines and two parts. Let part 1 have a set of production plans, and let $S_1 = \{1,\ldots,s_1\}$ be the index set of these plans. Now, for each production plan there is a binary variable $p^l \in \{0,1\}$ that has value one if and only if production plan l is chosen. Assuming we can choose at most one production plan, we have: $\sum_{l \in S_1} p^l \leq 1$. We also need more x–variables for part 1, with an extra index per process plan: x^l_{ij} has value one if and only if machine i and part j are in the same cell, and production plan l is in use.

First, we take only one process plan into consideration: $s_1 = 1$. Let P be the convex

hull of all feasible solutions. Then the equality $p^1 = 1$ defines a face F of P, and for this face (1)-(2) forms a minimal facial description. We can lift each of these inequalities with respect to p^1. Since the dimension gap between P and F is one, ordinary sequential lifting suffices to derive facet defining inequalities for P. Lifting the square inequalities, we derive:

$$\begin{cases} -p^1 + x^1_{11} + x_{12} + x^1_{21} - x_{22} \leq 1 \\ \quad x^1_{11} + x_{12} - x^1_{21} + x_{22} \leq 2 \\ -p^1 + x^1_{11} - x_{12} + x^1_{21} + x_{22} \leq 1 \\ \quad -x^1_{11} + x_{12} + x^1_{21} + x_{22} \leq 2 \end{cases}$$

Now let us make the case more interesting, and bring into consideration the second process plan: $s_1 = 2$. We interprete P as the face of a larger polytope Q, the convex hull of all feasible solutions for two process plans, that is induced by intersecting Q with the subspace defined by the equality $p^2 = 0$. This equality does not induce an equality system, since the dimension gap between Q and P is three. An inequality system is for example: $p^2 = 0$, $x^2_{11} = p^2$, and $x^2_{21} = p^2$. Then L is the lifting set of the inequality

$$\alpha(p^2 - 0) + \beta(x^2_{11} - p^2) + \gamma(x^2_{21} - p^2) - p^1 + x^1_{11} + x_{12} + x^1_{21} - x_{22} \leq 1.$$

$$L = \left\{ \begin{array}{lll} \alpha & \in I\!R & \\ \beta & \in I\!R & \\ \gamma & \in I\!R & \end{array} \middle| \begin{array}{lll} \gamma - \alpha - \beta & \leq & 0 \\ \gamma - \beta & \leq & 0 \\ \gamma - \alpha & \leq & 0 \\ \gamma & \leq & 1 \end{array} \right\}$$

The polyhedron L has two extreme points: $(0,0,0)$ and $(1,1,1)$. These correspond to respectively the following inequalities:

$$-p^1 + x^1_{11} + x_{12} + x^1_{21} - x_{22} \leq 1$$

and

$$-p^1 - p^2 + x^1_{11} + x^2_{11} + x_{12} + x^1_{21} + x^2_{21} - x_{22} \leq 1$$

The first one could have been derived by applying ordinary sequential lifting, but without the guarantee that it defines a facet of Q. The second one is due to extended lifting. The second type of inequalities give some extra insight that leads to a useful generalization. Let S_l be the index set of concurrent project plans for part l. Then, the following inequality is facet defining for the polytope that is the convex hull of the feasible solutions of the concurrent engineering cell formation problem:

$$-\sum_{l \in T_j} p^l_j + \sum_{l \in T_j} x^l_{ij} + \sum_{l \in T_j} x^l_{hj} + x^n_{ik} - x^n_{jk} \leq 1$$

for all $n \in S_k$, for all $T_j \subseteq S_j$ such that $T_j \neq \emptyset$, for all $h, i \in \{1 \ldots M\}$, for all $j, k \in \{1 \ldots N\}$. We refer to this class of inequalities as *concurrent square inequalities*. Although this is an exponential class of inequalities, it is easy to see that the separation problem is polynomial solvable.

Acknowledgement: The author wishes to express his gratitude towards the LNMB for supporting his PhD research.

References

[Bondy and Murty, (1976)]
J.A. Bondy, and U.S.R. Murty, *Graph Theory with Applications* New York: North Holland, 1976.

[Ham et al, (1985)]
I. Ham, K. Hitomi, and T. Yoshida, "Group Technology: Applications to Production Management" Kluwer-Nijhoff, 1988.

[Hyer and Wemmerlöv, (1989)]
N.L. Hyer, and U. Wemmerlöv, "Group technology in the US manufacturing industry: a survey of current practices", *International Journal of Production Research*, **27**, pp 287–1304.

[Klundert, (1996)]
J. van de Klundert, *Scheduling Problems in Automated Manufacturing.* Maastricht University, Maastricht, PhD–thesis, June 1996.

[Oerlemans, (1992)]
A.G. Oerlemans, *Production Planning for Flexible Manufacturing Systems.* Maastricht University, Maastricht, PhD–thesis, October 1992.

[Oosten, (1996)]
M. Oosten, *A Polyhedral Approach to Grouping Problems.* Maastricht University, Maastricht, PhD–thesis, December 1996.

[Schonberger, (1982)]
R.J. Schonberger, *Japanese Manufacturing Techniques*, The Free Press, 1982.

[Spieksma, (1992)]
F.C.R. Spieksma, *Assignment and Scheduling Algorithms in Automated Manufacturing.* Maastricht University, Maastricht, PhD–thesis, February 1992.

Determining Haemers' rank bound is NP-hard

René Peeters
Faculty of Economics
Tilburg University
E-mail: m.j.p.peeters@kub.nl

Abstract

We prove that determining Haemers's rank bound is NP-hard by giving a reduction from the 3-colorability problem to the problem of deciding whether or not the Haemers bound is equal to three.

This article is based on [16] and on Chapter 6 of my dissertation [15].

1 Introduction

Let G be a graph with vertex set $V(G) = \{1, 2, \ldots, n\}$. The **chromatic number** of G, which we denote by $\chi(G)$, is the minimum number of colors needed to color the vertices of G, such that the two endpoints of any edge have different colors. To determine the chromatic number of a general graph is a difficult (NP-hard) problem. Even the problem of deciding whether a graph is 3-colorable or not is already NP-complete (cf. [6, 3, 14]). A trivial lower bound on the chromatic number of G is the **clique number** of G (denoted by $\omega(G)$), that is, the number of vertices in a largest complete subgraph of G. Clearly these $\omega(G)$ vertices must be colored differently in every legal coloring of G. Also the problem of determining the clique number of a general graph is NP-hard (cf. [6, 3, 14]).

In 1979 Lovász [11] (See also [10, 12]) introduced for each graph a number $\theta(G)$ (the **Lovász bound**), with the following property:

$$\omega(G) \leq \theta(\overline{G}) \leq \chi(G).$$

($\overline{G}$ denotes the complementary graph of G.) In [7] it is proved that $\theta(G)$ can be calculated (or, in fact approximated, since $\theta(G)$ doesn't need to be rational) by the ellipsoid method in polynomial time. This implies for instance that for perfect graphs (these are graphs for which for all its subgraphs the chromatic number and the clique number are equal) the chromatic number and the clique number can be determined in polynomial time.

Let A be an $n \times n$-matrix (over some field) with all diagonal elements non-zero and with $A_{ij} = 0$ if i and j are adjacent in G. If a matrix satisfies these conditions, we say that the matrix *fits* G[1]. Clearly $\text{rank}(A) \geq \omega(G)$, since A has a diagonal matrix of size $\omega(G)$, with non-zero diagonal entries, as a submatrix. On the other hand, there exists a matrix A that

[1]This definition of a matrix fitting a graph is the same as by Haemers in [9], although in that paper the condition that $A_{ij} = 0$ if i and j are adjacent should be read as $A_{ij} = 0$ if i and j are **not** adjacent to be consistent with the rest of the paper.

fits G for which $\text{rank}(A) = \chi(G)$. Indeed, let G be colored with $\chi(G)$ colors and define the matrix A by

$$A_{ij} := \begin{cases} 1 & \text{if } i \text{ and } j \text{ are in the same color class,} \\ 0 & \text{otherwise,} \end{cases}$$

then A fits G and $\text{rank}(A) = \chi(G)$. So the minimum rank over all matrices fitting G is also a number between the clique number and the chromatic number of G. This number was introduced by Haemers ([8, 9]) and we will refer to it as the **Haemers bound**.

The Lovász bound and the Haemers bound were both introduced as an upper bound for the so-called Shannon capacity of a graph (cf. [17]), a concept which we will not further discuss here. Using his bound, Lovász [11] could prove that the pentagon had Shannon capacity equal to $\sqrt{5}$, solving a problem that was open for over twenty years. Since Lovász had no examples of graphs for which his bound was provable larger than the Shannon capacity, he raised the question whether or not his bound was equal to the Shannon capacity. This problem was solved by Haemers [8, 9] by introducing his rank bound. Although the Haemers bound is worse for most graphs, it is sometimes (much) better than the Lovász bound showing that the Lovász bound is not always equal to the Shannon capacity.

A drawback of the Haemers bound is its definition. It is defined as the minimum over all fields of the minimal possible rank of a matrix over this field fitting G. For this reason it is not easy to work with. In this article we will introduce some interesting classes of matrices fitting a graph over some fixed field. Of course these classes should be defined in such a way that the smallest rank in these classes is at most the chromatic number of the graph. Examples of suitible classes are the following:

$$\begin{aligned} \mathcal{A}_3(G,F) &:= \{A \in F^{n \times n} | A \text{ fits } G\} \\ \mathcal{A}_2(G,F) &:= \{A \in \mathcal{A}_3(G,F) | A \text{ symmetric}\} \\ \mathcal{A}_1(G,F) &:= \{A \in \mathcal{A}_2(G,F) | A_{ii} = 1\}. \end{aligned}$$

A question that arises is: What is the complexity of determining the minimum rank of a matrix fitting a graph G if the field and the class of matrices over which the minimimum should be taken are given? This is the main problem we concentrate on in this paper. For any class $\mathcal{A}$ of matrices over F we introduce the following number:

$$R(\mathcal{A}) := \min\{\text{rank}(A) | A \in \mathcal{A}\}.$$

In the next section we will show that for some small finite fields for some of the classes mentioned above the minimal rank in this class is equal to three if and only if the graph is 3-colorable. This means that for these classes determining the Haemers bound is NP-hard. This suggests that for all relevant classes of matrices fitting a graph determining the smallest rank in this class is NP-hard. In the last section we will prove this.

2 Minimal rank and the chromatic number

We start with two trivial remarks that inspired our first approach of determining the complexity of determining the Haemers bound. It turns out that for some finite fields F, the number $R(\mathcal{A}_i(G,F))$ is equal to the chromatic number of G if this chromatic number is small. First of all we trivially have for any field F:

$$R(\mathcal{A}_i(G,F)) = 1 \Leftrightarrow \chi(G) = 1.$$

and since for odd cycles we have that $R(\mathcal{A}_i(G,F)) = 3$, also

$$R(\mathcal{A}_i(G,F)) = 2 \Leftrightarrow \chi(G) = 2$$

for any field F. If for some field F we can prove that also

$$R(\mathcal{A}_i(G,F)) = 3 \Leftrightarrow \chi(G) = 3$$

then we immediately get that for this field F determining the minimal rank of all matrices in $\mathcal{A}_i(G,F)$ is NP-hard. In fact it is enough to prove this equivalence for a class of graphs for which the 3-colorability problem is still NP-complete such as planar graphs or graphs containing a triangle (see for instance [4]).

For finite fields it turns out to be relatively easy to verify whether or not the above relation holds. We will illustrate this by proving the relation for the class $\mathcal{A}_1(G, \mathbb{F}_7)$.

Theorem 2.1

$$R(\mathcal{A}_1(G, \mathbb{F}_7)) = 3 \quad \Leftrightarrow \quad \chi(G) = 3,$$

Proof: Let G be a graph for which the class $\mathcal{A}_1(G, \mathbb{F}_7)$ contains a rank-3 matrix, A say. Then A contains a non-singuler principal submatrix A_{11} of rank three and A can be partitioned as follows:

$$A = \left(\begin{array}{c|c} A_{11} & A_{12} \\ \hline A_{12}^T & A_{12}^T A_{11}^{-1} A_{12} \end{array}\right) = \left(\begin{array}{c} A_{11} \\ \hline A_{12}^T \end{array}\right) A_{11}^{-1} \left(\begin{array}{c|c} A_{11} & A_{12} \end{array}\right)$$

Because A_{11}, and hence A_{11}^{-1}, is symmetric en non-singular, we can decompose A_{11}^{-1} as follows:

$$A_{11}^{-1} = \begin{cases} B'^T \begin{pmatrix} 1 & & \\ & 1 & \\ & & 1 \end{pmatrix} B' & \text{if } \det A_{11}^{-1} \text{ is a square,} \\ B'^T \begin{pmatrix} 1 & & \\ & 1 & \\ & & 3 \end{pmatrix} B' & \text{if } \det A_{11}^{-1} \text{ is not a square.} \end{cases}$$

Hence there are two possibilities:

$$A = B^T B$$

for some $3 \times n$-matrix B, or

$$A = B^T \begin{pmatrix} 1 & & \\ & 1 & \\ & & 3 \end{pmatrix} B$$

for some $3 \times n$-matrix B.

In both cases there is only a limited number of possibilities for the columns of B. If we consider opposite columns as the same, we have 21 possibilities for the first case and 28 possibilities for the second case. In the worst case, every zero in the matrix A corresponds with an edge and all vertices should be colored differently if and only if the corresponding columns are orthogonal. Without loss of generality we may assume that vertices whose corresponding columns are the same or opposite are colored the same since the inner product of these columns is unequal to zero.

Define the graph Γ_1 as the graph with vertex set the 21 possible columns for the first case (with the standard inner product), two vertices being adjacent if and only if the columns

are orthogonal and the graph Γ_2 as the graph with vertex set the 28 possible columns of the second case (the non-standard inner product), two vertices being adjacent if and only if the columns have inner product zero. The maximal chromatic number of any graph for which there exists a rank-3 matrix in $\mathcal{A}_1(G, \mathbb{F}_7)$ is equal to largest chromatic number of Γ_1 and Γ_2.

Γ_1 turns out to be the line graph of the Heawoodgraph and is distance-regular (cf. [2]). An alternative definition is the following: Take as vertices the flags (incident point-line pairs) of the Fano plane, two flags being adjacent whenever they contain the same point or the same line. The 3-colorability problem reduces to the question if we can partition the 21 flags of the Fano plane into 3 sets of 7 flags each, such that each point and each line occurs in one of the flags of each set. Since there is a circulant line-point incidence matrix of the Fano plane (for instance with top row (1101000)) such a partition is possible. Γ_2 is the Coxeter graph, the distance-regular graph on 28 vertices with degree 3. By Brooks' theorem [1] it is 3-colorable.

■

Using the same method we can also prove the following equivalences:

Theorem 2.2 *Let G be any graph, then*

$$\begin{aligned} R(\mathcal{A}_1(G, \mathbb{F}_2)) = 5 &\Leftrightarrow \chi(G) = 5, \\ R(\mathcal{A}_1(G, \mathbb{F}_3)) = 4 &\Leftrightarrow \chi(G) = 4, \\ R(\mathcal{A}_1(G, \mathbb{F}_5)) = 3 &\Leftrightarrow \chi(G) = 3, \\ R(\mathcal{A}_1(G, \mathbb{F}_7)) = 3 &\Leftrightarrow \chi(G) = 3, \\ R(\mathcal{A}_2(G, \mathbb{F}_3)) = 3 &\Leftrightarrow \chi(G) = 3. \end{aligned}$$

Let G be a graph with $\omega(G) = 3$ then

$$R(\mathcal{A}_1(G, \mathbb{F}_9)) = 3 \Leftrightarrow \chi(G) = 3.$$

Let G be a graph with $\omega(G) = 5$ then

$$R(\mathcal{A}_1(G, \mathbb{F}_3)) = 5 \Leftrightarrow \chi(G) = 5.$$

■

Since the 3-colorability problem for graphs is NP-complete, also if G is restricted to have a triangle (see for instance [4]) we get the following result:

Corollary 2.3 *Determining the Haemers bound is NP-hard if we restrict to one of the following classes: $\mathcal{A}_i(G, F)$ for $i = 1$ and $F = \mathbb{F}_q$ with $q \in \{2, 3, 5, 7, 9\}$ and for $i = 2$ and $F = \mathbb{F}_3$.* ■

Unfortunately, in this way we can prove for only a few of the defined classes of matrices fitting a graph that the problem of deciding whether they contain a matrix of rank equal to three or not is NP-complete, but the results suggest that the problem is NP-complete for all classes $\mathcal{A}_i(G, F)$ over finite fields.

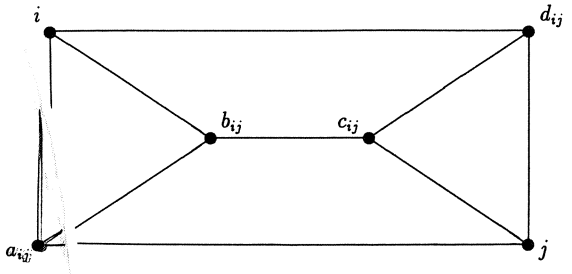


Figure 1: The graph H_{ij}

3 A reduction from 3C to RANK-3

In this section we prove that for all relevant classes of matrices fitting a graph G it is NP-hard to check whether or not it contains a matrix of rank equal to 3. More precisely, let for some fixed field F and any graph G, $\mathcal{A}(G)$ be a class of matrices over F, fitting G, such that it contains a matrix of rank $\chi(G)$. Define the following problem:

Name: RANK-3.

Input: A graph G.

Question: Does $\mathcal{A}(G)$ contain a matrix of rank 3?

Then

Theorem 3.1 *RANK-3 is NP-hard.*

Proof: Consider the 3-Coloration problem, (3C), which is defined as follows:

Name: 3C.

Input: A graph G.

Question: Is $\chi(G) \leq 3$?

It is proved in [4] (see also [3, 14]) that 3C is an NP-complete problem. We show that 3C is polynomially reducible to RANK-3.

Let $G = (V, E)$ with vertex set $V = \{1, 2, \ldots, n\}$ be the input I to the 3C-problem. First we construct a graph $f(I)$ such that $f(I)$ is 3-colorable if and only if G is. Secondly we prove that $f(I)$ is 3-colorable if and only if $\mathcal{A}(f(I))$ contains a rank-3 matrix fitting $f(I)$.

In order to construct the graph $f(I)$, introduce for each unordered pair of vertices from G, i and j with $i < j$ say, four extra vertices a_{ij}, b_{ij}, c_{ij} and d_{ij} and nine extra edges such that these nine edges form the graph H_{ij} as shown in Figure 1. So, apart from a possible edge between i and j, H_{ij} is the induced subgraph of $f(I)$ on the vertices $i, j, a_{ij}, b_{ij}, c_{ij}$ and d_{ij}. So $f(I)$ has $|V| + 2n(n-1)$ vertices and $|E| + \frac{9}{2}n(n-1)$ edges.

Notice that there are essential two different valid 3-colorings of H_{ij}, one with color classes $\{\{i, c_{ij}\}, \{a_{ij}, d_{ij}\}, \{b_{ij}, j\}\}$ and one with color classes $\{\{i, j\}, \{a_{ij}, c_{ij}\}, \{b_{ij}, b_{ij}\}\}$, so for one 3-coloring i and j get different colors and for the other coloring i and j are colored the same.

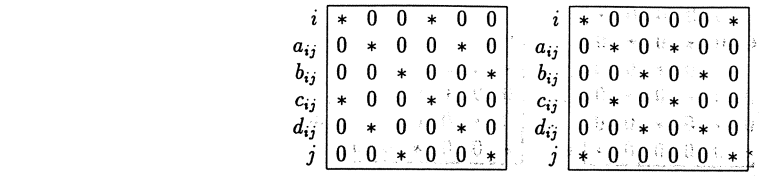


Figure 2: The two types of rank-3 matrices fitting H_{ij}

It follows that $f(I)$ is 3-colorable if and only if G is since a valid 3-coloring of $f(I)$ induces a 3-coloring of G and a 3-coloring of G can always be completed to a valid 3-coloring for $f(I)$.

It is an exercise to check that for any field there are essentially only two different types of matrices of rank 3 fitting H_{ij} corresponding to the two different 3-colorings of H_{ij}. These two types are shown in Figure 2 where a $*$ denotes a non-zero field element. The first one corresponds to the coloring with color classes $\{\{i, c_{ij}\}, \{a_{ij}, d_{ij}\}, \{b_{ij}, j\}\}$ and the second one to the coloring with color classes $\{\{i, j\}, \{a_{ij}, c_{ij}\}, \{b_{ij}, d_{ij}\}\}$. The row vectors of vertices from the same color class are scalar multiples of each other.

Finally we show that $f(I)$ is 3-colorable if and only if $\mathcal{A}(f(I))$ contains a matrix of rank 3. First of all, by assumption $\mathcal{A}(f(I))$ contains a matrix of rank 3 if $f(I)$ is 3-colorable. Now assume that there exists a rank-3 matrix, M say, fitting $f(I)$. We prove that $f(I)$ is 3-colorable as follows: Denote the 1-dimensional subspace of $\langle M \rangle$ spanned by the row vector of vertex i by V_i. We show that there are only three different spaces V_i inducing a 3-coloring of G that can be completed to a 3-coloring of $f(I)$.

Let for each $i \neq j \in \{1, 2, \ldots, n\}$ $H_i(j)$ be the 2-dimensional subspace of $\langle M \rangle$ spanned by the row vectors of the vertices from H_{ij} that are not in V_i, then clearly $\langle M \rangle = V_i \oplus H_i(j)$. For any j different from i the coordinate corresponding to i of all vectors in $H_i(j)$ is zero, while this coordinate is non-zero for each non-zero vector from V_i. So $H_i(j)$ is the 2-dimensional subspace of $\langle M \rangle$ consisting of all vectors for which the coordinate corresponding to vertex i is zero. It follows that $H_i(j)$ is independent of j which defines for any i a 2-dimensional subspace H_i of $\langle M \rangle$.

Since for any two vertices i and j we have a subgraph H_{ij}, the two 1-dimensional subspaces V_i and V_j are the same if and only if $H_i = H_j$. If $V_i \neq V_j$ then $V_i \subset H_j$ and $V_j \subset H_i$, so if $V_i \neq V_j$ then all V_k different from V_i and V_j are in the 1-dimensional subspace $H_i \cap H_j$, so there are only three different V_i's. ■

Corollary 3.2 *The RANK-3 problem is NP-complete for all the classes $\mathcal{A}_i(G, F)$ $(i = 1, 2, 3)$ with F a finite field.*

Remark 3.1 *In [15] it is proved that RANK-3 is still NP-complete if restricted to planar graphs or to planar unit disc graphs.*

Remark 3.2 *In [13] the authors remarked that "It seems to be difficult to find the smallest dimension in which a given graph G has an orthonormal representation." (in $\mathbb{R}^d$ provided with the standard inner product). It follows from the above theorems that if the problem of deciding whether or not a (planar) graph has an orthonormal representation in F^3 is in NP (which is the case if F is finite), it is NP-complete.*

References

[1] R.L. Brooks, On colouring the nodes of a network, *Proceedings of the Cambridge Philosophical Society* **37**, (1941) 194–197.

[2] A.E. Brouwer, A.M. Cohen and A. Neumaier, *Distance-Regular Graphs*, Ergebnisse der Mathematik 3.18, Springer, Heidelberg (1989).

[3] S. Even, *Graph Algorithms*, Pitman, (1979).

[4] M.R. Garey, D.S. Johnson and L.J. Stockmeyer, Some NP-Complete Graph Problems, *Theor. Comput. Sci.*, **1**, (1976) 237–267.

[5] M.R. Garey and D.S. Johnson, The Complexity of Near-Optimal Graph Coloring, *J. ACM.* **23**, (1976) 43–49.

[6] M.R. Garey and D.S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-completeness*, Freeman, San Francisco, (1979).

[7] M. Grötchel, L. Lovász, A. Schrijver, The ellipsoid method and its consequences in combinatorial optimization, *Combinatorica* **1** (2), (1981) 169–197.

[8] W. Haemers, On the problems of Lovász concerning the Shannon capacity of a graph, *IEEE Trans. Inform. Theory*, **25**, (1979) 231–232.

[9] W. Haemers, An upper bound for the Shannon capacity of a graph, *Colloqua Mathematica Societatis János Bolyai* **25**, Algebraic Methods in Graph Theory, Szeged (Hungary), (1978) 267–272.

[10] D.E. Knuth, The Sandwich Theorem, *Electronic J. Comb.* **1**, (1994) #A1.

[11] L. Lovász, On the Shannon capacity of a graph, *IEEE Trans. Inform. Theory* **25**, (1979) 1–7.

[12] L. Lovász, An Algorithmic Theory of Numbers, Graphs and Convexity, *CBMS Regional Conference Series in Applied Mathematics*, SIAM, (1986) §3.2.

[13] L. Lovász, M. Saks, A. Schrijver, Orthogonal Representations and Connectivity of Graphs, *Linear Algebra Appl.* **114/115**, (1989) 439–454.

[14] C.H. Papadimitriou, *Computational Complexity*, Addison-Wesley Publ. Co., (1994).

[15] R. Peeters, *Ranks and Structure of Graphs*, dissertation, Tilburg University, (1995).

[16] R. Peeters, Orthogonal representations over finite fields and the chromatic number of graphs, *Combinatorica* **16** (3), (1996) 417-431.

[17] C.E. Shannon, The zero-error capacity of a noisy channel, *IRE Trans. Inform. Theory* **3**, (1956) 3–15.

$1, \frac{3}{2}, \frac{13}{7}, \ldots$

Frits C.R. Spieksma
Department of Mathematics
Maastricht University
E-mail: spieksma@math.unimaas.nl

1 Prelude

Monday, February 1, 1988. This day was the first day of the first course for Ph. D. students organized by the LNMB; it was also the first day of my "AIO"-ship at the Department of Mathematics at Maastricht University. Hence, one could say that a lustrum for the LNMB marks at the same time a lustrum for myself as a researcher. Perhaps because of this I feel somewhat attached to the LNMB and am grateful for this opportunity to contribute to the lustrum book.

2 Introduction

The standard framework of worst-case analysis is usually as follows: imagine you're given this large instance (say $\mathcal{I}$) of a difficult (= $\mathcal{NP}$-hard) problem (say a minimization problem) with not much time to solve it. So using some guidelines you think reasonable (the simple heuristic H) you construct a feasible solution ($H(\mathcal{I})$) with a certain cost ($c(H(\mathcal{I}))$. And then, in some cases, one can produce the following statement:

$$c(H(\mathcal{I})) \leq \alpha \cdot \mathrm{OPT}(\mathcal{I}) \quad \text{for some } \alpha \in \mathbf{R}, \textit{ for all } \mathcal{I}, \tag{1}$$

where $\mathrm{OPT}(\mathcal{I})$ denotes the optimal value associated to instance $\mathcal{I}$.

At very first sight, it may seem that magic was needed to produce such a statement: how else can one say "for all $\mathcal{I}$" without enumerating all instances and running the heuristic on it? Well, when you get down to it, magic is perhaps not the right word describing how such a statement can be produced, but it sometimes feels that way when writing it down.

Let me spend a few words concerning terminology. The validity of (1) implies that α is an *upper bound for the worst-case ratio* of the heuristic H with respect to the problem considered. When, in addition to this, an instance $\mathcal{I}$ can be exhibited for which the heuristic actually delivers a solution with value $\alpha \cdot OPT(\mathcal{I})$, α can be called the *worst-case ratio* (or the *ratio* for short).

Not surprisingly, this contribution deals with the worst-case analysis of a simple heuristic for a difficult problem. In Section 3 I describe a certain k-dimensional assignment problem and a heuristic, and I discuss the results known sofar (which are succinctly summarized in the title). Section 4 introduces a basic observation, which is used in Section 5 to deduce, for the special case $k = 5$, two inequalities and an LP-model. This model enables us to construct, for the case $k = 5$, a tighter upper bound for the ratio than currently known. Section 6 indicates that this approach can be generalized to arbitrary k.

3 A difficult problem, a simple heuristic, and the results known sofar

3.1 A difficult problem

Let $k \geq 2$ be an integer, and let $K = \{1, \ldots, k\}$. Given is a complete k-partite graph $G = (V = \cup_{l=1}^{k} V_l, E)$ with $|V_l| = n$ for all $l \in K$. I will sometimes refer to the vertices of a set V_l as vertices of color l. For each $u \in V_i, v \in V_j, i, j \in K, i \neq j$, there is a nonnegative cost $c_{uv} \in \mathbb{R}$ associated to the corresponding edge $\{u, v\} \in E$. These costs are not arbitrary: apart from being nonnegative, I assume that the so-called triangle inequality holds, that is:

$$c_{uv} + c_{vw} \geq c_{uw} \text{ for all } u \in V_i, v \in V_j, w \in V_l, \text{ and for all } i, j, l \in K, i \neq j, i \neq l, j \neq l. \quad (2)$$

Define a clique $C \subset V$ as a set of vertices such that $|C \cap V_l| = 1$ for all $l \in K$. The cost of a clique is defined as $\sum_{u,v \in C} c_{uv}$. The problem is now to find a partition of V into n disjoint cliques $C_1, \ldots, C_n$ such that the sum of the costs of the cliques is minimal. This problem is called the k-dimensional assignment problem with clique costs (k-DAPC). Notice that for $k = 2$ the problem boils down to an assignment problem.

3.2 A simple heuristic

Given a large instance of this problem with not much time to solve it, what can one do? An idea is the following: specify a sequence of the k colors, and repeatedly assign the vertices of color i to the sets of vertices consisting of vertices of colors $1, \ldots, i-1$, for $i = 2, \ldots, k$. The total cost of assigning a particular vertex of color i to a particular set of vertices of colors $1, \ldots, i-1$ is taken to be the sum of the costs of the $i-1$ edges between the vertex of color i on the one hand, and each of the other $i-1$ vertices on the other hand. So the algorithm consists of solving iteratively $n-1$ assignment problems.

In order to describe this algorithm more formally, let $V_l = \{v_{l1}, \ldots, v_{ln}\}$ for all $l \in K$. Here is the heuristic H^k:

Step 1: Choose a sequence, say $1, 2, \ldots, k$.

Step 2: Set $P_j = \{v_{1j}\}$ for $j = 1, \ldots, n$. Set $l = 1$.

Step 3: While $l < k$ do

- i: For all $i, j = 1, \ldots, n$, compute $\delta_{ij} = \sum_{w \in P_j} c_{w, v_{l+1,i}}$.
- ii: Find an optimal assignment A between the vertices of V_{l+1} and the partial cliques P_j, $j = 1, \ldots, n$ with respect to the cost function δ.
- iii: Extend the partial cliques P_j according to the assignment, that is for each $\{P_j, v_{l+1,i}\} \in A$ set $P_j := P_j \cup \{v_{l+1,i}\}$.
- iv: $l := l + 1$.

Step 4: The cliques are now formed by P_j for $j = 1, \ldots, n$. Stop.

3.3 The results known sofar

Two chapters of my thesis (Spieksma [3]) are devoted to (variants of) k-DAPC: Chapter 3, co-authored with Yves Crama (published as [2]) shows that 3-DAPC is $\mathcal{NP}$-hard, presents the heuristic H^3 and proves the second number in the title as the ratio for this heuristic

applied to 3-DAPC. Chapter 4 of my thesis, co-authored with Hans-Jürgen Bandelt and Yves Crama (published as [1]) presents the heuristic H^k of Subsection 3.2, proves that $\frac{1}{2}k$ is an upper bound for the ratio, (or in other words shows that:

$$c(H^k) \leq \frac{1}{2}k \cdot OPT \text{ for all } k \geq 2), \tag{3}$$

and proves that $\frac{13}{7}$ is the ratio for 4-DAPC. For $k \geq 5$ it is unknown what the worst-case ratio of H^k with respect to k-DAPC is.

Let me add here that in [1] a heuristic is presented of polynomial complexity (albeit with a larger complexity than H^k) which achieves a ratio of $2 - \frac{2}{k}$ for all $k \geq 2$, thus a heuristic with a worst-case ratio bounded by 2 for all $k \geq 2$.

Finally, the reader may wonder whether there are practical applications which motivate the study of this heuristic for k-DAPC with $k \geq 5$. The answer is no; not that I'm aware of. The best reason I can give for looking into this is curiosity concerning the series $1, \frac{3}{2}, \frac{13}{7}, \ldots$ Where does it go, and is there perhaps a closed formula which describes these ratios? Perhaps disappointingly, these questions will remain unanswered at the end of this contribution.

4 An observation

The following notation will be used. Given an instance of k-DAPC, let F denote an optimal solution, and let H denote a solution found by H^k. Define for all $i, j \in K, i < j$:

$$F_{ij} := \{\{u,v\} \mid u \in V_i,\ v \in V_j,\ u \text{ and } v \text{ are contained in a clique from } F\},$$

$$d^F_{ij} := \sum_{\{u,v\} \in F_{ij}} c_{uv},$$

$$H_{ij} := \{\{u,v\} \mid u \in V_i,\ v \in V_j,\ u \text{ and } v \text{ are contained in a clique from } H\}, \text{ and}$$

$$d^H_{ij} := \sum_{\{u,v\} \in H_{ij}} c_{uv}.$$

More generally, given a solution S to k-DAPC, let for all $i, j \in K, i < j$:

$$S_{ij} := \{\{u,v\} \mid u \in V_i,\ v \in V_j,\ u \text{ and } v \text{ are contained in a clique from } S\}, \text{ and}$$

$$d^S_{ij} := \sum_{\{u,v\} \in S_{ij}} c_{uv}.$$

Also, I use:

$$d_{\bullet,i} := \sum_{j=1}^{i-1} d_{j,i} \quad \text{for all } i = 2, \ldots, k.$$

A crucial observation for the analysis which follows is the following one. Consider solution H found by the heuristic H^k. Given H, let us construct an alternative solution as follows. Choose a color $i, i < k$. Reassign the vertices of V_k to the partial cliques according to how in an *optimal solution* to this instance of k-DAPC the vertices of V_k and V_i are matched together. Call this solution S. Due to the fact that the heuristic finds an optimal matching between V_k and the partial cliques, we have the following inequality:

$$c(H^k) = d^H_{\bullet,2} + d^H_{\bullet,3} + \ldots + d^H_{\bullet,k} \leq d^H_{\bullet,2} + d^H_{\bullet,3} + \ldots + d^H_{\bullet,k-1} + d^S_{\bullet,k}.$$

Since $d^S_{ik} = d^F_{ik}$ by construction, and since $d^S_{jk} \leq d^H_{ji} + d^F_{ik}$ for all $j \neq i$ by the triangle inequalities (2), we obtain the following observation:
Observation:

$$c(H^k) \leq d^H_{\bullet,2} + d^H_{\bullet,3} + \ldots + d^H_{\bullet,k-1} + d^H_{1,i} + d^H_{2,i} + \ldots + d^H_{i,k-1} + (k-1)d^F_{ik}, \tag{4}$$

for all $i = 1, \ldots, k-1$.

5 The case $k = 5$

This section consists of three subsections. In each of the first two, an inequality is deduced. In the final subsection I use these inequalities in an LP-model to obtain a tighter upperbound than predicted by (3) for the case $k = 5$.

5.1 Inequality 1

Lemma 1 $c(H^5) \leq 4d^F_{12} + 3(d^F_{13} + d^F_{23}) + 6d^F_{34} + 4d^F_{45}$.

Proof:
We can derive this inequality as follows. Obviously:

$$c(H^5) = d^H_{\bullet,2} + d^H_{\bullet,3} + d^H_{\bullet,4} + d^H_{\bullet,5}. \tag{5}$$

Now consider iteratively each of the following inequalities, substitute it in (5), and proceed:

1: $d^H_{\bullet,5} \leq d^H_{\bullet,4} + 4d^F_{45}$.

This follows from Inequality (4) with $k = 5$ and $i = 4$.

2: $2d^H_{\bullet,4} \leq 2d^H_{\bullet,3} + 6d^F_{34}$.

This follows from Inequality (4) with $k = 4$ and $i = 3$.

3: $\frac{3}{2}d^H_{\bullet,3} \leq \frac{3}{2}d^H_{\bullet,2} + 3d^F_{23}$.

This follows from Inequality (4) with $k = 3$ and $i = 2$.

4: $\frac{3}{2}d^H_{\bullet,3} \leq \frac{3}{2}d^H_{\bullet,2} + 3d^F_{13}$.

This follows from Inequaltity (4) with $k = 3$ and $i = 1$.

5: $4d^H_{\bullet,2} \leq 4d^F_{\bullet,2}$.

Trivial.

□

5.2 Inequality 2

Lemma 2 $c(H^5) \leq \frac{46}{9}d^F_{12} + \frac{23}{9}(d^F_{13} + d^F_{23}) + \frac{4}{3}(d^F_{14} + d^F_{24} + d^F_{34}) + \frac{4}{3}(d^F_{15} + d^F_{25} + d^F_{35})$.

Proof:
As before we can derive this inequality as follows. Obviously:

$$c(H^5) = d^H_{\bullet,2} + d^H_{\bullet,3} + d^H_{\bullet,4} + d^H_{\bullet,5}. \tag{6}$$

Now consider iteratively each of the following inequalities, substitute it in (6), and proceed:

1: $d^H_{\bullet,5} \leq \frac{4}{3}(d^F_{15} + d^F_{25} + d^F_{35}) + \frac{1}{3}d^H_{\bullet,4} + \frac{2}{3}(d^H_{\bullet,3} + d^H_{\bullet,2})$.

This follows by summing Inequality (4) with $k = 5$ for $i = 1, 2, 3$, and dividing it by 3.

2: $\frac{4}{3}d^H_{\bullet,4} \leq \frac{4}{3}d^F_{\bullet,4} + \frac{8}{9}(d^H_{\bullet,3} + d^H_{\bullet,2})$.

This follows by summing Inequality (4) with $k = 4$ for $i = 1, 2, 3$, and dividing it by 3.

3: $\frac{23}{9}d^H_{\bullet,3} \leq \frac{23}{9}(d^F_{\bullet,2} + d^F_{\bullet,3})$.

This follows by summing Inequality (4) with $k = 3$ for $i = 1$ and 2, and dividing it by 2.

4: $\frac{23}{9}d^H_{\bullet,2} \leq \frac{23}{9}d^F_{\bullet,2}$.

Trivial.

□

5.3 An LP-model

How to use the two inequalities from Lemma's 1 and 2 from the previous subsections to obtain a better upperbound for the ratio than 5/2 as predicted by (3)? I am going to construct a linear combination of these two inequalities and next, using the triangle inequalities $d^F_{ij} \leq d^F_{il} + d^F_{lj}$ for $i, j, l = 1, \ldots, 5$ (which hold due to inequalities (2)), I intend to minimize the largest coeffficient of some d^F_{ij} term. Obviously, the largest coefficient in the resulting inequality determines an upper bound for the ratio. The problem of minimizing the largest coefficient can be casted into an LP-framework in the following way. Consider the following decision variables:

- z_{ij}: coefficient of d^F_{ij} in resulting inequality; $i, j = 1, \ldots, 5,\ i < j$,
- x_{ijl}: coefficient of triangle inequality $d^F_{ij} \leq d^F_{il} + d^F_{lj}$ for $i, j, l = 1, \ldots, 5,\ i < j, l \neq i, l \neq j$,
- α_1: coefficient of inequality of Lemma 1, and
- α_2: coefficient of inequality of Lemma 2.

Here is the LP-model called LP-5-DAPC:

$(LP-5-DAPC)$ minimize w

subject to

$$z_{12} = 4\alpha_1 + \frac{46}{9}\alpha_2 - x_{123} - x_{124} - x_{125} + x_{132} + x_{142} + x_{152} + x_{231} + x_{241} + x_{251}, \quad (7)$$

$$z_{13} = 3\alpha_1 + \frac{23}{9}\alpha_2 - x_{132} - x_{134} - x_{135} + x_{123} + x_{143} + x_{153} + x_{231} + x_{341} + x_{351}, \quad (8)$$

$$z_{14} = \frac{4}{3}\alpha_2 - x_{142} - x_{143} - x_{145} + x_{124} + x_{134} + x_{154} + x_{241} + x_{341} + x_{451}, \quad (9)$$

$$z_{15} = \frac{4}{3}\alpha_2 - x_{152} - x_{153} - x_{154} + x_{125} + x_{135} + x_{145} + x_{251} + x_{351} + x_{451}, \quad (10)$$

$$z_{23} = 3\alpha_1 + \frac{23}{9}\alpha_2 - x_{231} - x_{234} - x_{235} + x_{123} + x_{132} + x_{243} + x_{253} + x_{342} + x_{352}, \quad (11)$$

$$z_{24} = \frac{4}{3}\alpha_2 - x_{241} - x_{243} - x_{245} + x_{124} + x_{142} + x_{234} + x_{254} + x_{342} + x_{452}, \quad (12)$$

$$z_{25} = \frac{4}{3}\alpha_2 - x_{251} - x_{253} - x_{254} + x_{125} + x_{152} + x_{235} + x_{245} + x_{352} + x_{452}, \quad (13)$$

$$z_{34} = 6\alpha_1 + \frac{4}{3}\alpha_2 - x_{341} - x_{342} - x_{345} + x_{134} + x_{143} + x_{234} + x_{243} + x_{354} + x_{453}, \quad (14)$$

$$z_{35} = \frac{4}{3}\alpha_2 - x_{351} - x_{352} - x_{354} + x_{135} + x_{153} + x_{235} + x_{253} + x_{345} + x_{453}, \quad (15)$$

$$z_{45} = 4\alpha_1 - x_{451} - x_{452} - x_{453} + x_{145} + x_{154} + x_{245} + x_{254} + x_{345} + x_{354}, \quad (16)$$

$w \geq z_{ij}$ for all $i, j = 1, \ldots, 5, i < j$,

$\alpha_1 + \alpha_2 = 1$, and

all variables ≥ 0.

An explanation of the constraints of LP-5-DAPC is as follows. Consider equality (7). In this equality the coefficient of d_{12}^F in the final inequality (z_{12}) is determined. Now, obviously z_{12} must be equal to the linear combination of the inequalities of Lemma's 1 and 2 restricted to the term d_{12}^F $(4\alpha_1 + \frac{46}{9}\alpha_2)$ plus a term which indicates the "usage" of triangle inequalities in which d_{12}^F occurs. There are three triangle inequalities with d_{12}^F appearing on the left-hand side, namely $d_{12}^F \leq d_{13}^F + d_{23}^F$, $d_{12}^F \leq d_{14}^F + d_{24}^F$ and $d_{12}^F \leq d_{15}^F + d_{25}^F$, so the corresponding x-variables have coefficient -1, and there are six triangle inequalities with d_{12}^F appearing on the right-hand side (this is easily verified), so the corresponding x-variables have coefficient 1. In fact, each of the equalities (8)-(16) can be explained in a similar way. The remaining constraints of LP-5-DPAC are straightforward.

Solving this model yields $w = 2.301471$ and hence the following statement can be produced (cf. (1))

$$c(H^5) \leq 2.301471 \cdot \mathrm{OPT}(\mathcal{I}) \quad \text{for all } \mathcal{I}.$$

Notice that the upperbound for the ratio of H^5 with respect to 5-DAPC has decreased from 2.5 (see Inequality (3)) to 2.301471.

6 Discussion

My approach has focused exclusively on 5-DAPC. In order to generalize this approach to other values of k, one should be able to generalize the inequalities deduced in Lemma's 1 and 2. It turns out that this is possible, however, it seems out of the scope of this contribution to state these inequalities and prove their validity for general k. However, I couldn't resist constructing the corresponding LP's for $k = 6$ and 7 and solve them. It turned out that in each case there was an improvement (compared to (3)) of the upperbound for the ratio. So here is a table with the current best upper bounds for ratio's of heuristic H^k for k-DAPC for $k = 5, 6$ and 7.

k-DAPC	Upper bound for ratio
$k = 5$	2.301471
$k = 6$	2.699225
$k = 7$	3.062519

References

[1] Bandelt, H.-J., Y. Crama and F.C.R. Spieksma (1994), "Approximation algorithms for multi-dimensional assignment problems with decomposable costs", Discrete Applied Mathematics **49**, 25-50.

[2] Crama, Y. and F.C.R. Spieksma (1992), "Approximation algorithms for three- dimensional assignment problems with triangle inequalities", European Journal of Operational Research **60**, 273-279.

[3] Spieksma, F.C.R. (1992), "Assignment and scheduling algorithms in automated manufacturing", Ph. D. thesis of Maastricht University.

Part II

Stochastic Operations Research

Stochastic Operations Research: A Retrospective View

H.C. Tijms
Dept. of Econometrics
Free University Amsterdam
E-mail: tijms@econ.vu.nl

Abstract

This paper gives an overview of the contributions of young Dutch researchers to the field of stochastic operations research in the last decade. In particular, the contributions to the area of queueing, reliability and Markovian control are reviewed.

1 Introduction

Stochastic Operations Research has gone the last ten years through a flourishing-period of activity. In the last decade many significant contributions to the field have been made by several applied probability groups in the Netherlands. As has been recognized by the international research community in stochastic operations research, the scope of the influence of the Dutch contributions exceeds by far the size of the country. Partly this succes can be explained by the stimulating environment of cooperation between the several research groups. Cooperation that is reflected as well in the joint teaching program for PhD students in the LNMB. The purpose of this paper is to present a review of the main contributions made in the last ten years by junior researchers of the LNMB to the field of the stochastic operations research. The review will concentrate on contributions in the following areas:

a. Queueing

b. Reliability and Maintenance

c. Markov Decision Processes.

In each of these areas important contributions have been made by young Dutch researchers.

2 Queueing

The application of the mathematical theory of queues to performance analysis of practical problems has been much hampered by the computational untractability of the analytical results. However, since the early 1980's much progress has been made to the development of generally applicable and computationally tractable algorithms. Most of the algorithmic breakthroughs deal with multi-dimensional queueing processes. Such processes naturally arise in modern applications in computer, telecommunication and flexible manufacturing systems.

2.1 *Product-form solutions*

An important contribution of queueing network theory is that, under certain conditions, it is possible to obtain a simple exact solution for the joint distribution of the queue lengths at the various nodes in the form of a product-form solution. Important new insights in the product-form theory are provided by the studies of Boucherie (1992) and Smeitink (1992), where the first study also deals with product-form networks with positive and negative customers. The latter product-form networks have recently gained much interest in the performance analysis of reource request and allocation models. In general an obstacle for the practical application of product-form theory remains the computation of the normalization constant in the product-form solution because of the huge number of the states involved. Tractable algorithms for the computation of the normalization constant can only be given for special cases.

2.2 *Compensation approach*

The compensation approach has been developed by Adan (1991) for a class of two-dimensional, homogeneous random walks on the integer grid in the positive quadrant of the plane, where transitions to the north, north-east and east are not allowed. The development of this approach was motivated by the famous shortest-queue problem. The main idea of the compensation approach is

(a) characterize a set of product-form solutions which satisfy the equilibrium equations for the interior states

(b) try to construct a linear combination of these product-form solutions such that the equilibrium equations for the boundary states are also satisfied.

In addition to the shortest-queue problem, the compensation approach has been successfully applied to other problems as well including the multi-programming queue and the 2×2 clocked buffered switch of an interconnection network. In Van Houtum (1994) the compensation approach has been generalized to multi-dimensional random walks.

2.3 *Power-series algorithm*

The power-series algorithm is another numerical approach that can be used for multi-dimensional random walks, but its scope of application is actually much wider. The method was first developed by Hooghiemstra et al (1986) and later expanded by Blanc (1993) and his PhD students Van der Mei (1995) and Van den Hout (1996). It is based on power-series expansions of the state probabilities in terms of some parameter, usually the load of the system in queueing applications. The coefficients of the power-series expansions are computed by a recursive scheme. The power-series algorithm is a flexible method which is applicable to a wide class of multi-dimensional queueing systems including polling systems. In polling models several users compete for service by a single server who switches from one queue to another in order to provide service. This is a rich class of models having many practical applications. Many new insightful results for polling-systems are obtained in the studies of Groenendijk (1990) and Borst (1994). In addittion to polling models other examples of models to which the power-series algorithm can be applied include load-balancing models, coupled-processor models and parallel-processor models. The power-series algorithm works quite well for these multi-dimensional queueing models. However, in general convergence

properties and error estimates of the algorithm are still unknown. An interesting extension of the power-series algorithm to controlled Markov chains is given by Passchier (1996) who applied the algorithm to a controlled tandem queue amongst others.

2.4 *Geometric-tail approach*

The compensation approach and the power-series approach both have the nice feature that an infinite state space need not be truncated by brute-force. In practical applications brute-force truncations usually lead to very large state spaces and thus to very large computing times. Another approach that avoids brute-force truncation is the geometric-tail approach advocated in Tijms (1994). This approach has been successfully applied in the studies of Gouweleeuw (1996) and Van Eenige (1996) to a variety of discrete-time queueing systems. The geometric-tail approach was developed for one-dimensional infinite-state Markov chains. This simpleminded approach reduces the infinite set of equilibrium equations to a finite but small system of linear equations by using the geometric tail behaviour (if any) of the state probabilities. As a consequence of this tail behaviour, the numerical analysis of finite-capacity queues is sometimes easier through the analysis of the corresponding infinite-capacity queue when a proportionality relation holds between the state probabilities in the finite-capacity and infinite-capacity models, see Gouweleeuw (1996). The geometric-tail approach can sometimes also be applied to two-dimensional Markov chains whose state space is a semi-infinite strip in the plane and whose one-step transition probabilities satisfy a certain homogeneity condition. Then, by basic result from the theory of linear difference equations, the state probabilities of the interior states can be written as a finite linear combination of geometric distributions. The characterization of the state probabilities by Adan (1991) in his compensation approach is a natural extension of this result. In the two-dimensional model of the compensation approach both state variables are unbounded.

2.5 *Large-deviations technique*

It is only recently that the technique of large deviations has become an important tool to analyse queueing systems. The technique is used to estimate very small probabilities of "rare" events such as packet-loss in high-speed networks. The method finds expressions for rare event probabilities in queueing models which are the basis for the development of algorithms for executing quick simulation. The main idea of the approach is

(a) from large deviations the statistical behaviour of the system leading to the rare event is derived

(b) a twisted probability model of the system is constructed such that the rare event becomes most likely, and

(c) analytical verification that the variance reduction obtained in the simulation is asymptotically optimal.

Important contributions to rare event analysis and quick simulation for communication networks have been made in the study of Mandjes (1996). The contributions include rare event analysis for large communication systems with general Markov fluid sources, buffer and bandwidth allocation in ATM networks and call blocking in ATM networks. Similar models stemming from telecommunication applications were analysed in Awater (1994) by using more classical methods such as eigenvalue analysis.

2.6 *Complex-function methods*

Techniques from complex function theory have been used for a long time to obtain analytical results for queueing systems. These results are often in the form of transforms. It is widely recognized that the transforms are useful to do asymptotic analysis, but it was believed for a long time that the transforms are not useful for calculating numerical values. The conventional wisdom has been that computation of complex roots and the inversion of Laplace transforms are difficult and numerically ill-posed problems. However the studies of Regterschot (1986), De Klein (1988) and Van Ommeren (1989) show that for a broad class of queueing systems solution techniques from complex function theory such as Wiener-Hopf factorization lead to efficient and numerically stable algorithms. Amongst others the Wiener Hopf technique can be successfully used to solve the versatile single-server queueing model with Markov modulated arrivals. That model is frequently used in telecommunication applications. Many instances of the usefulness of numerical inversion methods for solving queueing systems are given in Gouweleeuw (1996).

3 Reliability and Maintenance

In the past numerous papers have appeared on reliability and availability analysis of repairable systems with standby redundancy. Nearly all of these papers dealt with steady-state availability, that is, the long-run fraction of time the system is available. Hardly any attention was paid to interval availability, that is, the probability distribution of the availability of the system over a given but finite period of time. Transient availability analysis is very important for practical applications, e.g. computer systems are sold with a gauranteed availability over a limited period of time and oil and gas contracts guarantee minimum levels of oil and gas deliveries over a certain period of time. Recently, practically useful approximations for interval availability distributions have been obtained in the studies of Van der Heijden (1992) and Smith (1997). The results obtained for the k-out-of-n model with cold or hot standby do not involve the ususal assumption of exponentially distributed life times and repair times. The approximations in Smith (1997) have the nice feature that the computation times do not explode when k and n increase.

Besides the performance analysis of reliability systems there has been done quite some research in recent years on maintenance optimization models with multiple components. Clearly, maintenance has a strong impact on reliability. Apart from improving system performance by preventive maintenance, considerable cost reductions can be achieved when preventive actions are made. Optimal decision rules and heuristic rules for the maintenance of various multi-component systems, have been developed in the studies Smeitink (1992), Vanneste (1992) and Wildeman (1996). An important model in the study of Smeitink (1992) is the model for ranking of maintenance activities in multi-component systems when there is only a limited opportunity for doing preventive maintenance. The studies of Vanneste (1992) and Wildeman (1996) deal amongst others with a maintenance model for coordinating maintenance activities and use Markov decision theory to obtain (sub)optimal rules for this model.

4 Markov Decision Processes

In the past decade there has been an impressive continuation of the research in the fields of Markov decision processes (MDP). The theoretical aspects of MDP are continuously stim-

ulated by new practical applications such as in the field of telecommunication and maintenance optimization. Markov decision methods were used in the studies of Vanneste (1992) and Awater (1994) to develop algorithms for specific applications in these fields. However, fundamental contributions to the theory of MDP were also made in the last decade. Controlled queueing models are particularly suited for showing MDP results. In the study of Spieksma (1990) new results are obtained for establishing the optimality of threshold policies in a variety of queueing systems with constrained admission control. In addition this study deals with the fundamental problem of ergodicity conditions in multi-dimension Markov chains with a view towards applications in queueing networks. The thesis of Koole (1992) is mainly concerned with scheduling problems of queues: to which servers should customers be assigned and to which queues should servers be assigned? Using dynamic programming the solution of an optimal control rule is established for a variety of queueing models including the practically important model with Markov driven arrivals. A relatively new development in Markov decision theory is the study of problems with partial information. Again such models are inspired by design problems in telecommunication. Contrary to MDP with full information, little is known about the existence and the form of optimal control rules. In the thesis of Loeve (1994) this question is addressed, where in particular much attention is paid to the so-called periodic policies for which the same decision prescription returns periodically. The thesis of Loeve (1994) also develops algorithms to compute optimal control rules and applies these algorithms to a variety of queueing models with partial information or decentralized control. Such models are becoming increasingly important in real-world applications. Undoubtedly, much further research on Markov decision models with partial information or decentralized control will be seen in the future.

References

[1] I.J.B.F. Adan. *A Compensation Approach for Queueing Problems.* PhD thesis. Eindhoven University of Technology, 1991.

[2] G. Awater. *Broadband Communication-Modelling, Analysis and Synthesis of an ATM Switching Element.* PhD thesis. Delft University of Technology, 1994.

[3] J.P.C. Blanc. Performance analysis and optimization with the power-series algorithm. In L. Donatiello and R.D. Nelson, editors. *Performance Evaluation of Computer and Communication Systems,* p. 53-80. North-Holland, Amsterdam, 1993.

[4] S.C. Borst. *Polling Systems.* PhD thesis, Tilburg University, 1994.

[5] R.J. Boucherie. *Product-Form in Queueing Networks.* PhD thesis, Vrije University, Amsterdam, 1992.

[6] S.J. De Klein. *Fredholm Integral Equations in Queueing Analysis.* PhD thesis. University of Utrecht, 1988.

[7] F.N. Gouweleeuw. *A General Approach to Computing Loss Probabilities in Finite-Buffer Queues.* PhD thesis. Vrije University, Amsterdam, 1996.

[8] W.P. Groenendijk. *Conservation Laws in Polling Systems.* PhD thesis. University of Utrecht, 1990.

[9] G. Hooghiemstra, M. Keane amd S. van der Ree. Power series for stationary distributions of coupled processor models. *SIAM Journal on Applied Mathematics* **48**, 1159-1166, 1988.

[10] G. Koole. *Stochastic Scheduling and Dynamic Programming.* PhD thesis. University of Leiden, 1992.

[11] J.A. Loeve. *Markov Decision Chains with Partial Information.* PhD thesis. University of Leiden, 1995.

[12] M.R.H. Mandjes. *Rare Event Analysis of Communication Networks.* PhD thesis. Vrije University, Amsterdam, 1996.

[13] O. Passchier. *The Theory of Markov Games and Queueing Control.* PhD thesis. University of Leiden, 1996.

[14] G.J.K Regterschot. *Wiener-Hopf Factorization Techniques in Queueing Models.* PhD thesis. Twente University of Technology, 1986.

[15] E. Smeitink. *Stochastic Models for Repairable Systems.* PhD thesis. Vrije University, Amsterdam, 1992.

[16] M.A.J. Smith. *On the Availability of Failure Prone Systems.* PhD thesis. Erasmus University, Rotterdam, 1997.

[17] F.M. Spieksma. *Geometrically Ergodic Markov Chains and the Optimal Control of Queues.* PhD thesis. University of Leiden, 1990.

[18] H.C. Tijms. *Stochastic Models.* Wiley, New York, 1994.

[19] S.G. Vanneste. *Maintenance Policies for Complex Systems.* PhD thesis. Tilburg University, 1992.

[20] M.J.A. Van Eenige. *Queueing Systems with Periodic Service.* PhD thesis. Eindhoven University of Technology, 1996.

[21] M.C. Van der Heijden. *Performance Analysis for Reliability and Inventory Models.* PhD thesis. Vrije University, Amsterdam, 1993.

[22] G.J. Van Houtum. *New Approaches for Multi-Dimensional Queueing Systems.* PhD thesis. Eindhoven University of Technology, 1995.

[23] W.B. Van den Hout. *The Power-Series Algorithm: A Numerical Approach to Markov Processes.* PhD Thesis. Tilburg University, 1995.

[24] R. Van der Mei. *Polling Systems and the Power-Series Algorithm.* PhD thesis. Tilburg University, 1995.

[25] J.C.W. Van Ommeren. *Asymptotic Analysis of Queueing Systems.* PhD thesis. Vrije University, Amsterdam, 1989.

[26] R. Wildeman. *The Art of Grouping Maintenance.* PhD thesis. Erasmus University, Rotterdam, 1996.

Performance Modelling and Analysis of an ATM-DQDB Interworking Unit

S. Aalto[a], J.L. van den Berg[b], E. Smeitink[b], J.T. Virtamo[a]
[a] *VTT Telecommunications, Otakaari 7 B, FIN-02150 Espoo, Finland*
[b] *Royal PTT Nederland NV, PTT Research, NL-2260 AK Leidschendam, The Netherlands*
E-mail: e.smeitink@research.kpn.com

Abstract

We study the receiving part (RIWU) of an interworking unit between ATM and DQDB networks carrying traffic in the direction ATM-to-DQDB. For connectionless (CL) traffic, two operating modes are possible: cell-to-slot interworking and frame interworking. Models are developed for the buffer occupancy in both of these modes, specifically taking into account the variable service rate available to the RIWU due to other traffic on the DQDB bus. In the cell-to-slot mode, the CL part of the RIWU is modelled as a fluid buffer with Markov modulated input and service processes roughly describing the burstiness of the cell arrival process and variations in the available bus capacity. In the frame interworking mode, we consider separately the read-in and read-out stages of the RIWU, the main emphasis lying in the read-out stage. The bandwidth sharing of the DQDB bus is described by a head-of-line processor sharing (PS-HOL) discipline. The models are used for the numerical studies of the buffer size required to guarantee a given maximum allowed loss probability.

1 Introduction

We consider the network scenario depicted in Fig. 1 where ATM and DQDB networks are connected by an interworking unit (IWU). Such an IWU has been designed and implemented by the RACE 2032 project COMBINE. The IWU consists of two unidirectional (DQDB-to-ATM and ATM-to-DQDB) parts called the sending and the receiving interworking unit (SIWU and RIWU), respectively. This paper concerns modelling of the operation of the RIWU for connectionless (CL) traffic. Problems related to the SIWU have been covered e.g. in [1]. Results of both of these studies have been presented in a COMBINE deliverable [2].

On the ATM side, the CL traffic is carried in a separate CL overlay network set up using semipermanent virtual paths (VP) as its links and CL servers (CLS) as its nodes. There are several incoming VPs connected to the CLS but only one between the CLS and the RIWU.

On the DQDB side, there are several stations connected to the buses (one for each direction), the RIWU being one of them. The other stations generate background traffic sharing the bandwidth of the bus with the traffic from the RIWU, according to the DQDB queue arbitrated (QA) access mechanism. The access delay for CL cells (segments) is not

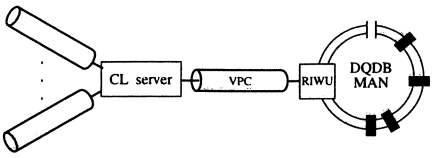


Figure 1: Physical model of a CL-network over ATM interconnected to a DQDB network through a RIWU.

guaranteed due to fluctuations in the number of requests for transmission from other DQDB stations.

On the frame level, the service rate as seen by the RIWU is variable. A buffer in the RIWU is required to accommodate incoming CL traffic bursts to the stochastically varying available bus capacity. Specifically, our aim in this paper is to analyse the content distribution of the RIWU buffer for dimensioning purposes. The interworking of CL traffic can be realized in two different modes: in cell-to-slot interworking ATM cells are directly converted to DQDB segments and in frame interworking full frames are first reassembled in the RIWU before being again segmented and forwarded to the DQDB. We develop models for both of these modes. An important feature in these models is that they take into account the variable service rate available to the RIWU on the DQDB bus, due to the bandwidth sharing with all other sources using the same bus. We do not explicitly consider the influence of connection oriented (CO) traffic on the handling of CL traffic. This influence can be approximately taken into account by subtracting the rate of CO traffic from the speed of the DQDB bus. The models for the two interworking modes will be presented and analysed separately in sections 2 and 3.

Main conclusions are given in section 4.

2 Cell-to-Slot RIWU

2.1 Model description

In cell-to-slot interworking the RIWU receives cells from the ATM network and puts these cells on the DQDB-bus, according to the DQDB QA access mechanism. The cell stream is modulated by the frame arrival process. Thus, on the frame level, it is natural to model the RIWU as a fluid buffer with Markov modulated input and service rates. The modulated service rate models the non-constant capacity of the DQDB-bus available for the RIWU. During periods in which the arrival rate is larger than the service rate, cells ("fluid") can be buffered. The capacity of the buffer is finite and will be denoted by B. Cells that cannot be buffered, due to lack of buffer space, are lost. Now, the problem is to find, for given input and service parameters, the minimal buffer size B such that the cell loss probability is less than a certain value P_{loss}.

To get a manageable (but still useful) model we assume that both the input and service processes are of *on-off* type. The choice of on-off arrivals is reasonable when frames arrive at the RIWU with fixed rate on a single incoming VP. The unit of time is chosen such that, in the on-state, the arrival rate is equal to 1. The service rate (in the on-state) is denoted by c. This parameter is also called *slow-down factor*.

In addition, the lengths of the off and on states of the arrival (service) process are assumed to be exponentially distributed with means $1/\lambda_1$ ($1/\lambda_2$) and $1/\mu_1$ ($1/\mu_2$), respectively. The parameters of the service process, λ_2 and μ_2, are specifically chosen to account for the influence of the background load on the RIWU. The choice is based on the assumption that, on the frame level, a DQDB bus with many stations operates according to the Processor Sharing (PS) discipline (see subsection 3.2). More precisely, the mean and the squared coefficient of variation of the time to serve a frame are chosen to be equal to the corresponding quantities in the PS queue. For a background load ρ_{bg} the mean and the squared coefficient of variation of the service time in the PS queue are given by $(L/C)/(1-\rho_{bg})$ and $(2+\rho_{bg})/(2-\rho_{bg})$, respectively, see e.g. [3]. Here L/C is the service time of a frame with average size L at full bus rate C (available to the CL traffic). The analysis in the next subsection, however, holds for general service parameters λ_2, μ_2.

2.2 Analysis

We follow closely the analysis of a similar model of a two-stage production line given in [4]. Describe the state of the system by the triple (a, b, x), with a the state of the arrival process ($a = 0$ means "off", $a = 1$ means "on"), b the state of the service process ($b = 0$ means "off", $b = 1$ means "on") and x the buffer occupancy. Now, we consider the behaviour of the system during a regeneration cycle with the regeneration points defined as the entrances in state $(0, 1, 0)$. Let P_T be the expected number of cells transmitted on the DQDB-bus during a regeneration cycle and T the expected cycle length. Then the cell loss probability P_{loss} is given by:

$$P_{loss} = 1 - \frac{P_T/T}{\lambda_1/(\lambda_1+\mu_1)}, \tag{1}$$

where P_T/T is the throughput and $\lambda_1/(\lambda_1+\mu_1)$ is the offered traffic.

In fact P_T and T can be seen as *costs* per cycle. Define $\alpha(x)$, $\beta(x)$, $\gamma(x)$ and $\delta(x)$ as the costs per unit of time in state $(1, 0, x)$, $(0, 1, x)$, $(0, 0, x)$ and $(1, 1, x)$, respectively. Then, if $\alpha(x) = \beta(x) = \gamma(x) = \delta(x) = 1$, for $0 \le x \le B$, the expected cost per cycle is equal to T. If $\alpha(x) = \gamma(x) = 0$ and $\beta(x) = \delta(x) = c$, for $0 < x \le B$, and $\alpha(0) = \beta(0) = \gamma(0) = 0$, $\delta(0) = \min(1, c)$, then the expected cost per cycle is equal to P_T.

To derive the expected cost per cycle, C_T, we introduce the functions $f(\cdot)$, $g(\cdot)$, $h(\cdot)$ and $l(\cdot)$ as follows. Let $f(x)$ be the expected cost until the end of the cycle if the system is in state $(1, 0, x)$, $0 \le x \le B$. The functions $g(x)$, $h(x)$, $l(x)$ are defined analogously for the case that the system is in state $(0, 1, x)$, $(0, 0, x)$, $(1, 1, x)$, respectively. Now, the expected cost per cycle can be written as

$$C_T = \frac{\beta(0)}{\lambda_1+\mu_2} + \frac{\mu_2}{\lambda_1+\mu_2}h(0) + \frac{\lambda_1}{\lambda_1+\mu_2}l(0). \tag{2}$$

To determine the functions f, g, h and l divide the costs into the costs during the first small time interval Δ and the costs during the rest of the cycle, and let $\Delta \to 0$. This leads to the following set of differential equations:

$$\begin{aligned}
-f'(x) &= \alpha(x) - (\lambda_2+\mu_1)f(x) + \mu_1 h(x) + \lambda_2 l(x), \quad 0 \le x < B, && (3)\\
cg'(x) &= \beta(x) - (\lambda_1+\mu_2)g(x) + \mu_2 h(x) + \lambda_1 l(x), \quad 0 < x < B, && (4)\\
0 &= \gamma(x) - (\lambda_1+\lambda_2)h(x) + \lambda_1 f(x) + \lambda_2 g(x), \quad 0 \le x \le B, && (5)\\
(c-1)l'(x) &= \delta(x) - (\mu_1+\mu_2)l(x) + \mu_1 g(x) + \mu_2 f(x), \quad 0 < x < B. && (6)
\end{aligned}$$

The boundary conditions obtained from (3), (4) and (6) are, respectively,

$$
\begin{aligned}
0 &= \alpha(B) - (\lambda_2 + \mu_1) f(B) + \mu_1 h(B) + \lambda_2 l(B), && (7)\\
0 &= g(0), && (8)\\
0 &= \delta(B) - (\mu_1 + \mu_2) l(B) + \mu_1 g(B) + \mu_2 f(B), \quad c \leq 1, && (9)\\
0 &= \delta(0) - (\mu_1 + \mu_2) l(0) + \mu_1 g(0) + \mu_2 f(0), \quad c \geq 1. && (10)
\end{aligned}
$$

Substitution of (5) into (3) and (4) leads, together with (6), to a set of three first order linear differential equations in the functions f, g and l with boundary conditions (7), (8) and (9) or (10). Assuming $c \neq 1$, this set can be written in matrix form as

$$\boldsymbol{f}'(x) = \boldsymbol{c}(x) + \boldsymbol{A}\boldsymbol{f}(x), \tag{11}$$

where $\boldsymbol{f}(x) := (f(x), g(x), l(x))^T$ and $\boldsymbol{f}'(x) := (f'(x), g'(x), l'(x))^T$. The matrix $\boldsymbol{A}$ and the vector $\boldsymbol{c}(x)$ are given by

$$\boldsymbol{A} = \begin{pmatrix} \lambda_2 + \frac{\lambda_2\mu_1}{\lambda_1+\lambda_2} & -\frac{\lambda_2\mu_1}{\lambda_1+\lambda_2} & -\lambda_2 \\ \frac{1}{c}\frac{\lambda_1\mu_2}{\lambda_1+\lambda_2} & -\frac{1}{c}\left(\lambda_1 + \frac{\lambda_1\mu_2}{\lambda_1+\lambda_2}\right) & \frac{\lambda_1}{c} \\ \frac{\mu_2}{c-1} & \frac{\mu_1}{c-1} & -\frac{(\mu_1+\mu_2)}{c-1} \end{pmatrix}, \tag{12}$$

$$\boldsymbol{c}(x) = \left(-\alpha(x) - \frac{\mu_1}{\lambda_1+\lambda_2}\gamma(x),\ \frac{1}{c}\left(\beta(x) + \frac{\mu_2}{\lambda_1+\lambda_2}\gamma(x)\right),\ \frac{1}{c-1}\delta(x)\right)^T. \tag{13}$$

The special case $c = 1$ can be shown (see [4]) to reduce to the solution of a simple first order linear differential equation and will not be considered here.

The general solution of the homogeneous equation (cf. (11)) is given by $C_1\boldsymbol{e}_1 + C_2\boldsymbol{e}_2 e^{\rho_2 x} + C_3\boldsymbol{e}_3 e^{\rho_3 x}$, where the ρ_i and $\boldsymbol{e}_i$ $(i = 1, 2, 3)$ are the different eigenvalues and corresponding eigenvectors of the matrix $\boldsymbol{A}$ and the C_i are arbitrary constants (note that $\rho_1 \equiv 0 \neq \rho_2, \rho_3$ and $\boldsymbol{e}_1 = (1,\ 1,\ 1)^T$). In the special case $\frac{\lambda_1}{\lambda_1+\mu_1} = c\frac{\lambda_2}{\lambda_2+\mu_2}$ the eigenvalues ρ_2 and ρ_3 are degenerate (equal). We shall not consider this case here; more about the analysis of this case can be found in [4].

For the cases $\alpha(x) = \beta(x) = \gamma(x) = \delta(x) = 1$ (to determine T) and $\alpha(x) = \gamma(x) = 0$, $\beta(x) = \delta(x) = c$ (to determine P_T) $\boldsymbol{c}(x)$ is independent of x. So, the solution of the inhomogeneous equation (cf. (11)) can be obtained in a standard way by using the solution of the homogeneous equation and the boundary conditions (7), (8) and (9) or (10).

2.3 Results on buffer dimensioning

The analysis of the fluid flow model described in the previous section yields explicit results for the overflow probability P_{loss}, for a given buffer capacity B. We have determined the allowed normalized load $\tilde{\rho}_0 = \rho_0/(1 - \rho_{bg})$ of the RIWU (ρ_0 is the partial load) as a function of the buffer capacity for a predefined loss level $P_{loss} = 10^{-4}$. (Note that in this section we are dealing with *cell* loss probabilities; the corresponding *frame* loss probabilities will be larger, but in general they will be of the same order of magnitude). We have investigated the influence of the slow-down factor c and the background load ρ_{bg}. Fig. 2 contains results for four cases which correspond to background load $\rho_{bg} = 0,\ 0.6$ and slow-down factor $c = 0.1,\ 0.8$. The results show a knee in the curves at $\tilde{\rho}_0 = 0.6 - 0.7$, i.e. a load up to 60 or 70 percent can be achieved with relatively small buffers, but from that point on little can

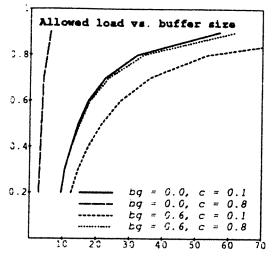


Figure 2: Allowed load at $P_{loss} = 10^{-4}$ as a function of the buffer size in the CL part of the cell-to-slot RIWU.

be gained with adding extra buffer space (note that this does not hold for the "optimistic" case with $\rho_{bg} = 0$ and $c = 0.8$).

Further, it is seen from the results in Fig. 2 that for a large slow-down factor ($c = 0.8$) the influence of the background load on the results is relatively large: the buffer space required to allow a load of 0.7 increases from 4 for the case with $\rho_{bg} = 0$ to 24 for the case $\rho_{bg} = 0.6$. For the case $c = 0.1$ this influence is much smaller, but still considerable: for allowed load 0.7 the required buffer space for $\rho_{bg} = 0,\ 0.6$ is 23 and 36, respectively. From the typical knee in the curves in Fig. 2 we conclude first that in general there is no sense in trying to achieve higher allowed loads than 0.7. Further, it is concluded that it is hardly possible to choose a (generally) proper value for the buffer space in the RIWU without further specifications w.r.t. the background load and the slow-down factor. However, as a small slow-down factor represents the worst case and a background load of $\rho_{bg} = 0.6$ on the DQDB network seems to be reasonable, it is recommended to choose the buffer size around 25-30 frames. With a mean frame length of 1500 bytes this is equivalent to 45 kbytes.

2.4 Further considerations

As we concluded from the numerical results, the required buffer space depends considerably on the slow-down factor c. This slow-down factor is mainly determined by the sum of the rates of the incoming VPs and the total rate of the CO traffic (the latter rate is subtracted from the DQDB-bus speed; hence it determines the maximum rate available for CL traffic). At this moment it is not clear what the practical values are. At least it seems that a slow down factor of less than 1 is rather conservative.

In our model we made the assumption that frames arrive as separate cell bursts (the "on-periods") with exponentially distributed intervals (the "off-periods"). In practice a cell burst may consist of several (interleaved) frames and frames may arrive in parallel when the RIWU has two or more incoming VPs for CL traffic. In these cases more buffer space is required (in fact the required buffer space is linear in the mean number of frames contained in a single burst).

3 Frame RIWU

In the frame interworking mode, the buffer of the RIWU consists logically of two separate stages, the read-in stage and the read-out stage. Physically, though, there is only a single buffer. Its content distribution can be calculated as the sum of the contents of the read-in and the read-out stages, which are approximately independent, cf. [1].

Our goal is to obtain the required buffer size of the RIWU such that the probability of frame loss, P_{loss}, is smaller than 10^{-4}. It depends, among other things, on the type of CLS, i.e. whether the CLS has buffers and the frames are reassembled in it or not. It also depends on the way memory is organised in the RIWU. The basic alternatives are static, dynamic and fully dynamic memory allocations, corresponding to a fixed maximum frame length reservation for each frame, reservation according to the actual frame size and reservation and release of memory as the cells flow in and out (cf. [1, 2]).

First we briefly present the results for the content distribution of the read-in stage, which rely on standard queueing theory and new results for the distribution of the amount of required work in the $M/M/1$ processor sharing queue. The emphasis of this section, however, is on the results for the content distribution of the read-out stage taking the background load on the DQDB network into account. Finally we combine these results to obtain the required total buffer size including both stages.

3.1 Analysis of read-in stage

We assume that over each path to the CLS frames arrive according to a Poisson process and that the total arrival rate of frames destined for the RIWU is λ. Further, all lengths of all frames coming to the CLS are exponentially distributed with mean L. Let ρ^{VP} denote the load on the output VP of the CLS, i.e.

$$\rho^{VP} = \frac{\lambda L}{C_{out}}, \tag{14}$$

where C_{out} denotes the capacity (speed) of the VP between the CLS and the RIWU. Besides the load ρ^{VP} and the memory organisation, the content distribution of the read-in stage depends on the characteristics of the CLS. We will consider two cases: a CLS with (practically) no buffers and no reassembly of frames and a CLS with buffers which reassembles frames before forwarding them to the RIWU.

CLS without buffers and no reassembly of frames. In this case the frames are transmitted on the output VP in parallel (interleaved). Since the CLS has (virtually) no buffers, it must send out frames at the same speed as it receives them. Thus, assuming that all frames arrive at the same speed C_{in}, at most $m = C_{out}/C_{in}$ frames can be sent out simultaneously. In this case the distribution p^{in} of the number of frames in the read-in phase at the RIWU equals the distribution of the number of frames in an $M/M/m/m$ queue (representing the output buffer of the CLS) with arrival rate λ and mean service time L/C_{in}. Due to the fact that frames are sent out independently of each other, the content distribution of the read-in stage for the three different memory organisations can be directly obtained from p^{in}.

CLS with frame buffering and reassembly. Next we consider the case of a CLS with buffers which reassembles frames. Now the number of frames which are in the read-in phase at the RIWU depends on the service strategy of the output buffer of the CLS. Extreme cases are

FIFO and PS disciplines. In the former there is at most one frame in the read-in phase, namely the one that is being sent by the CLS. In the latter all frames at the CLS that are destined to the RIWU are being transmitted simultaneously and thus are in the read-in phase at the RIWU.

We only consider the PS case (the FIFO case being trivial). The distribution p^{in} of the number of frames in the read-in phase at the RIWU now equals the distribution of the number of frames in an $M/M/1$ processor sharing queue with load ρ^{VP} given by (14). The content distribution of the read-in stage for the static memory organisation can be obtained directly from p^{in}. The content distribution for the dynamic (fully dynamic) memory organisation corresponds to the distribution of the total amount of required work (finished work) in the $M/M/1$ processor sharing queue. Explicit expressions for these distributions are derived in [2] (Vol II, Appendix D).

3.2 Analysis of read-out stage

To model the read-out stage, the QA access mechanism of the DQDB bus is assumed to approximate (on the frame level) an ideal head-of-line processor sharing (PS-HOL) discipline described e.g. in [5, 6]. If n stations are transmitting simultaneously, each of them gets one nth of the total bus capacity C (available to the CL traffic). The discrete time version of the model has been used in [7] to study mean packet delays in a symmetrically loaded multi-priority DQDB system.

Let us call the RIWU station 0 and denote the number of other stations by N. We assume that frames enter the output buffer of the RIWU according to a Poisson process with rate λ_0 and that the other stations together generate frames according to a Poisson process with rate λ_{bg} (background traffic). The lengths of all frames are exponentially distributed with mean L. As before, we denote the load generated by the RIWU and the other stations by $\rho_0 = \lambda_0 L/C$ and $\rho_{bg} = \lambda_{bg} L/C$, respectively, and the total load by $\rho_{tot} = \rho_0 + \rho_{bg}$. Further, let p_N^{out} denote the steady state distribution of the number of frames, K, in the read-out stage at the RIWU in a system with N other stations.

The total number of frames in the whole DQDB system behaves clearly as in an ordinary $M/M/1$ queue: the stationary distribution is geometric with parameter ρ_{tot}. In addition, the number of frames at each of the $N+1$ stations obviously constitute a $(N+1)$-dimensional Markov process. Unfortunately, there seems to be no closed form solution for the stationary probabilities even in the simplest case $N = 1$ and $\rho_0 = \rho_{bg}$. For any finite value of N the distribution p_N^{out} can, in principle, be obtained by numerically solving the forward Kolmogorov equations of the Markov process but the computational complexity becomes large when $N > 1$. However, in the case $N = \infty$ the state of the system is described by the number of frames at the RIWU and the total number of background frames, resulting in a two-dimensional Markov process.

Using successive overrelaxation (see e.g. [8]) we were able to obtain p_1^{out} and p_∞^{out}. The content distribution of the read-out stage for the different memory organisations can be easily obtained from p^{out}.

Worst case considerations. Before presenting the numerical results we wish to deduce the worst case background traffic conditions for the read-out buffer content with respect to the number of other stations, the distribution of the background load among these stations and the level of the background load.

First, one can easily show that, for given values ρ_0 and ρ_{bg} and a symmetric distribution

of the background load over the other stations, p_N^{out} is smaller in distribution than p_{N+1}^{out} for all $N \geq 0$. Thus, assuming an infinite number of other stations represents a worst case scenario for the loss probability in the RIWU. Note that the assumption of an infinite number of other stations implies that the transmission of each background frame starts immediately on the DQDB network and that p_∞^{out} does not depend on the distribution of the background load over the other stations.

Second, one may ask how a given total background load ρ_{bg} should be distributed among a finite number, N, of background stations to get highest losses. By heuristic arguments one can deduce that a symmetrically loaded system is worst in this sense. In fact, an asymmetric load is roughly equivalent to a symmetrical load with a reduced number of background stations. For instance, the extreme case where the background load is concentrated on a single station is precisely the symmetric case with $N = 1$. The conclusion then follows from the preceding result.

Third, the most tricky question is how the background ρ_{bg} load affects the losses for fixed normalized load $\tilde{\rho}_0$ and N (with a symmetric background system). Note that the total load $\rho_{tot} = \tilde{\rho}_0 + \rho_{bg}(1 - \tilde{\rho}_0)$ changes with ρ_{bg}. We use again heuristic reasoning and consider the behaviour of the *average* queue length for three different background loads. First, when $\rho_{bg} = 0$ the RIWU queue *is* an $M/M/1$ queue with the load $\rho_0 = \tilde{\rho}_0$. Next, when

$$\rho_{bg} = \rho_{bg}^{sym} =_{def} \frac{N\tilde{\rho}_0}{1 + N\tilde{\rho}_0} \tag{15}$$

the system is completely symmetric (including the RIWU): $\rho_0 = \rho_{bg}/N$. In this case the mean queue length, $\mathrm{E}[K^{sym}]$, of the RIWU read-out buffer is same as the mean total queue length of the whole DQDB system divided by $N + 1$ or, as is easily verified,

$$\mathrm{E}[K^{sym}] = \frac{\tilde{\rho}_0}{1 - \tilde{\rho}_0}, \tag{16}$$

i.e. the mean queue length of an $M/M/1$ queue with load $\tilde{\rho}_0$. Finally, consider the system with a higher background load $\rho_{bg} = N/(N + 1) > \rho_{bg}^{sym}$. In this case the load of the RIWU is $\rho_0 = \tilde{\rho}_0/(N + 1)$. Now, if the service rate of the RIWU were $C/(N + 1)$ (constant), the system would be equivalent to an $M/M/1$ queue with load $(N + 1)\rho_0 = \tilde{\rho}_0$. But as a minimum service rate $C/(N + 1)$ is always guaranteed by the PS-HOL discipline, this must be an easier case than $M/M/1$ queue with load $\tilde{\rho}_0$.

By the continuity, these observations suggest that, starting from $\rho_{bg} = 0$, the mean queue length first increases with ρ_{bg}, then returns to the same level when $\rho_{bg} = \rho_{bg}^{sym}$, and continues to decrease with higher background loads. This behaviour is verified by simulation results shown in Fig. 3, where $(\mathrm{E}[K] - \mathrm{E}[K^{sym}])/\mathrm{E}[K^{sym}]$ is plotted against $\tilde{\rho}_0$ for three different values of background load, $\rho_{bg} = 0.1,\ 0.3,\ 0.5$ with a) $N = 1$ and b) $N = 3$. Note that e.g. in the case $N = 1$ the curves $\rho_{bg} = 0.1$ and 0.3 indeed cross level 0 close to the theoretical crossing points of (15) $\tilde{\rho}_0 = 1/9 \approx 0.11$ and $3/7 \approx 0.43$, respectively. The curve $\rho_{bg} = 0.5$ stays below level 0 in the case $N = 1$ as it should as the load $N/(N + 1)$ was deduced to be an easier case than an $M/M/1$ queue with load $\tilde{\rho}_0$.

Thus for any finite value of N and given $\tilde{\rho}_0$ there is a worst case $\rho_{bg} < \rho_{bg}^{sym}$. In the limit $N \to \infty$ the upper bound tends to one, $\rho_{bg}^{sym} \to 1$. We conjecture that in this limit the buffer requirement increases with increasing background load (worst case ρ_{bg} tends to 1) for all $\tilde{\rho}_0$. An indication of this behaviour is given by Fig. 3.

Numerical results. In Fig. 4a the maximum allowed normalized load $\tilde{\rho}_0$ of the RIWU is plotted as a function of the buffer size. The number of background stations N is varied,

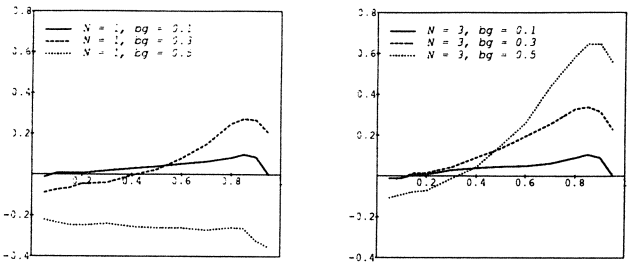


Figure 3: $(\mathrm{E}[K] - \mathrm{E}[K^{sym}])/\mathrm{E}[K^{sym}]$ as a function of the normalized load of the RIWU with different background loads and number of background stations: a) $N = 1$, b) $N = 3$.

$N = 1, 2, 3, \infty$, the background load being fixed $\rho_{bg} = 0.6$ and symmetric. The results for the intermediate values $N = 2$ and 3 were obtained using Monte Carlo simulation. As we deduced earlier, the buffer requirement becomes greater as the number of stations increases.

In Fig. 4b the maximum allowed normalized load $\tilde{\rho}_0$ is plotted as a function of the buffer size for different ρ_{bg} with $N = \infty$ (reflecting the worst case). We see that, for a given normalized load $\tilde{\rho}_0$, the required amount of memory indeed increases with the background load ρ_{bg} confirming our conjecture. This behaviour can also intuitively be explained by the fact that high background loads result in longer periods during which the service capacity available to the RIWU is small compared to the arrival rate of frames, i.e. during these periods the RIWU experiences (over)load.

It is not possible to dimension the buffer in such a way that a given normalized load of, say $\tilde{\rho}_0 = 0.8$, can be allowed for every possible background load. One has to choose a dimensioning point both for $\tilde{\rho}_0$ and ρ_{bg} based on a trade-off with a rapidly increasing buffer size if these values are pushed too close to one.

3.3 Results on buffer dimensioning

We now obtain the required buffer size of the RIWU for two different CLSs and for the three different memory organisations considered. The buffer of the RIWU is dimensioned such that in our models $P_{loss} < 10^{-4}$ in the following parameter range: $\rho^{VP} < 0.8$, $\tilde{\rho}_0 < 0.7$, $\rho_{bg} < 0.6$ and $N = \infty$ (worst case).

For the CLS with no buffer and no reassembly the required buffer sizes of the RIWU for the different memory organisations are:

Static memory allocation:	83	L_{max}	(763 kbytes)
Dynamic memory allocation:	106	L	(159 kbytes)
Fully dynamic memory allocation:	87	L	(131 kbytes)

where we have additionally assumed $m < 20$ (maximum number of interleaved frames). The corresponding numbers for the CLS which buffers frames and uses a processor sharing (PS)

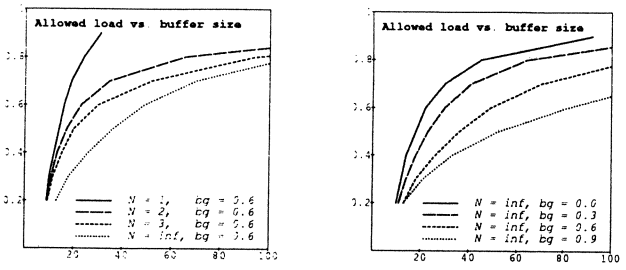


Figure 4: Allowed normalized load $\tilde{\rho}_0$ of the RIWU for $P_{loss} = 10^{-4}$ as a function of the read-out buffer size: a) N is a parameter and $\rho_{bg} = 0.6$; b) ρ_{bg} is a parameter and $N = \infty$. Fully dynamic memory allocation is assumed.

service strategy are:

Static memory allocation:	73	L_{max}	(671 kbytes)
Dynamic memory allocation:	99	L	(149 kbytes)
Fully dynamic memory allocation:	77	L	(116 kbytes)

The numbers in parentheses correspond to a mean frame size $L = 1500$ bytes and a maximum frame size $L_{max} = 9188$ bytes. In all cases the size of the buffer is mainly determined by the memory requirement of the read-out stage as can be seen by comparing the results to those in Fig. 4. The difference between the two CLS types, however, is due to the differences in the required read-in stage buffers.

4 Conclusions

We have analysed the buffer requirements of a RIWU carrying CL traffic from an ATM network to a DQDB network. Buffer size affects the allowable normalized load, consistent with a given loss probability, of the RIWU in relation to the mean bandwidth available at the DQDB bus (when CO traffic and the traffic of other stations has been subtracted). Increasing the buffer size is only effective up to a certain point determined by the knee in the allowable load vs. buffer size curve. This has been used as the basis for dimensioning of the buffers.

Several factors affect the analysis. The main distinction has been made between the cell-to-slot and frame operating modes of the RIWU, which lead to different modelling approaches. In the case of a frame mode RIWU, additionally, the operating mode of the CLS as well as the memory organization of the RIWU affect the buffer requirements. All the basic cases have been covered by our analysis.

The main emphasis in this work has been in modelling the effect of the variable capacity of the DQDB bus due to the traffic from other sources. We have concluded that worst case conditions arise when the number of other sources is very large and their load increases.

A crucial assumption in the analysis of the models presented in this paper is that of the Poissonian frame arrivals. An interesting topic for further research is to study the influence

of other (e.g. more bursty) frame arrival processes on buffer requirements of the RIWU. Finally, we note that basically the same analysis that has been presented in this paper applies to any station attached to the DQDB bus.

References

[1] S. Aalto, I. Norros, J. Virtamo, K. Kvols and S. Manthorpe, Performance aspects of streaming and message modes of interworking, Proceedings of the First International Symposium on Interworking, Bern, November 1992.

[2] I. Norros (editor), IWU Design Choices Based on Performance Analysis, Race COMBINE deliverable D4, R2032/KTA/WP1/DS/P/004/b1, 1993.

[3] T.J. Ott, The Sojourn-Time Distribution in the $M/G/1$ Queue with Processor Sharing, *Journal of Applied Probability* **21** (1984) 360–378.

[4] J. Wijngaard, The Effect of Interstage Buffer Storage on the Output of Two Unreliable Production Units in Series, with Different Production Rates, *AIIE Transactions* **11** (1979) 42–47.

[5] K.W. Fendick and M.A. Rodrigues, A Heavy-Traffic Comparison of Shared and Segregated Buffer Schemes for Queues with the Head-of-Line Processor-Sharing Discipline, *Queueing Systems* **9** (1991) 163–190.

[6] J.A. Morrison, Head of the Line Processor Sharing for Many Symmetric Queues with Finite Capacity, *Queueing Systems* **14** (1993) 215–237.

[7] P.G. Potter and M. Zukerman, A Discrete Shared Processor Model for DQDB, *Computer Networks and ISDN Systems* **20** (1990) 217–222.

[8] H.C. Tijms, Stochastic Modelling and Analysis, John Wiley, New York, 1986.

Explicit solutions for queueing problems

Ivo Adan
Eindhoven University of Technology
Department of Mathematics and Computer Science
E-mail: iadan@win.tue.nl

1 Introduction

Many queueing problems can be modeled as random walks on a multi-dimensional grid. A time-dependent analysis of such models appears to be possible only in rare cases, and even then the solutions are quite complicated. Therefore, in this paper attention is focussed upon the equilibrium behavior of these models, rather then upon their time-dependent behavior.

The equilibrium distribution of a random walk on a grid is the solution of a set of equilibrium equations. These equations can be viewed as difference equations. In the theory of differential equations, the continuous analogue of difference equations, a well-known solution approach is separation of variables (see e.g. Garabedian [11]). This method attempts to solve differential equations by constructing sums of product-form solutions. It seems natural to investigate whether it is also possible to solve equilibrium equations by sums of product-form solutions. And if so, under which conditions are such solutions feasible, and which techniques can be used to find such solutions? In the attempts to find sums of product forms as solution, three main directions may be distinguished, namely:

1. Exactly one product form as solution;
2. A finite sum of product forms as solution;
3. A countably infinite sum of product forms as solution.

The first direction is the oldest one. It has also been studied most systematically, see e.g. Jackson [14], Gordon and Newell [12], Baskett et al. [5], Kelly [15], Lavenberg and Reiser [17] and Van Dijk [9]. In [5] a practical characterization is given of queueing networks with a product-form solution. The conditions under which the solution is a single product form are rather severe, but, most strikingly, they do not depend on the dimension of the state space. This feature makes this product-form approach very important, since it is virtually the only more general approach for really complex queueing systems.

The other directions are newer and may be viewed as a generalization of the first one. The results in these directions are less systematic than the first one. This paper aims at reviewing some of the particular techniques required for the construction of sums of product form solutions.

An important application in the second direction is the multi-server queue with Erlang (or Cox) arrivals and services. Queueing problems of this type can be described as a random walk on a multi-dimensional grid which is unbounded in only one direction. In section 2 we will show that the queue length distribution can be written as a sum of products. To find

this solution we use a direct approach which is based on separation of variables. The third direction is based on the compensation approach. This approach has been developed for one rather general class of multi-dimensional random walks (unbounded in each direction) and for several related special cases. An important example in this direction is the shortest queue problem. In section 3 we will sketch the analysis of this problem to demonstrate the basic ideas of the compensation approach.

2 Finite sum of products: System with Erlang servers

In this section we study a system with c parallel identical servers and a common queue. The service times are Erlang-r distributed with mean r/μ. This means that a service has to go through up to r exponential stages, each with mean $1/\mu$. The service discipline is first-come first-served. Jobs arrive according to a Poisson stream with rate λ. This system can be modeled as a continuous-time Markov process with states $\boldsymbol{n} = (n_0, n_1, \ldots, n_c)$, where n_0 is the number of waiting jobs and n_i is the number of remaining service stages for server i, $i = 1, \ldots, c$. So n_0 ranges from 0 to ∞ and n_i, $i = 1, \ldots, c$, from 0 to r, where $n_i = 0$ is only possible if $n_0 = 0$. Note that completion of a service stage at server i leads to a departure if $n_i = 1$. The flow diagram for $c = 1$ is depicted in figure 1.

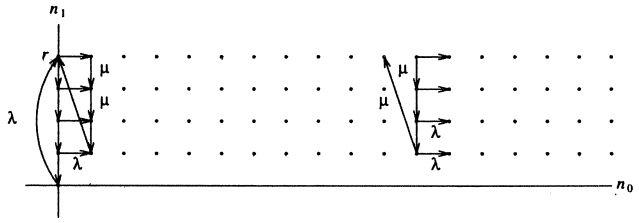


Figure 1: Flow diagram for the model with $c = 1$

Our aim is to determine the equilibrium probabilities $p(\boldsymbol{n})$. Once these probabilities are known we can compute performance characteristics such as, for example, the mean queue length and the mean waiting time. We will show that $p(\boldsymbol{n})$ can be expressed as a finite sum of products of powers. To do so we first introduce some notations and formulate the equilibrium equations for $p(\boldsymbol{n})$ that will be relevant to the analysis. Let $\boldsymbol{e}_i = (0, \ldots, 0, 1, 0, \ldots, 0)$ have $c+1$ components, with the one at the same place as n_i in $\boldsymbol{n}$ and let $\delta(n)$ be 1 if $n = r$ and 0 otherwise. By equating the rate out of and the rate into state $\boldsymbol{n}$ we obtain

$$p(\boldsymbol{n})(\lambda + c\mu) = p(\boldsymbol{n} - \boldsymbol{e}_0)\lambda + \sum_{i=1}^{c} p(\boldsymbol{n} + \boldsymbol{e}_i)\mu(1 - \delta(n_i)) + \sum_{i=1}^{c} p(\boldsymbol{n} + \boldsymbol{e}_0 - (r-1)\boldsymbol{e}_i)\mu\delta(n_i)\,, \tag{1}$$

which is valid for all states $\boldsymbol{n}$ with $n_0 \geq 1$. The equations (1) form the inner conditions, the equations in states with $n_0 = 0$ form the boundary conditions. The precise form of the boundary conditions is not relevant to the analysis, and therefore it is omitted.

The approach to solve the equilibrium equations will be based on separation of variables, an elementary approach for the solution of partial differential equations. Below we demonstrate this approach for a simple problem of conduction of heat (cf. Carslaw and Jaeger [8]). Then we will show how the same approach can be used to determine the probabilities $p(\boldsymbol{n})$.

An analogue: Conduction of heat in a thin rod
Consider the following problem (see also figure 2):

$$u_{xx} - u_t = 0, \qquad 0 < x < 1, t > 0, \tag{2}$$
$$u(0,t) = u_x(1,t) = 0, \qquad t \geq 0, \tag{3}$$
$$u(x,0) = v(x), \qquad 0 \leq x \leq 1. \tag{4}$$

The function $u(x,t)$ can be interpreted as the temperature in a thin rod along the interval $0 \leq x \leq 1$. The end $x = 0$ is maintained at zero temperature, while the end $x = 1$ is isolated (no flow of heat). The initial temperature at $t = 0$ is given by $v(x)$. This problem may be solved by first constructing solutions of the form

$$u(x,t) = X(x)T(t), \tag{5}$$

satisfying (2) and boundary conditions (3). By linearity of (2)–(3) any linear combination of solutions in this set satisfies these equations. The next step is to construct a linear combination which also satisfies initial condition (4). Substitution of (5) in (2)–(3) yields

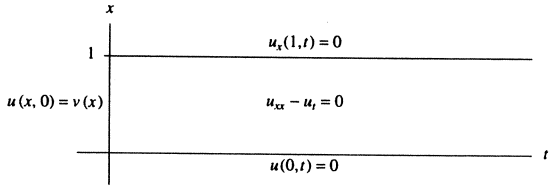


Figure 2: Equations for the temperature $u(x,t)$ in a thin rod

$$\frac{X''(x)}{X(x)} = \frac{T'(t)}{T(t)} = \text{constant} = \lambda, \qquad X(0) = X'(1) = 0,$$

where λ is the separation constant. Only for the values $\lambda = \lambda_j = -(j + 1/2)^2\pi^2$ with $j = 0, 1, \ldots$ these equations have a nontrivial solution, namely

$$u_j(x,t) = \sin(x\sqrt{-\lambda_j})e^{\lambda_j t}.$$

Then the solution of (2)–(4) can be written as a linear combination of functions in this set,

$$u(x,t) = \sum_{j=0}^{\infty} c_j u_j(x,t),$$

where the coefficients c_j follow from initial condition (4).

We now return to the queueing problem and try to use the same approach as above. In the first step we construct solutions of the form

$$p(\boldsymbol{n}) = F_0(n_0)F_1(n_1)\ldots F_c(n_c) \tag{6}$$

satisfying the equations (1). In the second step we use these functions to construct a linear combination also satisfying the conditions for $n_0 = 0$. Insertion of (6) in (1) yields

$$\lambda + c\mu = \frac{F_0(n_0 - 1)}{F_0(n_0)}\lambda + \sum_{i=1}^{c} \frac{F_i(n_i + 1)}{F_i(n_i)}\mu(1 - \delta(n_i)) + \sum_{i=1}^{c} \frac{F_0(n_0 + 1)}{F_0(n_0)} \frac{F_i(1)}{F_i(r)}\mu\delta(n_i).$$

Considering this equation for $n_i < r$ (so $\delta(n_i) = 0$) leads to the conclusion that

$$\frac{F_0(n_0-1)}{F_0(n_0)} = \text{constant} = \frac{1}{\alpha_0}, \qquad n_0 \geq 1,$$
$$\frac{F_i(n_i+1)}{F_i(n_i)} = \text{constant} = \alpha_i, \qquad n_i = 1,\ldots,r-1,\ i = 1,\ldots,c.$$

So the functions $F_i(n_i)$ are powers of α_i. Hence the solutions (6) are of the form

$$p(\boldsymbol{n}) = \alpha_0^{n_0} \cdots \alpha_c^{n_c},$$

where the (separation) constants α_i have to satisfy

$$\lambda + c\mu = \frac{\lambda}{\alpha_0} + \sum_{i=1}^{c} \alpha_i \mu + \sum_{i=1}^{c} (\frac{\alpha_0}{\alpha_i^r} - 1)\alpha_i \mu \delta(n_i)\,. \tag{7}$$

To satisfy this relation for all n_i the coefficients of the functions $\delta(n_i)$ must be zero, so we obtain that $\alpha_0 = \alpha_1^r = \ldots = \alpha_c^r$, or equivalently,

$$\alpha_0 = \alpha_1^r, \qquad \alpha_2 = x_2\alpha_1, \ldots, \alpha_c = x_c\alpha_1, \tag{8}$$

with $x_2^r = \ldots = x_c^r = 1$. Substitution of (8) in (7) yields the following equation for α_1:

$$\lambda + c\mu = \frac{\lambda}{\alpha_1^r} + \alpha_1\mu + \sum_{i=2}^{c} x_i \alpha_1 \mu. \tag{9}$$

The condition that the sum of all probabilities $p(\boldsymbol{n})$ is equal to 1 implies that only products with $|\alpha_0| < 1$, or equivalently, $|\alpha_1| < 1$ are useful. For each feasible choice of x_i it can be shown that equation (9) has exactly r roots α_1 with $|\alpha_1| < 1$ provided the utilization condition $\lambda r/\mu < c$ holds. So we find r^c products satisfying (1). We label these products $\alpha_{0,j}^{n_0} \cdots \alpha_{c,j}^{n_c}, j = 1,\ldots,r^c$. This concludes the first step. In the second step we express $p(\boldsymbol{n})$ as a linear combination of the products in this set. The number of unknown coefficients in this linear combination is sufficient to also satisfy the equilibrium equations for states $\boldsymbol{n}$ with $n_0 = 0$. Below we summarize our findings. Rigorous proofs may be found in [4].

Theorem 2.1 *Provided $\lambda r/\mu < c$, there exist coefficients c_j such that*

$$p(\boldsymbol{n}) = \sum_{j=1}^{r^c} c_j \alpha_{0,j}^{n_0} \cdots \alpha_{c,j}^{n_c}$$

for all $\boldsymbol{n}$ with $n_i \geq 1, i = 1,\ldots,c$.

Remark 2.2 *(Extensions)* The approach demonstrated in this section also works in case of Erlang distributed interarrival times, see [4]. A similar technique, based on separation of variables, has been applied by Bertsimas [7, 6] to solve the $E_k/C_2/s$ and $C_k/C_r/s$ system, respectively.

Remark 2.3 *(Matrix-geometric approach)* The equilibrium distribution can also be represented in a matrix-geometric form (see e.g. chapter 3 in Neuts [19]). There is a close relation between Theorem 2.1 and this representation. Namely, the factors $\alpha_{0,j}$ are the eigenvalues of the rate matrix and the products $\alpha_{1,j}^{n_1} \cdots \alpha_{c,j}^{n_c}$ are the associated eigenvectors.

3 Infinite sum of products: Shortest queue problem

The system we consider in this section consists of two parallel and identical servers, each with its own queue (see figure 3(a)). The service times are exponentially distributed with mean 1. Jobs arrive according to a Poisson stream with rate 2ρ where $0 < \rho < 1$. On arrival a job joins the shortest queue, and if queues have equal length, joins either queue with probability 1/2. The state of the system can be described by the pair (m, n) where m is the length of the shortest queue and n is the difference between the longest and shortest queue. The flow diagram is shown in figure 3(b). Let $p(m, n)$ be the equilibrium probability for state (m, n). The object in this section is the determination of $p(m, n)$.

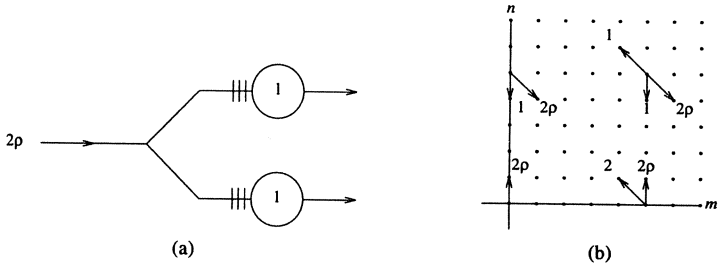


Figure 3: (a) Shortest queue system (b) Flow diagram

The equilibrium equations are given below. In these equations we have eliminated the probabilities $p(m, 0)$ from (11) and (13) by substituting (14)–(15). This is done to simplify the presentation. The analysis can now be restricted to the probabilities $p(m, n)$ with $n > 0$ satisfying (10)–(13). The equations (14)–(15) may be treated as definition for $p(m, 0)$.

$$
\begin{aligned}
p(m,n)2(\rho+1) &= p(m-1,n+1)2\rho + p(m,n+1) + p(m+1,n-1), \\
&\qquad m>0, n>1, && (10)\\
p(m,1)2(\rho+1) &= p(m-1,2)2\rho + p(m,2) \\
&\quad +(p(m,1)2\rho + p(m+1,1))\frac{1}{\rho+1} \\
&\quad +(p(m-1,1)2\rho + p(m,1))\frac{\rho}{\rho+1}, \qquad m>0, && (11)\\
p(0,n)(2\rho+1) &= p(0,n+1) + p(1,n-1), \qquad n>1, && (12)\\
p(0,1)(2\rho+1) &= p(0,2) + (p(0,1)2\rho + p(1,1))\frac{1}{\rho+1} + p(0,1), && (13)\\
p(m,0)2(\rho+1) &= p(m-1,1)2\rho + p(m,1), \qquad m>0, && (14)\\
p(0,0)2\rho &= p(0,1). && (15)
\end{aligned}
$$

The usual approaches to solve the equilibrium equations are based on generating functions (see e.g. [16, 10]). In this section we present an approach which directly tries to solve the equations. The idea is similar to the one in the previous section. We first try to find a set of products $\alpha^m\beta^n$ satisfying the inner conditions (10). Then we use the products in this set to construct a linear combination which also satisfies the boundary conditions (11)–(13). The first part is easy. Substituting $\alpha^m\beta^n$ into (10) and then dividing by common factors yield a quadratic equation for α and β (see figure 4).

Lemma 3.1 *The product $\alpha^m\beta^n$ satisfies (10) if*

$$\alpha\beta 2(\rho+1) = \beta^2 2\rho + \alpha\beta^2 + \alpha^2. \tag{16}$$

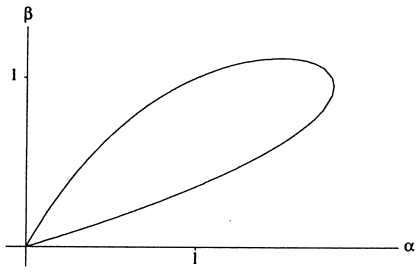


Figure 4: Curve (16) characterizing the set of products $\alpha^m\beta^n$ satisfying (10)

The problem we are now facing is different from the one in the previous section. There we found a finite set of products satisfying the inner conditions and these products were all used to construct the solution. Now we have a continuum of products satisfying (10). How do we select the appropriate products from this set? The selection is based on a compensation idea (which explains the name of the approach). This idea has an interesting analogue in electrostatics, where it is known as the method of images.

An analogue: Potential problem of conducting spheres

Consider two non intersecting conducting spheres, whose centers are A and B, their radii a and b and their potentials Φ_a and 0, respectively. Suppose that their distance of centers is c (see figure 5). Below we show how the potential Φ outside the spheres can be found by the method of images (see e.g. Maxwell [18]).

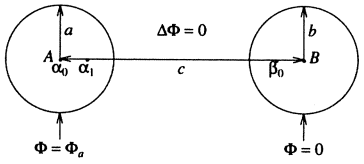


Figure 5: Potential problem of two non intersecting conducting spheres

If the spheres did not influence each other ($c = \infty$), then the potential Φ is that of point charge $\alpha_0 = a\Phi_a$ located at A. However, since c is finite, the potential does not vanish on sphere B. Therefore we place inside sphere B a new point charge β_0 at distance c_0 from B on the ray AB, and choose β_0 and c_0 such that the sum of the potentials of the charges α_0 and β_0 vanishes on sphere B. Note that the charges must be placed inside the spheres, since their potentials must be solutions to the Laplace equation outside the spheres. But, by adding charge β_0 we alter the potential on sphere A. To keep that potential unaltered we again place inside sphere A a point charge α_1 at distance d_1 from A on the ray AB, and choose α_1 and d_1 such that the potential of α_1 and β_0 vanishes on sphere A. In doing so we altered the potential on sphere B, and so on. We keep on adding point charges inside sphere A and B to alternately satisfy the boundary conditions on the two spheres. This results in

an infinite sequence of point charges. The value of Φ outside the two spheres is given by the sum of the potentials of these charges.

We will now use the same approach as above to find the probabilities $p(m,n)$. The products $\alpha^m\beta^n$ satisfying (10) play the role of the point charges inside the spheres. The conditions (11)–(13) on the horizontal and vertical boundary act as the boundary conditions on the two spheres.

The starting solution, which is correct far away from the vertical boundary, is given by

$$p(m,n) = \alpha_0^m \beta_0^n \tag{17}$$

where $\alpha_0 = \rho^2$, $\beta_0 = \rho^2/(2+\rho)$. This solution satisfies the inner conditions (10) and the horizontal boundary conditions (11). But it violates the vertical boundary conditions (12). Therefore we add a product $c_1\alpha_1^m\beta_1^n$ to (17) and choose c_1, α_1 and β_1 with α_1, β_1 satisfying (16) such that the sum

$$p(m,n) = \alpha_0^m \beta_0^n + c_1\alpha_1^m\beta_1^n$$

satisfies (12). But the new term violates the horizontal boundary conditions (11). So we add again a product $c_2\alpha_2^m\beta_2^n$ and so on. We keep on adding products, each one satisfying (10), so as to alternately satisfy the two boundary conditions. This results in an infinite sum of products. The sum is a formal solution of the equilibrium equations. What remains is the proof of convergence. This can be found in [1, 3]. The conclusion is formulated below.

Theorem 3.2 *There exist products $\alpha_i^m\beta_i^n$ satisfying (10) and coefficients c_i such that the equilibrium probabilities $p(m,n)$ can be expressed as*

$$p(m,n) = \sum_{i=0}^{\infty} c_i \alpha_i^m \beta_i^n, \qquad m \geq 0, n > 0.$$

Remark 3.3 *(Explicit determination of α_i, β_i, c_i)* The α_i, β_i, c_i mentioned in Theorem 3.2 can be solved explicitly, see [3]. Hence Theorem 3.2 provides an explicit characterization of $p(m,n)$. And based on the expression for $p(m,n)$ similar expressions may be derived for performance characteristics, such as the mean waiting time and mean queue lenghts.

Remark 3.4 *(General result)* Above we developed an approach to solve the shortest queue problem. But what is the scope of this approach? In [2, 3] it has been applied to a class of two dimensional Markov processes on the lattice in the positive quadrant of $\mathbb{R}^2$. For the processes in this class the transition rates are restricted to neighboring states and they are constant in the interior points and also constant on each of the axes (see figure 6). It appears that the equilibrium probabilities $p(m,n)$ can be expressed as an infinite sum of products, which can be found by the compensation approach, provided

$$q_{0,1} = q_{1,1} = q_{1,0} = 0. \tag{18}$$

So there may be no transitions in the interior points to the North, North-East and East. This result can be extended to processes of dimension 3 or higher (see Van Houtum [13]).

References

[1] I.J.B.F. Adan, J. Wessels, W.H.M. Zijm, *Analysis of the symmetric shortest queue problem*, Stochastic Models, 6 (1990), pp. 691–713.

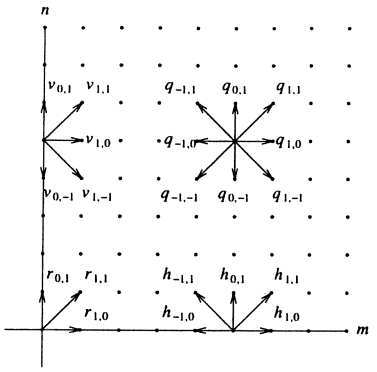


Figure 6: Flow diagram for a Markov process with constant rates and transitions restricted to neighboring states. $q_{i,j}$ is the transition rate from (m,n) to $(m+i,n+j)$ with m, $n > 0$ and a similar notation is used for the transition rates on each of the axes

[2] I.J.B.F. ADAN, J. WESSELS, W.H.M. ZIJM, *A compensation approach for two-dimensional Markov processes*, Adv. Appl. Prob., 25 (1993), pp. 783–817.

[3] I.J.B.F. ADAN, *A compensation approach for queueing problems*, CWI Tract 104, Stichting Mathematisch Centrum, Amsterdam, 1994.

[4] I. J. B. F. ADAN, W. A. VAN DE WAARSENBURG, J. WESSELS, *Analyzing $E_k|E_r|c$ queues*, EJOR, 92 (1996), pp. 112–124.

[5] F. BASKETT, K. M. CHANDY, R. MUNTZ, J. PALACIOS, *Open, closed and mixed networks of queues with different classes of customers*, J. Assoc. Comput. Mach., 22 (1975), pp. 248–260.

[6] D. BERTSIMAS, X.A. PAPACONSTANTINOU *Analysis of the stationary $E_k/C_2/s$ queueing system*, EJOR, 37 (1988), pp. 272–287.

[7] D. BERTSIMAS, *An analytic approach to a general class of $G/G/s$ queueing systems*, Opns. Res., 38 (1990), pp. 139–155.

[8] H. S. CARSLAW, J. C. JAEGER, *Conduction of heat in solids*, Oxford, 1959.

[9] N. M. VAN DIJK, *Queueing networks and product forms: a systems approach*, Wiley, Chichester, 1993.

[10] L. FLATTO, H.P. MCKEAN, *Two queues in parallel*, Comm. Pure Appl. Math., 30 (1977), pp. 255–263.

[11] P. F. GARABEDIAN, *Partial differential equations*, Wiley, Chichester, 1967.

[12] W. J. GORDON, G. F. NEWELL, *Closed queueing systems with exponential servers*, Opns. Res., 15 (1967), pp. 254–265.

[13] G. J. VAN HOUTUM, *New approaches for multi-dimensional queueing systems*, Thesis, Eindhoven University of Technology, Eindhoven, 1995.

[14] J. R. JACKSON, *Networks of waiting lines*, Opns. Res., 5 (1957), pp. 518–521.

[15] F. P. KELLY, *Reversibility and stochastic networks*, Wiley, New York, 1979.

[16] J.F.C. KINGMAN, *Two similar queues in parallel*, Ann. Math. Statist., 32 (1961), pp. 1314–1323.

[17] S. S. LAVENBERG, M. REISER, *Stationary state probabilities at arrival instants for closed queueing networks with multiple types of customers*, J. Appl. Prob., 17 (1980), pp. 1048–1061.

[18] J. C. MAXWELL, *A treatise on electricity and magnetism*, Clarendon Press, Oxford, 1904.

[19] M. F. NEUTS, *Matrix-geometric solutions in stochastic models*, Johns Hopkins University Press, Baltimore, 1981.

On the transition matrix of Markov chains obtained via cyclic mappings

Richard J. Boucherie*
Department of Econometrics
Universiteit van Amsterdam
E-mail: boucheri@fee.uva.nl

Abstract

This note shows that the transition matrix of the restriction of a Markov chain to a subset A of the state space S can be obtained from the transition matrix of the original Markov chain at S if a cyclic mapping can be defined. The transition matrix of the restricted process can then be expressed as a sum of terms all obtained from the original Markov chain.

1 Introduction

Recently, the transient behaviour of Markov chains has regained considerable attention. In particular, the transition matrix of queueing networks consisting of infinite-server queues has been analyzed in great detail (cf. Massey and Whitt [7]). These results extend previous results on these networks (Foley [4], Harrison and Lemoine [5], Kingman [6]) to more general arrival and service processes. The result of [4, 5, 6, 7] is that the transition matrix for the number of customers present at the stations of a queueing network of infinite-server queues is of product-form. In contrast, Boucherie and Taylor [2] show that the results of [7] cannot be extended to more general queueing networks: a transient product-form for queueing networks with interaction between the queues can be obtained for networks of infinite-server queues, only. Therefore, closed form results for more complicated queueing networks require different solution concepts.

Boucherie [1] has shown that the transient distribution of the Engset loss model with s servers and $N = 2s + 1$ sources can be expressed as a sum of two product-form distributions arising in the model with N servers instead of s. This note provides a theoretical motivation of this result and extends it to sums containing multiple terms. The result of this note is particularly useful when the transition matrix for the unconstrained Markov chain at state space S can be easily obtained. Then, under the conditions provided in this note, the transition matrix for the constrained case with state space $A \subset S$ can be derived from the transition matrix at S as a sum of terms all obtained from the transition matrix at S.

*The research of R.J. Boucherie has been made possible by a fellowship of the Royal Netherlands Academy of Arts and Sciences. Part of this work was carried out while the author was ERCIM fellow at INRIA Centre Sophia Antipolis, France.

2 Model and results

Consider a time-homogeneous, conservative, stable, regular, continuous-time Markov chain $\mathbf{X} = \{X(t),\ t \geq 0\}$ at countable state space S, with q-matrix $Q = (q(\mathbf{n}, \mathbf{n}'),\ \mathbf{n}, \mathbf{n}' \in S)$, where $q(\mathbf{n}, \mathbf{n}) = -\sum_{\mathbf{n}' \neq \mathbf{n}} q(\mathbf{n}, \mathbf{n}')$. Under these conditions the transition matrix

$$P(\mathbf{n}'', \mathbf{n}; t) = \mathbf{P}\{X(t) = \mathbf{n} | X(0) = \mathbf{n}''\},\quad \mathbf{n}, \mathbf{n}'' \in S,\ t \geq 0,$$

is the unique solution of the Kolmogorov forward equations (cf. Chung [3, Theorem II.18.3]), for $\mathbf{n}, \mathbf{n}'' \in S$,

$$\frac{dP(\mathbf{n}'', \mathbf{n}; t)}{dt} = \sum_{\{\mathbf{n}' \in S,\ \mathbf{n}' \neq \mathbf{n}\}} \{P(\mathbf{n}'', \mathbf{n}'; t) q(\mathbf{n}', \mathbf{n}) - P(\mathbf{n}'', \mathbf{n}; t) q(\mathbf{n}, \mathbf{n}')\}. \tag{1}$$

Definition 2.1 (Restriction) *The restriction of* $\mathbf{X}$ *to* $A \subset S$ *is the Markov chain* $\mathbf{X}_A = \{X_A(t),\ t \geq 0\}$ *at state space* A *with* q*-matrix* $Q_A = (q_A(\mathbf{n}, \mathbf{n}'),\ \mathbf{n}, \mathbf{n}' \in A)$ *given by*

$$q_A(\mathbf{n}, \mathbf{n}') = \begin{cases} q(\mathbf{n}, \mathbf{n}'), & \text{if } \mathbf{n} \neq \mathbf{n}', \\ -\sum_{\mathbf{n}^* \in A,\ \mathbf{n}^* \neq \mathbf{n}} q(\mathbf{n}, \mathbf{n}^*), & \text{if } \mathbf{n} = \mathbf{n}'. \end{cases} \tag{2}$$

Note that $\{X_A(t),\ t \geq 0\}$ is conservative, stable and regular. Let $P_A(\mathbf{n}'', \mathbf{n}; t)$ denote its transition matrix.

Definition 2.2 (Cyclic mapping) *Let* $\{A_k\}_{k=1,\ldots,K}$ *be a partition of* S*:* $A_i \cap A_j = \emptyset$, $i \neq j$, $\cup_{k=1}^{K} A_k = S$. *A mapping* $F : S \to S$ *is a cyclic mapping of order* K *of the Markov chain* $\mathbf{X}$ *when* F *has the following properties:*

$$F \text{ is injective and surjective;} \tag{3}$$

$$q(\mathbf{n}, \mathbf{n}') = q(F(\mathbf{n}), F(\mathbf{n}')),\quad \mathbf{n}, \mathbf{n}' \in S; \tag{4}$$

$$F(A_k) = A_{k+1},\ k = 1, \ldots, K-1,\quad F(A_K) = A_1. \tag{5}$$

The following lemma is an immediate result of (3) and (4) above. In combination with (5), the lemma shows that the equilibrium distribution on the sets in the partition must be equal when $\mathbf{X}$ has a cyclic mapping.

Lemma 2.3 (Equilibrium) *If* $\mathbf{X}$ *is ergodic with cyclic mapping* F*, then the equilibrium distribution,* $\pi = (\pi(\mathbf{n}),\ \mathbf{n} \in S)$*, satisfies*

$$\pi = \pi \circ F.$$

Proof If $\mathbf{X}$ is ergodic then the equilibrium distribution exists and is the unique distribution that satisfies the global balance equations

$$\sum_{\{\mathbf{n}' \in S,\ \mathbf{n}' \neq \mathbf{n}\}} \{\pi(\mathbf{n}') q(\mathbf{n}', \mathbf{n}) - \pi(\mathbf{n}) q(\mathbf{n}, \mathbf{n}')\} = 0,\quad \mathbf{n} \in S.$$

Insertion of $\nu(\mathbf{n}) = \pi(F(\mathbf{n}))$ into this equation gives, for $\mathbf{n} \in S$,

$$\begin{aligned} &\sum_{\{\mathbf{n}' \in S,\ \mathbf{n}' \neq \mathbf{n}\}} \{\nu(\mathbf{n}') q(\mathbf{n}', \mathbf{n}) - \nu(\mathbf{n}) q(\mathbf{n}, \mathbf{n}')\} \\ &\overset{(4)}{=} \sum_{\{\mathbf{n}' \in S,\ \mathbf{n}' \neq \mathbf{n}\}} \{\pi(F(\mathbf{n}')) q(F(\mathbf{n}'), F(\mathbf{n})) - \pi(F(\mathbf{n})) q(F(\mathbf{n}), F(\mathbf{n}'))\} \\ &\overset{(3)}{=} \sum_{\{\mathbf{n}^* \in S,\ \mathbf{n}^* \neq \mathbf{n}^{**}\}} \{\pi(\mathbf{n}^*) q(\mathbf{n}^*, \mathbf{n}^{**}) - \pi(\mathbf{n}^{**}) q(\mathbf{n}^{**}, \mathbf{n}^*)\} = 0, \end{aligned}$$

where $\mathbf{n}^{**} \in S$ is the unique state such that $F(\mathbf{n}) = \mathbf{n}^{**}$. Unicity of π completes the proof. □

The following theorem relates the transition matrix of the process restricted to A to the transition matrix of the original process.

Theorem 2.4 *Let $\{A_k\}_{k=1,\ldots,K}$ be a partition of S, and let F be a cyclic mapping of order K. Assume that for $\mathbf{n} \in A_1$, $\mathbf{n}' \notin A_1$,*

$$q(\mathbf{n}, \mathbf{n}') > 0 \Longleftrightarrow \mathbf{n}' = F(\mathbf{n}) \text{ or } \mathbf{n}' = F^{-1}(\mathbf{n}). \tag{6}$$

If $\mathbf{P}\{X(0) \in A_1\} = 1$, the restriction $\mathbf{X}_{A_1}$ of $\mathbf{X}$ to A_1 has transition matrix

$$P_{A_1}(\mathbf{n}'', \mathbf{n}; t) = \sum_{k=1}^{K} P(\mathbf{n}'', F^k(\mathbf{n}); t), \quad \mathbf{n}, \mathbf{n}'' \in A_1, \ t \geq 0, \tag{7}$$

where F^k is the k-fold convolution of F with itself.

Proof Since $\mathbf{X}_{A_1}$ is conservative, stable and regular, the transition matrix is the unique solution of the Kolmogorov forward equations (1) at A_1. Denote $A_0 := A_K$, and $F^0 := F^K$. Insertion of Q_{A_1} and P_{A_1} into the Kolmogorov forward equations gives for $\mathbf{n} \in A_1$, $t > 0$,

$$\begin{aligned}
\frac{dP_{A_1}(\mathbf{n}'', \mathbf{n}; t)}{dt} &\overset{(7)}{=} \sum_{k=1}^{K} \frac{dP(\mathbf{n}'', F^k(\mathbf{n}); t)}{dt} \\
&\overset{(1)}{=} \sum_{k=1}^{K} \sum_{\{\mathbf{n}' \in S,\ \mathbf{n}' \neq F^k(\mathbf{n})\}} \Big\{ P(\mathbf{n}'', \mathbf{n}'; t) q(\mathbf{n}', F^k(\mathbf{n})) - P(\mathbf{n}'', F^k(\mathbf{n}); t) q(F^k(\mathbf{n}), \mathbf{n}') \Big\} \\
&\overset{(6)}{=} \sum_{k=1}^{K} \sum_{\{\mathbf{n}' \in A_k,\ \mathbf{n}' \neq F^k(\mathbf{n})\}} \Big\{ P(\mathbf{n}'', \mathbf{n}'; t) q(\mathbf{n}', F^k(\mathbf{n})) - P(\mathbf{n}'', F^k(\mathbf{n}); t) q(F^k(\mathbf{n}), \mathbf{n}') \Big\} \\
&+ \sum_{k=1}^{K} \sum_{\mathbf{n}' \in A_{k+1}} \Big\{ P(\mathbf{n}'', \mathbf{n}'; t) q(\mathbf{n}', F^k(\mathbf{n})) - P(\mathbf{n}'', F^k(\mathbf{n}); t) q(F^k(\mathbf{n}), \mathbf{n}') \Big\} \\
&+ \sum_{k=1}^{K} \sum_{\mathbf{n}' \in A_{k-1}} \Big\{ P(\mathbf{n}'', \mathbf{n}'; t) q(\mathbf{n}', F^k(\mathbf{n})) - P(\mathbf{n}'', F^k(\mathbf{n}); t) q(F^k(\mathbf{n}), \mathbf{n}') \Big\} \\
&= \sum_{k=1}^{K} \sum_{\{\mathbf{n}' \in A_1,\ \mathbf{n}' \neq \mathbf{n}\}} \Big\{ P(\mathbf{n}'', F^k(\mathbf{n}'); t) q(F^k(\mathbf{n}'), F^k(\mathbf{n})) - P(\mathbf{n}'', F^k(\mathbf{n}); t) q(F^k(\mathbf{n}), F^k(\mathbf{n}')) \Big\} \\
&+ \sum_{k=1}^{K} \sum_{\mathbf{n}' \in A_1} \Big\{ P(\mathbf{n}'', F^{k+1}(\mathbf{n}'); t) q(F^{k+1}(\mathbf{n}'), F^k(\mathbf{n})) - P(\mathbf{n}'', F^k(\mathbf{n}); t) q(F^k(\mathbf{n}), F^{k+1}(\mathbf{n}')) \Big\} \\
&+ \sum_{k=1}^{K} \sum_{\mathbf{n}' \in A_1} \Big\{ P(\mathbf{n}'', F^{k-1}(\mathbf{n}'); t) q(F^{k-1}(\mathbf{n}'), F^k(\mathbf{n})) - P(\mathbf{n}'', F^k(\mathbf{n}); t) q(F^k(\mathbf{n}), F^{k-1}(\mathbf{n}')) \Big\} \\
&\overset{(4),(6)}{=} \sum_{k=1}^{K} \sum_{\{\mathbf{n}' \in A_1,\ \mathbf{n}' \neq \mathbf{n}\}} \Big\{ P(\mathbf{n}'', F^k(\mathbf{n}'); t) q_{A_1}(\mathbf{n}', \mathbf{n}) - P(\mathbf{n}'', F^k(\mathbf{n}); t) q_{A_1}(\mathbf{n}, \mathbf{n}') \Big\} \\
&+ \sum_{k=1}^{K} \Big\{ P(\mathbf{n}'', F^{k+1}(\mathbf{n}); t) q(F^{k+1}(\mathbf{n}), F^k(\mathbf{n})) - P(\mathbf{n}'', F^k(\mathbf{n}); t) q(F^k(\mathbf{n}), F^{k+1}(\mathbf{n})) \Big\} \\
&+ \sum_{k=1}^{K} \Big\{ P(\mathbf{n}'', F^{k-1}(\mathbf{n}); t) q(F^{k-1}(\mathbf{n}), F^k(\mathbf{n})) - P(\mathbf{n}'', F^k(\mathbf{n}); t) q(F^k(\mathbf{n}), F^{k-1}(\mathbf{n})) \Big\}
\end{aligned}$$

$$\stackrel{(*)}{=} \sum_{k=1}^{K} \sum_{\{\mathbf{n}' \in A_1,\ \mathbf{n}' \neq \mathbf{n}\}} \left\{ P(\mathbf{n}'', F^k(\mathbf{n}'); t) q_{A_1}(\mathbf{n}', \mathbf{n}) - P(\mathbf{n}'', F^k(\mathbf{n}); t) q_{A_1}(\mathbf{n}, \mathbf{n}') \right\}$$

$$\stackrel{(7)}{=} \sum_{\mathbf{n}' \in A_1} \left\{ P_{A_1}(\mathbf{n}'', \mathbf{n}'; t) q_{A_1}(\mathbf{n}', \mathbf{n}) - P_{A_1}(\mathbf{n}'', \mathbf{n}; t) q_{A_1}(\mathbf{n}, \mathbf{n}') \right\}.$$

Normalisation of P_{A_1} can easily be verified from (7). □

It is interesting to observe that under the conditions of Theorem 2.4 at each subset A_k the transition matrix converges to the same limit: from Lemma 2.3

$$\lim_{t \to \infty} P(\mathbf{n}'', F^k(\mathbf{n}); t) = \pi(\mathbf{n}), \quad \mathbf{n} \in A,\ k = 1, \ldots, K.$$

Remark 2.5 In condition (6) transitions from A_k to A_{k+1} and from A_k to A_{k-1} are allowed only. Observe that these transitions are not *required* for the statement of Theorem 2.4 to be justified. In particular, replacing (6) by

$$q(\mathbf{n}, \mathbf{n}') > 0 \Longleftrightarrow \mathbf{n}' = F(\mathbf{n}) \tag{8}$$

allows transitions from A_k to A_{k+1} but not from A_k to A_{k-1}. This shows that the result of Theorem 2.4 is not a consequence of balance between successive sets. In fact, from the proof of Theorem 2.4 it is apparent that (8) is responsible for the terms involving transitions from A_k to A_{k+1} cancelling in the equality $(*)$.

Remark 2.6 (Applicability of the result) Theorem 2.4 relates the transition matrix of $\mathbf{X}_A$ to the transition matrix of $\mathbf{X}$. Application of this theorem therefore requires the latter transition matrix to be known. A motivation for Theorem 2.4 comes from equilibrium analysis of queueing networks, where the equilibrium distribution in the unconstrained case is usually much easier to obtain than the equilibrium distribution in the constrained case. In the example below, this situation is carried over to the transition matrix: the transition matrix of the unconstrained case is of product-form as obtained from [2], which enables the evaluation of the transition matrix of the constrained case.

3 Example

This section provides an example of the result of Theorem 2.4. In this example the transition matrix at S is known to be of product-form. The example considers the case of 2 sets, A and A^c, but can be extended to multiple sets. First the product-form results are reviewed in section 3.1. The example of a transition matrix that is the sum of two product-forms is given in section 3.2.

3.1 A product-form network

Consider a closed queueing network consisting of M infinite-server queues containing N customers. A state of the queueing network is the vector $\mathbf{n} = (n_1, \ldots, n_M)$ with components n_i denoting the number of customers present at station i. The state space S of this queueing network is

$$S = \{\mathbf{n} \in \mathbb{N}_0^M \mid \sum_{i=1}^{M} n_i = N\}.$$

Let μ_i be the service rate at station i. A customer completing service at station i is routed to station j with probability p_{ij}. The transition rates are

$$q(\mathbf{n}, \mathbf{n}') = \begin{cases} n_i \mu_i p_{ij}, & \mathbf{n}' = \mathbf{n} - \mathbf{e}_i + \mathbf{e}_j, \\ 0, & \text{otherwise}, \end{cases} \tag{9}$$

where $\mathbf{e}_i$ is the i-th unit vector containing a 1 in place i, zeros elsewhere.

Assume that the initial distribution of the Markov chain modelling this queueing network is $P(\mathbf{n}''; 0) = \mathbf{1}(\mathbf{n}'' = \mathbf{e}_1)$, i.e., initially all customers reside at station 1. From [2], the transition matrix is

$$P_S(\mathbf{n}'', \mathbf{n}; t) = N! \prod_{k=1}^{M} \left(\frac{c_k(t)}{\mu_k} \right)^{n_k} \frac{1}{n_k!}, \quad \mathbf{n}'', \mathbf{n} \in S, \quad t \geq 0, \tag{10}$$

where $\{c_i(t)\}_{i=1}^{M}$ the solution of

$$\frac{1}{\mu_k} \frac{dc_k(t)}{dt} = \sum_{i=1}^{M} \{c_i(t) p_{ik} - c_k(t) p_{ki}\}, \quad k = 1, \ldots, M, \tag{11}$$

with initial conditions $c_k(0) = \mathbf{1}(k = 1)$. The product-form transition matrix (10) is determined by an M dimensional differential equation, which substantially reduces the effort required for determining this transition matrix. The product-form transition matrix will be used in the example below.

3.2 A sum of two product-forms

Consider a closed queueing network consisting of $M = 2R$ infinite-server stations which are grouped into two clusters, cluster 1 containing stations $1, \ldots, R$, and cluster 2 containing the remaining stations. Assume that each cluster has a unique input/output station from which customers can route to the other cluster. Let stations R and $R+1$ be these input/output stations. Let $s \in \mathbb{N}$, and assume that $N = 2s + 1$ customers are present in the queueing network. The assumption on the cluster structure then implies that the routing probabilities satisfy

$$\sum_{j=1}^{R} p_{ij} = 1, \quad 1 \leq i < R, \qquad \sum_{j=R+1}^{2R} p_{ij} = 1, \quad R+1 < i \leq 2R.$$

Let A be the state space for the queueing network in which the number of customers at cluster 1 is restricted not to exceed s, that is

$$A = \{\mathbf{n} \in S | \sum_{i=1}^{R} n_i \leq s\},$$

then, under the condition $N = 2s + 1$, $S = A \cup A^c$. Further, assume that the service rates and routing probabilities are such that

$$\mu_i = \mu_{2R+1-i}, \quad p_{ij} = p_{2R+1-i,2R+1-j}, \quad i, j = 1, \ldots, 2R, \tag{12}$$

which implies that cluster 2 behaves exactly like cluster 1.

Define the cyclic mapping, F, of order 2 as

$$F(\mathbf{n}) = F(n_1, \ldots, n_{2R}) = (n_{2R}, \ldots, n_1),$$

that is $F(\mathbf{n})_i = n_{2R+1-i}$. It can easily be shown that F satisfies all criteria of Definition 2.2, and Theorem 2.4. Therefore, we may now conclude that the transition matrix of the Markov chain $\{X_A(t),\ t \geq 0\}$, at state space A, with transition rates (9) satisfying (12) is given by

$$P(\mathbf{n}'', \mathbf{n}; t) = N! \prod_{k=1}^{M} \left(\frac{c_k(t)}{\mu_k}\right)^{n_k} \frac{1}{n_k!} + N! \prod_{k=1}^{M} \left(\frac{c_k(t)}{\mu_k}\right)^{F(\mathbf{n})_k} \frac{1}{F(\mathbf{n})_k!}, \quad \mathbf{n}'', \mathbf{n} \in A, \quad t \geq 0,$$

where $\{c_i(t)\}_{i=1}^{M}$ is determined from the differential equation (11).

Observe that for $R = 1$ the example reduces to the Engset loss model. The assumption (12) then implies that the service rate equals the rate at which customers arrive to the station. This model was discussed in [1], where the transition matrix was shown to be a sum of 2 product-forms.

References

[1] Boucherie, R.J. (1993) A note on the transient behaviour of the Engset loss model. *Stochastic Models* **9**, 145-156.

[2] Boucherie, R.J. and Taylor, P.G. (1993) Transient product form distributions in queueing networks. *Discrete Event Dynamic Systems: Theory and Applications* **3**, 375-396.

[3] Chung, K.L. (1967) *Markov chains with stationary transition probabilities.* 2nd ed. Springer-Verlag.

[4] Foley, R.D. (1982) The non-homogeneous $M/G/\infty$ queue. *Opsearch* **19**, 40-48.

[5] Harrison, J.H. and Lemoine, A.J. (1981) A note on networks of infinite-server queues. *Journal of Applied Probability* **18**, 561-567.

[6] Kingman, J.F.C. (1969) Markov population processes. *Journal of Applied Probability* **6**, 1-18.

[7] Massey, W.A. and Whitt, W. (1993) Networks of infinite server queues with nonstationary Poisson input, *Queueing Systems* **13**, 183-250.

Queueing systems with periodic service

M.J.A. van Eenige
National Aerospace Laboratory NLR
P.O.Box 90502, 1006 BM Amsterdam
E-mail: eenige@nlr.nl

1 Introduction

Various real-life situations are modelled in a natural way by queueing systems with periodic service. In these systems a server is offering service to customers periodically. Figure 1 gives an example of a periodic service policy. In this figure the server alternately does not offer service for 30 time units and offers service for 10 time units.

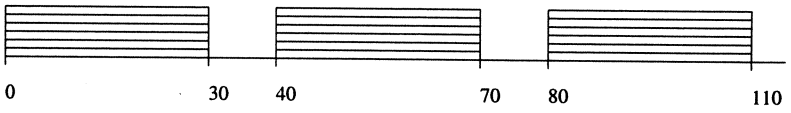


Figure 1: *A representation of a periodic service policy*

Fixed-cycle traffic lights, computer systems with periodic access schemes and periodic production rules are three examples of situations that can be modelled by systems with periodic service. More precisely, for the first example consider a traffic light at an intersection. Cars that approach this intersection from a certain direction alternately face red and green time periods of fixed duration. In a real-time computer system the capacity for executing tasks is shared by different types of tasks. Some of these tasks have strict time-critical requirements. To meet these requirements these tasks have priority and their executions are scheduled periodically. So the system's capacity is available to the ordinary tasks periodically. For the third example consider a machine at a production centre. Every four weeks, say, this machine produces a certain type of product for one week.

Queue lengths and sojourn times are important performance measures, and many other performance measures, such as the fraction of customers served in time, can be obtained from them. To evaluate queueing systems with periodic service we need techniques for determining these performance measures. In the literature both analytical and approximative approaches have been applied.

Typical analytical techniques are the generating-function technique (e.g., Darroch [4] and Rubin & Zhang [13]), the use of Laplace-Stieltjes transforms (e.g., Şahin & Bhat [14], Schassberger [15], and Ott [11]), and the matrix-geometric approach (e.g., Alfa & Neuts [2]). Unfortunately we face both analytical and numerical problems when applying these techniques. More specifically, for the generating-function technique an important and well-known problem is the determination of the solutions of a characteristic equation. Further these solutions have to be substituted into a system of regularity conditions. Since these solutions may be closely clustered, solving this nearly linearly dependent system can lead to

numerical difficulties. The use of Laplace-Stieltjes transforms requires the often difficult task of solving integral equations and of inverting these transforms for obtaining explicit results. If we want to apply the matrix-geometric approach, we have to solve a polynomial matrix equation. Solving this equation may be time consuming when the matrices are large or when the utilisation factor is close to one. Furthermore, the size of the matrices involved becomes quite large when this approach is applied to queueing systems with periodic service.

Approximations for the average waiting time or queue length have been derived by, e.g., Webster [20], Fischer [7], and Federgruen & Green [6]. However, information about averages only is often insufficient for evaluating queueing systems; other performance measures, like the fraction of customers served in time, are important too. Furthermore, some of these approximations are only valid for a rather limited class of queueing systems with periodic service. For instance, some approximations have been derived for the case that the service times are deterministic and for the case that the queue length cannot increase during periods the server is offering service.

So both analytical and approximative approaches as found in the literature may not be quite suited or may be too limited for analysing and evaluating systems with periodic service. This raises the question whether there are useful techniques for analysing and evaluating these systems; in particular whether there are techniques for determining the queue-length and sojourn-time distribution of customers. In this paper we present two techniques for determining the queue-length distribution. The results of these techniques can be used to obtain the sojourn-time distribution.

In the first technique we consider the queue length at certain time instants. This technique then exploits that, for a broad and important class of arrival-processes and service-time distributions, the tail of the stationary queue-length distribution at these instants is asymptotically geometric. This technique is a generalisation of the approach in Tijms & Van de Coevering [18].

For the second technique we derive a periodic system of equations describing the queue-length process. Each of these equations is related to Lindley's equation for the $D/G/1$ queueing system. The second technique solves this system of equations by a moment-iteration technique which is based on De Kok [10] and it uses the first two moments of the service times only.

The outline of this paper, which is based on Van Eenige [5], is as follows. In Section 2 we describe a class of queueing systems with periodic service. For convenience we consider the queueing systems in discrete time. In this way the complexity of the analysis is reduced considerably. Moreover, it enables us to use probabilistic arguments to obtain the quantities of interest. In Section 3 we show that the queue-length process for these systems reduces to the study of a Markov chain. The technique exploiting the tail behaviour of the stationary distribution of this chain is presented in Section 4. The moment-iteration method is the topic of Section 5. A summary, some extensions and the conclusions are given in Section 6. It is remarked that we confine ourselves to presenting the results without proofs; the proofs can be found in [5].

2 Model

We consider a single-server queueing system in discrete time by dividing the time-axis into intervals of equal length. Such an interval is called a slot. Service is offered periodically: there is service during on-periods and no service during off-periods. The length of the off-periods and of the on-periods are both constant. An off-period and the next on-period together are

called a cycle. So the length of a cycle is also constant. The length of an on-period and of a cycle are measured in numbers of slots, and denoted by A and C, respectively. The slots in a cycle are numbered $1, 2, \ldots, C$.

In each slot of the cycle exactly one customer is assumed to arrive with probability p, where $0 < p < 1$, and no customer arrives with probability $1 - p$. Arrivals in different slots are assumed to be independent. Hence the arrival process of customers is a Bernoulli process with parameter p.

The service times of customers are measured in numbers of slots. The probability generating function F of a service time is

$$F(z) = \sum_{i=1}^{B} b(i) \left(\frac{1-\beta}{1-\beta z} \right)^i z^i, \qquad 0 < \beta < 1,$$

where B is a positive constant and $\{b(i), i = 1, 2, \ldots, B\}$ a probability distribution. In other words, the service time of a customer consists of i service phases with probability $b(i)$ where a service phase is geometrically distributed with parameter β. We note that this class of service-time distributions consists of all finite mixtures of negative binomial distributions with the same parameter β. Mixtures of negative binomial distributions can be used to approximate Poisson mixtures arbitrarily close (cf. Steutel & Van Eenige [16]).

Customers are served in the order of their arrival. The arrival process and service times are assumed to be independent. Further, the service of a customer that is interrupted (due to an off-period) is resumed where it was interrupted.

Customer arrivals, and the start and completion of service phases occur at slot boundaries. For convenience we assume that the completion of service phases (and hence customer departures) occur just before slot boundaries, and that arrivals and the start of a service phase occur just after slot boundaries. Further, if the server is idle upon a customer arrival, he starts servicing this customer immediately.

3 Queue-length process

In this section we study the queue-length process of customers. Once the stationary queue-length distribution is known, the sojourn-time distribution can be calculated exactly. For the calculation of the sojourn-time distribution we refer the interested reader to Van Eenige [5].

To analyse the queue-length process of customers we consider the system at the first slot boundary of cycles, i.e., at the slot boundaries between two consecutive cycles. Let X_k denote the number of service phases at this imbedded time instant for cycle k with $k = 1, 2, 3, \ldots$. Then the stochastic process $\{X_k, k = 1, 2, 3, \ldots\}$ is a homogeneous discrete-time Markov chain with state space the non-negative integers and with (a possibly random) initial state X_1. The stationary transition probabilities $p_{i,j}$ of this chain only depend on i and j only through their difference $j - i$ if $i \geq A$ or $j \geq C \cdot B$ (or both) as is stated in the next lemma. Furthermore this lemma gives an explicit characterisation of these probabilities.

Lemma 1 *For $i \geq A$ or $j \geq C \cdot B$ or both, $p_{i,j} := q_{j-i}$. The shifted probability generating function $Q(z) := z^A \sum_{h=-A}^{C \cdot B + A} q_h z^h$ is given by*

$$Q(z) = z^A (\beta + (1-\beta)/z)^A (1 - p + p \sum_{i=1}^{B} b(i) z^i)^C,$$

where the terms $(\beta + (1-\beta)/z)$ *and* $(1 - p + p\sum_{i=1}^{B} b(i)z^i)$ *are related to the probability generating function of the number of service phases completed in a slot (in an on-period of course) and that of the number of service phases arriving in a slot, respectively.*

The probabilities $p_{i,j}$ that are not contained within Lemma 1, are generally hard to characterise explicitly. However, they can be determined recursively from the one-slot transition probabilities as follows. Let Y_n denote the number of service phases at the n-th slot boundary in the cycle with $n = 1, 2, \ldots, C$, S_n the number of service phases arriving in the n-th slot of the cycle, and δ_n a random variable on $\{0, 1\}$ with $\Pr\{\delta_n = 1\} = 1 - \beta$. Then the following relations are easily deduced

$$Y_{n+1} = \begin{cases} Y_n + S_n, & n = 1, 2, \ldots, C - A, \\ \max\{0, Y_n + S_n - \delta_n\}, & n = C - A + 1, C - A + 2, \ldots, C. \end{cases} \tag{1}$$

¿From these relations the remaining transition probabilities $p_{i,j}$ can be computed, since $p_{i,j} = \Pr\{Y_{C+1} = j | Y_1 = i\}$.

Under the assumption that the Markov chain is irreducible and aperiodic, and that the average number of slots work arriving per cycle is strictly less than A (i.e., the service capacity per cycle), this chain is ergodic (cf. Pakes [12]) so that it has a unique stationary distribution $\{\pi_j, j = 0, 1, 2, \ldots\}$. This stationary distribution is the unique solution to the system of equilibrium equations of the Markov chain and to the normalisation equation. Using Lemma 1 this system can be written as

$$\pi_j = \pi_0 p_{0,j} + \pi_1 p_{1,j} + \cdots + \pi_{A+j} p_{A+j,j}, \qquad j = 0, 1, \ldots, C \cdot B - 1, \tag{2}$$

$$\pi_j = \pi_{j-C\cdot B} q_{C\cdot B} + \pi_{j-C\cdot B+1} q_{C\cdot B-1} + \cdots + \pi_{j+A} q_{-A}, \qquad j \geq C \cdot B. \tag{3}$$

The normalisation equation is as usual

$$\sum_{j=0}^{\infty} \pi_j = 1. \tag{4}$$

Taking a closer look at equations (3) we notice that they constitute a $(C \cdot B + A)$-th order homogeneous linear difference equation with constant coefficients. So from the theory of difference equations (e.g., Henrici [9]) we know that there are $C \cdot B + A$ (not necessarily distinct) solutions of the form $\pi_j = z^j$ to this equation. By linearly combining these solutions we may satisfy the equations (2) and (4). However only for solutions z^j with $|z| < 1$ the coefficient in the linear combination can be non-zero, since otherwise this linear combination cannot satisfy the equation (4). By Rouché's Theorem (cf. Titchmarsh [19]) one can show that there are exactly $C \cdot B$ such (not necessarily distinct) solutions.

There are two standard techniques for solving the difference equation and its boundary equations. The first directly seeks solutions of the form $\pi_j = z^j$ and after that uses a linear combination of the solutions z^j with $|z| < 1$ to satisfy the equations (2) and (4). The other is the generating-function technique.

Application of either of these techniques shows that the stationary distribution can expressed as

$$\pi_j = \sum_{k=1}^{K} \sum_{i=1}^{m_k} \lambda_{k,i} j^{i-1} z_k^j, \qquad j = 0, 1, 2, \ldots, \tag{5}$$

where z_k with $k = 1, 2, \ldots, K$ are the K distinct solutions inside the unit circle to the characteristic equation (corresponding to the difference equation), which is

$$z^{C\cdot B} = q_{C\cdot B} + q_{C\cdot B-1} z + \cdots + q_{-A} z^{C\cdot B+A}, \tag{6}$$

where m_k denotes the multiplicity of the solution z_k (with $m_1 + m_2 + \cdots + m_K = C \cdot B$), and where the $\lambda_{k,i}$'s denote constants.

Unfortunately application of these techniques can lead to numerical difficulties. Firstly it is in general hard to determine all solutions to the characteristic equation accurately. Secondly, even if we are able to compute all these solutions accurately, some of them tend to be closely clustered so that (after substitution) the system of equations (2) and (4) becomes nearly linearly dependent. In general it is however unclear under what conditions and properties of the model these problems occur. Therefore we present two numerical approaches for determining the stationary imbedded queue-length distribution. These two approaches appear to be numerically stable.

4 Geometric-tail technique

¿From the form (5) of the stationary probabilities it directly follows that the largest solution in absolute value to the equation (6) within the unit circle determines the tail behaviour of the stationary distribution. As can be proved this solution is the unique solution to this equation in the interval $(0, 1)$. This result, the kind of which also appears in the theory of branching process (e.g., Athreya & Ney [3]), is presented in the next lemma, where d denotes the greatest common divisor of $C \cdot B$ and the powers of z having positive coefficients at the right-hand side of equation (6).

Lemma 2 *Let $z_1, z_2, \ldots, z_{C \cdot B}$ be the $C \cdot B$ not necessarily distinct solutions to equation (6) inside the unit circle. Exactly one of these solutions (z_1, say) lies in the interval $(0, 1)$. Furthermore, if $d = 1$ then $|z_k| < z_1$ for $k = 2, 3, \ldots, C \cdot B$, and if $d > 1$ then $|z_k| = z_1$ for $k = 2, 3, \ldots, d$ and $|z_k| < z_1$ for $k = d + 1, d + 2, \ldots, C \cdot B$.*

¿From the form (5) and Lemma 2 it can be shown that the tail of the stationary distribution is asymptotically

$$\lim_{j \to \infty} \frac{\pi_{j+d}}{\pi_j} = z_1^d. \tag{7}$$

So the unique positive solution to the equation (6) inside the unit circle determines the tail behaviour completely. This solution can be computed easily and accurately by, e.g., bisection. For numerical stability we suggest to take the logarithm at both sides of equation (6) first. Adapting the algorithm in Tijms & Van de Coevering [18] to the case $d > 1$ we can exploit this behaviour for numerical purposes.

By the existence of the limit (7) a straightforward approximation is

$$\pi_{j+d} = z_1^d \pi_j, \qquad j \geq J, \tag{8}$$

where J is an integer for which the quotient π_{J+d}/π_J is (fairly) good approximated by z_1^d. So it remains to compute the probabilities $\pi_0, \pi_1, \ldots, \pi_{J+d-1}$. These probabilities are the unique solution to the system of equilibrium equations for the states $j = 0, 1, \ldots, J + d - 1$ and to the normalisation equation, after substituting the approximation (8).

Clearly the computational effort of this approximation is low and the results are accurate if (8) is a good approximation for the quotient π_{J+d}/π_J for small values of J. It turns out that this approximation yields accurate results for relatively small values of J. Moreover, this way of 'cleverly' truncating the infinite system of equilibrium equations to a finite one is advantageous from a computational point of view and is less sensitive to the utilisation of the system than when using simple truncation, as can be seen in Table 1. However, the

			$\beta = 2/3$, $B = 1$, $b(1) = 1$				$\beta = 1/3$, $B = 2$, $b(2) = 1$			
ρ	C	A	J_{exp}	J_{std}	T_{exp}	T_{std}	J_{exp}	J_{std}	T_{exp}	T_{std}
0.75	60	15	20	20	50	60	30	30	70	80
	120	30	20	20	50	60	40	40	80	80
	180	45	30	30	50	60	30	40	70	90
0.90	60	15	20	20	150	170	30	30	210	230
	120	30	30	30	150	200	40	40	210	200
	180	45	30	30	140	170	40	50	210	220
0.95	60	15	20	20	300	310	30	30	370	430
	120	30	20	30	280	300	50	40	370	450
	180	45	30	30	270	310	50	40	370	420

Table 1: *The thresholds J_{exp} and J_{std} for the geometric-tail technique and the thresholds T_{exp} and T_{std} for the simple truncation in order to compute the average and the standard deviation of the number of service phases in the system in six-decimal accuracy for different values of the utilisation factor $\rho := p \cdot C \cdot B/(A(1-\beta))$.*

theoretical foundation of this approximation is still incomplete, so that an appropriate value for J has to be determined experimentally.

Finally notice that the stationary queue-length distribution at other slot boundaries in the cycle can be computed from the imbedded queue-length distribution by using the equations (1). Further, by the Bernoulli-arrivals-see-time-averages property (cf. Halfin [8]) this distribution for the n-th slot boundary in a cycle is also the distribution of the number of service phases upon an arbitrary arrival at this slot boundary. With this property we can compute the sojourn-time distribution of a customer.

5 Moment-iteration technique

The technique exploiting the geometrical tail behaviour of the stationary distribution makes a detailed use of the service-time distribution of customers. In practice however, one usually has only (approximate) knowledge about the first two moments of this distribution. In this section we present a technique for analysing the queue-length process of customers that uses only this information. More specifically, we present a technique for approximating the first two moments of the queue length and the probability of an empty queue at the start of a cycle.

The starting point for the technique is the equations (1). Let $X_{k,n}$ denote the number of slots work at the n-th slot boundary in the k-th cycle with $k = 1, 2, 3, \ldots$ and $n = 1, 2, \ldots, C$, and $\hat{S}_n$ the number of slots work arriving in the n-th slot of the cycle. Then for $k = 1, 2, 3, \ldots$ these equations can be rewritten as

$$X_{k,n+1} = \begin{cases} X_{k,n} + \hat{S}_n, & n = 1, 2, \ldots, C - A, \\ \max\{0, X_{k,n} + \hat{S}_n - 1\}, & n = C - A + 1, C - A + 2, \ldots, C, \end{cases} \tag{9}$$

where $X_{k,C+1}$ should be read as $X_{k+1,1}$. Note that

$$X_{k,C-A+1} = X_{k,1} + \sum_{n=1}^{C-A} \hat{S}_n. \tag{10}$$

Equation (9) has a similar form as Lindley's equation which describes the waiting-time process in a $D/G/1$ queueing system. For this system in continuous time De Kok [10] develops an efficient moment-iteration algorithm for approximating the stationary waiting-time distribution. He uses only the first two moments of the service-time distribution. The algorithm presented below is an adaptation of this algorithm. In this algorithm it is implicitly assumed that we have a procedure for fitting discrete distributions by matching their first two moments. After presenting the algorithm we make some remarks on the fitting procedure used.

Let $\sigma_{k,n}$ denote the standard deviation of $X_{k,n}$ and define (in distribution)

$$Y_{k,n} \stackrel{d}{=} (X_{k,n}|X_{k,n} > 0) - 1, \qquad k = 1, 2, 3, \ldots \text{ and } n = 1, 2, \ldots, C. \tag{11}$$

Then from equation (9) we have for $n = C - A + 1, C - A + 2, \ldots, C$

$$\Pr\{X_{k,n+1} > 0\} = \Pr\{X_{k,n} = 0\}\Pr\{\hat{S}_n > 1\} + \Pr\{X_{k,n} > 0\}\Pr\{Y_{k,n} + \hat{S}_n \geq 1\}. \tag{12}$$

The moment-iteration algorithm proceeds as follows.

1. Initialisation: Set $\mathrm{E}\{X_{1,1}\} = \mathrm{E}\{X_{1,1}^2\} = 0$ and $k := 1$, so $\sigma_{1,1} = 0$ and $\Pr\{X_{1,1} = 0\} = 1$.

2. Iteration: Approximate the first two moments of $X_{k,C-A+1}$ and $\Pr\{X_{k,C-A+1} > 0\}$ from (10), using the approximations for the first two moments of $X_{k,1}$ and for $\Pr\{X_{k,1} = 0\}$. For $n = C - A + 1, C - A + 2, \ldots, C$

   (a) compute the approximations for the first two moments of $Y_{k,n}$ using (11); fit a discrete distribution to $Y_{k,n}$ by matching these moments in order to approximate $\Pr\{Y_{k,n} = 0\}$;

   (b) compute the approximations for the first two moments of $X_{k,n+1}$ using (9) and the approximation for $\Pr\{X_{k,n+1} > 0\}$ from (12).

3. Stopping criterion: Compute the approximation for $\sigma_{k,C+1}$. If the approximations for the differences $|\mathrm{E}\{X_{k,C+1}\} - \mathrm{E}\{X_{k-1,C+1}\}|$ and $|\sigma_{k,C+1} - \sigma_{k-1,C+1}|$ are both small enough then execute Step 4. Otherwise set $k := k + 1$ and $X_{k,1} := X_{k-1,C+1}$, and execute Step 2.

4. Approximation: The first two moments of the stationary imbedded queue-length distribution are approximated by $\mathrm{E}\{X_{k,C+1}\}$ and $\mathrm{E}\{X_{k,C+1}^2\}$, and the stationary probability that the queue is not empty by $\Pr\{X_{k,C+1} > 0\}$.

In this algorithm we need a procedure for fitting discrete distributions on the first two moments. Adan, Van Eenige & Resing [1] present such a (novel) procedure. Firstly, however, they answer the question what combinations of mean and coefficient of variation are possible for discrete distributions concentrated on the non-negative integers. (Notice that all the random variables involved in the moment-iteration algorithm assume values in the set $\{0, 1, 2, \ldots\}$.) Their procedure for fitting discrete distributions is based on the analogue to the procedure for continuous distributions in Tijms [17]. Tijms uses hyperexponential and mixtures of two Erlang distributions as distributions for fitting. However, the discrete analogues to these distributions do not suffice to cover all the combinations of mean and coefficient of variation possible for discrete distributions on the non-negative integers. To

cover these combinations Adan, Van Eenige & Resing use four distributions: a Poisson distribution, and mixtures of two binomial distributions, of two negative binomial distributions and of two geometric distributions. For details we refer to their paper [1].

In Table 2 we display for several examples the 'exact' values (Ex) and the approximative values (MI) from the moment-iteration technique for the probability that the queue is not empty ($\Pr\{X > 0\}$) at the start of a cycle and for the average number of service phases ($\mathrm{E}\{X\}$) at the start of a cycle. The 'exact' values are those computed when truncating the states $j > T$ with T sufficiently large. This table shows that application of the fitting procedure mentioned above in the moment-iteration algorithm gives very good approximations. Other results, for which we refer to Van Eenige [5], indicate that this procedure yields also very good approximations for the tail probabilities of the sojourn-time distribution. Finally we remark that our numerical examples indicate that the moment-iteration algorithm always terminates. However, we have not been able to prove this yet.

			$\beta = 2/3$, $B = 1$ and $b(1) = 1$				$\beta = 1/3$, $B = 2$ and $b(2) = 1$			
			$\Pr\{X > 0\}$		$\mathrm{E}\{X\}$		$\Pr\{X > 0\}$		$\mathrm{E}\{X\}$	
ρ	C	A	Ex	MI	Ex	MI	Ex	MI	Ex	MI
0.75	60	15	0.478	0.485	4.87	4.84	0.429	0.431	2.87	2.90
	120	30	0.387	0.393	3.85	3.97	0.517	0.512	2.14	2.30
	180	45	0.331	0.340	3.18	3.40	0.277	0.288	1.66	1.87
0.90	60	15	0.755	0.774	18.49	18.28	0.725	0.741	12.00	11.88
	120	30	0.694	0.711	16.95	16.77	0.653	0.669	10.76	10.77
	180	45	0.650	0.666	15.82	15.80	0.604	0.625	9.86	9.99
0.95	60	15	0.871	0.889	42.02	41.67	0.854	0.876	28.03	27.66
	120	30	0.835	0.859	40.29	39.58	0.811	0.835	26.60	26.11
	180	45	0.808	0.833	38.97	38.22	0.779	0.804	25.52	25.10

Table 2: *The 'exact' and approximative values for the probability of a non-empty queue and the average number of service phases at the start of a cycle for different values of the utilisation factor* $\rho := p \cdot C \cdot B/(A(1-\beta))$.

6 Conclusions

In this paper we presented two numerical techniques for analysing and evaluating the queue-length process of customers in discrete-time queueing systems with periodic service. In contrast with analytical approaches, these techniques appear to be numerically stable. Moreover, they yield accurate approximations for the performance measures of interest. Further, these techniques are applicable to generalisations of the systems considered here. For instance, they allow for multiple on- and off-periods in a cycle, and they can deal with possibilities occurring naturally at production centres: the possibility of producing to stock and of working overtime. Moreover, they can deal with periodically time-dependent Bernoulli arrival processes and with service-time distributions that depend on the slot of arrival. Finally it is remarked that these techniques are also applicable to other queueing systems, since they only exploit the explicit structure of the equilibrium equations (as in the geometric-tail technique) or the relation between the quantities of interest at certain imbedded time instants (as in

the moment-iteration method). Hence we expect that they can also be applied successfully to other systems.

References

[1] I. Adan, M. van Eenige, and J. Resing (1995), Fitting discrete distributions on the first two moments, *Probability in the Engineering and Informational Sciences* **9**, pp. 623-632.

[2] A.S. Alfa and M.F. Neuts (1995), Modelling vehicular traffic using the discrete time Markovian arrival process, *Transportation Science* **29**, pp. 109-117.

[3] K.B. Athreya and P.E. Ney (1972), *Branching Processes*, Springer-Verlag, Berlin.

[4] J.N. Darroch (1964), On the traffic-light queue, *Annals of Mathematical Statistics* **35**, pp. 380-388.

[5] M.J.A. van Eenige (1996), *Queueing Systems with Periodic Service*, Ph.D. thesis, Eindhoven University of Technology.

[6] A. Federgruen and L. Green (1986), Queueing systems with service interruptions, *Operations Research* **34**, pp. 752-768.

[7] M.J. Fischer (1977), Analysis and design of loop service systems via a diffusion approximation, *Operations Research* **25**, pp. 269-278.

[8] S. Halfin (1983), Batch delays versus customer delays, *The Bell System Technical Journal* **62**, pp. 2011-2015.

[9] P. Henrici (1968), *Discrete Variable Methods in Ordinary Differential Equations*, John Wiley & Sons, New York.

[10] A.G. de Kok (1989), A moment-iteration method for approximating the waiting-time characteristics of the $GI/G/1$ queue, *Probability in the Engineering and Informational Sciences* **3**, pp. 273-287.

[11] T.J. Ott (1984), On the $M/G/1$ queue with additional inputs, *Journal of Applied Probability* **21**, pp. 129-142.

[12] A.G. Pakes (1969), Some conditions for ergodicity and recurrence of Markov chains, *Operations Research* **17**, pp. 1058-1061.

[13] I. Rubin and Z. Zhang (1988), Message delay analysis for TDMA schemes using contiguous-slot assignments, *Proceedings of the IEEE International Conference on Communications 1988*, Philadelphia, pp. 418-422.

[14] I. Şahin and U.N. Bhat (1971), A stochastic system with scheduled secondary inputs, *Operations Research* **19**, pp. 436-446.

[15] R. Schassberger (1974), A broad analysis of single server priority queues with two independent input streams, one of them Poisson, *Advances in Applied Probability* **6**, pp. 666-688.

[16] F.W. Steutel and M.J.A. van Eenige (1996), Note on the approximation of distributions on $\mathbb{Z}_+$ by mixtures of negative binomial distributions, Memorandum COSOR 96-13, Department of Mathematics and Computing Science, Eindhoven University of Technology, (to appear in *Stochastic Models* **13**, (1997)).

[17] H.C. Tijms (1986), *Stochastic Modelling and Analysis: A Computational Approach*, John Wiley & Sons, New York.

[18] H.C. Tijms and M.C.T. van de Coevering (1991), A simple numerical approach for infinite-state Markov chains, *Probability in the Engineering and Informational Sciences* **5**, pp. 285-295.

[19] E.C. Titchmarsh (1960), *The Theory of Functions* (second ed.), Oxford University Press, London.

[20] F.V. Webster (1958), *Traffic signal settings*, Road Research Technical Paper No. 39, Road Research Laboratory, Department of Scientific and Industrial Research, Her Majesty's Stationery Office, London.

A general approach to computing loss probabilities in finite-buffer queues

Frank N. Gouweleeuw
McKinsey
Amsterdam
E-mail: frank_gouweleeuw@mckinsey.com

Abstract

This paper discusses a general method for computing loss probabilities in finite-buffer queues. The method is based on a relation between the steady-state probabilities of a finite-buffer queue and the corresponding infinite-buffer queue. It is exact for several queues of the $M/G/1$-type, and serves as an excellent approximation for many other queueing systems.

1 Introduction

In many practical situations, such as manufacturing and telecommunication systems, we encounter queueing systems with a finite buffer. A finite-buffer queue is in general more difficult to analyze than the corresponding infinite-buffer queue. In principle, any practical finite-buffer queueing system can be modelled as a Markov process by incorporating sufficient information in the state space description. When the size of the state space is small, it is relatively easy to obtain stationary (and also transient) solutions quickly and accurately by means of numerical methods, such as the successive overrelaxation method or iterative aggregation/disaggregation methods. In many queueing systems, however, the dimension and size of the state space will grow quickly beyond any practical bound. The process of obtaining stationary solutions in these cases becomes much more difficult and time consuming. This makes it desirable to use approximation methods that are more efficient to implement, but still give sufficiently accurate results.

This paper will focus primarily on a generally applicable approximation method for the loss probability of a customer in queueing models with general input. This method is based on the relation between a finite-buffer queue and its corresponding infinite-buffer queue. For a class of $M/G/1$ type queues it is known that the state probabilities of the finite-buffer and the corresponding infinite-buffer queues are proportional to each other for a limited set of states. This proportionality can be explored to derive an exact formula for the loss probability of a customer in various finite-buffer $M/G/1$-type queues. First we will discuss and extend the class of queueing systems for which an exact proportionality relation holds between the state probabilities of the finite- and the infinite-buffer queue. Second, we will show that such a proportionality can serve as an approximating assumption in queueing systems in which it does not hold exactly. From this assumption an excellent approximation for the loss probability of a customer can be derived.

Despite the simplicity of the proportionality relation, its practical application is hampered by the fact that the state probabilities of the infinite-buffer queue have to be calculated. Although the analysis of an infinite-buffer queue is in general less demanding than the analysis of a finite-buffer queue, it is not always possible to obtain stationary solutions for infinite-buffer queues. Even the single-server $GI/G/1$ queue and the multi-server $M/G/c$ queue with Poisson arrivals permit no simple analytical solution. There are, however, a number of practical queueing models with infinite buffer capacity in which we can obtain the generating function of the state probabilities. These generating functions can be inverted numerically by means of a discrete version of the fast Fourier transform (FFT) method.

2 The batch arrival $GI/G/c$ queue

Consider the single-arrival $GI/G/c/K+c$ queue, where the interarrival times of customers have a general distribution $F(x)$ with expectation $\mathsf{EA} = \lambda^{-1}$. There are c servers to handle service requests and the service time of a customer has a general distribution $G(x)$ with expectation $\mathsf{ES} = \mu^{-1}$. The system load ρ is defined as $\rho = \lambda/\mu c$ and we assume that $\rho < 1$. There is a buffer of size K to store incoming customers that find all c servers busy and those customers that find the system completely full are rejected and do not influence the system. Let π_j $(j = 0, \ldots, K+c)$ be the long run fraction of customers finding upon arrival j other customers present in the system. Denote by P_{loss} the long run fraction of customers that are lost. Since we consider a single-arrival queue, this loss probability can be calculated by $P_{loss} = \pi_{K+c}$. The probability π_{K+c} can be calculated for various queueing models using the embedded Markov chain approach, but this can be computer intensive when done for several large buffer sizes. Since it is often less involving to solve the corresponding infinite-buffer queue, various approximations for the loss probability of a customer have been developed, based on the steady state probabilities of the infinite-buffer queue. Define $\pi_j^{(\infty)}$ $(j = 0, 1, \ldots)$ as the long run fraction of customers finding upon arrival j other customers present in the corresponding infinite-buffer $GI/G/c$ queue. These probabilities are well defined under the assumption $\rho < 1$.

A well known approximation for the loss probability of a customer is

$$P_{loss}^{appl} = \sum_{j=K+c}^{\infty} \pi_j^{(\infty)}, \tag{1}$$

see e.g. Kleinrock [2]. The right-hand side of this equation can be interpreted as the probability that K+c or more customers are found upon arrival in the infinite-buffer queue. Therefore, this approximation is often referred to as the tail approximation. No theoretical and hardly any empirical evidence is presented in the literature to support this approximation. We will show later that this approximation is not even asymptotically exact and it can differ a factor $(1-\rho)^{-1}$ from the exact value of the loss probability as K gets large. In fact, in Daigle [3] it was shown numerically that this factor $(1-\rho)^{-1}$ can already be obtained for moderate values of the buffer size K. Therefore, it is surprising that even in recent literature (e.g. Bruneel [4, pp. 148, 152]) the above equation is used to approximate the loss probability of a customer.

A more refined approximation was presented in Tijms [5] and extended in Gouweleeuw [1]. The crucial assumption underlying this approximation is that the probabilities π_j are proportional to $\pi_j^{(\infty)}$ for $j < K+c$ and not for $j = K+c$, that is

$$\pi_j = \gamma \pi_j^{(\infty)}, \qquad j = 0, \ldots\ldots, K+c-1, \tag{2}$$

for some constant $\gamma > 0$. This proportionality relation is exact for the $M/G/1$ queue, as was first shown by Keilson [6] and subsequently, in a more direct way by Cooper [7]. In Tijms [5] an alternative proof of this equation is presented, showing that this equation is exact for both the $M/G/1$ and the $M/M/c$ queue. This proof uses basic results from the theory of regenerative processes together with the fact that the arrival process is memoryless. Keilson and Servi [8] present a more general theorem, showing under which conditions the state probabilities of two general multivariate Markov chains are proportional. They use this elegant theorem to prove that the above relation is exact for the $M/G/1$ queue with server vacation. Given this relation, using simple renewal theoretic results, Tijms [5] derived the following approximation for the loss probability of a customer:

$$P_{loss}^{app3} = \frac{(1-\rho)\sum_{j=K+c}^{\infty} \pi_j^{(\infty)}}{1-\rho\sum_{j=K+c}^{\infty} \pi_j^{(\infty)}}. \tag{3}$$

This approximation is exact for the $M/G/1/K+1$ and the $M/M/c/K+c$ queues.

The above discussion dealt with the case of partial rejection of batches. In a recent publication by Gouweleeuw [1] it was shown that the analysis for the partial rejection case can easily be extended to the batch arrival queue under the complete rejection strategy. Under this strategy an arriving batch whose size exceeds the number of unoccupied places in the system is completely rather than partially rejected. The loss probability for the case of complete rejection differs significantly from the loss probability for the case of partial rejection. It was proven that the approximation is exact for the batch-arrival $M^X/G/1/K+1$ and the $M^X/M/c/K+c$ queues, provided that the batch size has a constant value. For an extensive numerical investigation of the approximation for those queueing systems for which it is not exact, we refer to Gouweleeuw [1].

3 Discrete time queues

The previous section discussed various continuous-time queueing models. However, many practical queueing systems such as fixed-cycle traffic lights and ATM-systems in telecommunication networks, operate on a discrete-time basis. A discrete-time queueing system is characterized by time-slotted and synchronous service. The time axis is divided into intervals (called time slots) of equal length. Without loss of generality we normalize the length of a slot to unit time. We shall refer to the entities which are served as packets. Packets are allowed to arrive at any arbitrary time in a slot. Since a new service can only start at the beginning of a slot, packets which arrive in a certain slot are eligible for service from the beginning of the next slot. The service times are positive multiples of time slots. The packets completing service in a certain slot (say slot n) are considered to be leaving the system at the end of that slot. This means that such a packet will leave behind those packets that arrived in slot n as well as those packets already present at the beginning of slot n.

A classical example of a discrete-time queueing system is a stream of vehicles approaching an intersection controlled by fixed-cycle traffic lights. It takes a unit of time (a slot) for a vehicle to cross the traffic lights. The green and red periods of the traffic lights are assumed to be constant. A more recent example of a discrete-time queueing system is an Asynchronous Transfer Mode (ATM) system in telecommunication networks. In an ATM network, data from various sources is segmented into fixed-size ATM cells and statistically multiplexed

to high-speed transmission lines and switches. An extensive survey of discrete-time models used in telecommunication systems can be found in Bruneel and Kim [4].

The approximating method that was presented in the previous section can easily be extended to the class of discrete-time queues. Under a mild asumption on the arrival process, it has been proved that the approximation leads to an exact expression for the discrete-time $GI/G/1$ queue. The approximation is asymptotically exact when the buffer size gets large. The derivation of the approximation is similar to that of the continuous-time queue. Although the derivation of the approximation is given for renewal input and relies heavily on renewal-theoretic arguments, the approximation turns out to be applicable as well for more complicated arrival processes.

4 Conclusion

In this paper we have discussed a heuristic for the loss probability of a customer in various queueing systems with finite capacity. This heuristic as a simple structure an is a clear improvement over the well known tail approximation. The practical application of the method is, however, hampered by the fact that the steady state probabilities of the corresponding infinite-buffer queue have to be computed.

5 References

[1] Gouweleeuw, F.N. (1996), "A general approach to computing loss probabilities in finite-buffer queues", Ph.D. thesis, Vrije Universiteit, Amsterdam.

[2] Kleinrock, L. (1975), Queueing Systems, Voll: Theory, John Wiley, New York.

[3] Daigle, J.N. (1992), Queueing Theory for Telecommunications, Addison Wesley, Reading.

[4] Bruneel, H. and Kim, B.G. (1993), Discrete-time Models for Communication Systems including ATM, Kluwer Academic Press Boston.

[5] Tijms, H.C. (1986), Stochastic Models and Analysis: A Computational Approach, John Wiley, London.

[6] Keilson, J. (1966), "The ergodic queue length distribution for queueing systems with finite .capacity", *Journal of the Royal Statistical Society, Series B*, **28**, pp. 190-201

[7] Cooper, R.B. (1981), Introduction to Queueing Theory, Edward Arnold, London.

[8] Keilson, J. and Servi, L.D. (1989), "Blocking probability for M/G/1 vacation systems with occupancy level dependent schedules", *Operations Research*, **37**, pp. 134-140.

Analysis of a customer assignment model with decentralized control

Arie Hordijk and Anneke Loeve*
Department of Mathematics and Computer Science
University of Leiden
E-mail: hordijk@rulwi.leidenuniv.nl

Abstract

In this paper we analyse the assignment of customers in a queueing system consisting of a tandem of parallel queues. We will present three algorithms, to find good routing policies. These algorithms are generally applicable to Markov decision chains with partial information.

1. Introduction

In this paper we analyse the assignment of customers in a queueing system consisting of a tandem of parallel queues. Customers arrive to the first centre of the system and have to be routed to one of two identical exponential servers, each with his own waiting queue. When the customer is served, he arrives in the second centre where again he has to be routed to one of two identical exponential servers.
The routing of the customers in a centre may only depend on the numbers of customers in the queues of that centre, and not on the numbers of customers in the queues of the other centre. Because routing decisions have to be found in both centres, we will call this a model with decentralized control.

We want to find deterministic routing policies, maximizing the discounted throughput of customers. From results in [1], we know the optimal routing policy in the second centre. By choosing this routing policy fixed, we can use algorithms for models with partial information to find good routing policies for customers arriving in the first centre.

The queueing system is modelled as a Markov decision chain with partial information.
We will present three algorithms, which are generally applicable to Markov decision chains with partial information.

The outline of the paper is as follows.
In Section 2, we introduce the queueing model. In Section 3, we describe the algorithms. In Section 4, the Markov decision chain and the cost function are given and in Section 5, numerical results with the different algorithms are presented.

* The research of this author has been supported by the Netherlands Organization for Scientific Research (N.W.O.).

2. Queueing model

Consider the queueing network of Figure 1.

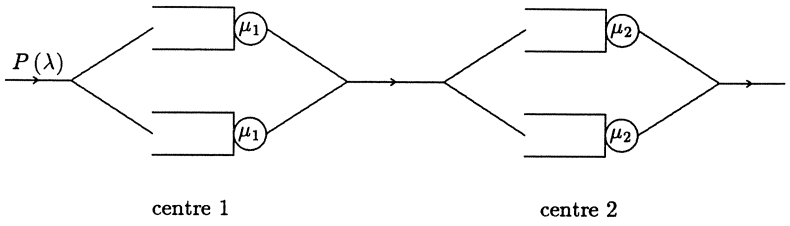


Figure 1. Tandem of parallel queues

The queueing network consists of two centres. In each centre there are two identical exponential servers, each with his own waiting queue. The service rates in the first centre are equal to μ_1 and in the second centre equal to μ_2. Customers arrive to the first centre according to a Poisson process with rate λ and they have to be routed to one of the servers. When they are served, they go to the second centre, where again they have to be assigned to one of the servers. Customers served in the second centre leave the system.
We assume all interarrival times and service times to be mutually independent. Furthermore, in each queue the customers are served on a *First In First Out* (FIFO) basis. Also, there is an initial distribution over the state space given.

The state of the system is denoted by the 4-tuple (i_1, i_2, i_3, i_4) where i_j denotes the number of customers in the j-th queue in centre 1 (including customers in service) if $j = 1$ or 2, and i_j denotes the queue length of queue $j-2$ in centre 2 if $j = 3$ or 4.
For centre i, there exists a capacity N_i $(i = 1, 2)$, such that no more customers are accepted when there are N_i customers present in the centre. Customers, arriving to a full first centre are blocked and leave the system. Customers, arriving in a full second centre, return to the server they just left and receive another service in the first centre. Because of these buffers, it is impossible that the number of customers grows to infinity. However, we only use those buffers to be able to compute performance measures for the system, and we make those buffers as large as possible to avoid effects caused by their finiteness. In this model, we only consider values for λ in $[0, \min\{2\mu_1, 2\mu_2\})$ for fixed μ_1 and μ_2.

We are interested in deterministic routing policies for customers in both centres. The routing in a centre may depend only on the numbers of customers in the queues of that particular centre.

It is shown in [1] that an optimal policy for the Markov decision chain with complete information will use the Shortest Queue Policy (SQP) in centre 2. This policy assigns customers in centre 2 to the shortest queue and in case of equal queues it routes the customers with probability $\frac{1}{2}$ to queue 1 and probability $\frac{1}{2}$ to queue 2. In fact, the routing in states with equal queue lengths may be chosen arbitrarily. It also follows from the results in [1] that the SQP is the optimal policy for centre 2 in case of decentralized control. Hence, we fix the policy in centre 2 to the SQP and we apply an algorithm for models with partial

information to obtain a good routing policy in centre 1.
The policy should maximize the discounted throughput of customers given the initial distribution.

Note that, although the numbers of customers in the second centre may not be used, implicitly there is some information used about this centre, namely the knowledge of the routing policy there. Also the initial distribution over the state space as a whole is known, which gives us information about the second centre too.

3. Algorithms

In this section we will present three algorithms to compute good policies for Markov decision chains with no state information. First, we will describe the notations used for Markov decision chains with partial information.

Consider a Markov decision chain with state space $E = \{1, 2, \ldots, N\}$. The state space is partitioned into sets E_s, $s = 1, 2, \ldots, K$, such that at each decision moment the only information available about the system is the set of the partition in which the state of the system is contained (cf. [2], [4]). Thus, an admissible decision rule prescribes the same decision for all states in a partition set E_s for all s. A Markov policy is called admissible if its decision rule at any time point is admissible.
We assume that the action set in each state is the same, namely, $A = \{1, 2, \ldots, M\}$.

The transition probabilities when in state i action a is chosen are denoted by p_{iaj} ($j = 1, 2, \ldots, N$) and the expected one step cost by c_{ia}. For the transition matrix and the vector of immediate costs when decision rule π is chosen, we will write $P(\pi)$ and $c(\pi)$ respectively. Hence, $P(\pi)_{ij} = \sum_{a=1}^{M} \pi_{ia} p_{iaj}$ and $c(\pi)_{ia} = \sum_{a=1}^{M} \pi_{ia} c_{ia}$. A deterministic decision rule is often denoted by f.

For the vector of expected discounted costs under a stationary policy $R = (\pi, \pi, \pi, \ldots) = \pi^{\infty}$, we will use the notation $v^{\alpha}(\pi)$, when the discount factor $\alpha \in [0, 1)$ is used. It holds that (cf. [6]) $v^{\alpha}(\pi)$ is the unique solution of

$$v^{\alpha}(\pi) = c(\pi) + \alpha P(\pi) v^{\alpha}(\pi). \tag{1}$$

Equivalently,

$$v^{\alpha}(\pi) = (I - \alpha P(\pi))^{-1} c(\pi). \tag{2}$$

The initial distribution is given by an N-dimensional vector β, where $\mathbb{P}(\text{initial state is } i) = \beta_i$. To define the optimal policy, we use the expected total α-discounted cost under a policy given the initial distribution. For a stationary policy π^{∞} these costs can be computed as the inner product of β and $v^{\alpha}(\pi)$, denoted as $\beta^T v^{\alpha}(\pi)$.

The following algorithm is introduced in [2],

Algorithm 1.

Choose an initial admissible deterministic decision rule f^0 and $\epsilon > 0$.

Let $x^1 = \beta^T$ *and choose* $v^1 \in \mathbb{R}^N$.

Define for $n = 1, 2, 3, \ldots$:

$$t^n(s,a) = \sum_{i \in E_s} x_i^n \left\{ c_{ia} + \alpha \sum_{j=1}^{N} p_{iaj} v_j^n \right\},$$
for $s = 1, 2, \ldots, K$, $a = 1, 2, \ldots, M$.

For $i \in E_s$, *if* $f^{n-1}(i) \in \operatorname{argmin}_{a \in A} t^n(s,a)$ *then*
$f^n(i) = f^{n-1}(i) \, \forall i \in E_s$,
else $f^n(i) = a$ *for some* $a \in \operatorname{argmin}_{a \in A} t^n(s,a) \ \forall i \in E_s$;

$$x^{n+1} = \beta^T + \alpha x^n P(f^n);$$

$$v^{n+1} = c(f^n) + \alpha P\left(f^{n+1}\right) v^n;$$

Stop if $\|x^{n+1} - x^n\| < \epsilon$ *and* $\|v^{n+1} - v^n\| < \epsilon$.

Here, $\|x\| < \epsilon$ means that $|x_i| < \epsilon$ for all i.
If the algorithm stops, then policy $(f^n)^\infty$ is chosen.

The second algorithm is a policy improvement algorithm, based on Algorithm 1 and on an algorithm of Kulkarni and Serin (cf. [4]).

Algorithm 2.

1. *Choose an initial admissible deterministic decision rule* f^0 *and* $\epsilon > 0$;
   $m = 1$.

2. *Let* $x^1 = \beta^T$ *and choose* $v^1 \in \mathbb{R}^N$.
   Define for $n = 1, 2, 3, \ldots$:

   $$x^{n+1} = \beta^T + \alpha x^n P(f^m);$$

   $$v^{n+1} = c(f^m) + \alpha P(f^m) v^n;$$

   Stop if $\|x^{n+1} - x^n\| < \epsilon$ *and* $\|v^{n+1} - v^n\| < \epsilon$.
   Let $v^\alpha(f^m) = v^{n+1}$ *and* $x^\alpha(f^m) = x^{n+1}$.

3. *Compute the quantities* $t^m(s,a)$ *for* $s = 1, 2, \ldots, K$ *and* $a = 1, 2, \ldots, M$ *defined by*

   $$t^m(s,a) = \sum_{i \in E_s} x_{ia}(f^m) \left\{ c_{ia} + \alpha \sum_{j=1}^{N} p_{iaj} v_j^\alpha(f^m) - v_i^\alpha(f^m) \right\}.$$

   Define for $s = 1, 2, \ldots, K$: $A^m(s) = \{a \in A |\ t^m(s,a) < 0\}$.
   If $A^m(s) = \emptyset$ *for all* s, *then STOP;*
   else

Choose an s^* for which $A^m(s^*)$ is nonempty and choose an action $a^* \in A^m(s^*)$. Define f^{m+1} by

$$f^{m+1}(i) = \begin{cases} a^* & \text{if } i \in E_{s^*}; \\ f^m(i) & \text{otherwise.} \end{cases}$$

4. *$m = m + 1$ and go to step 2.*

The third algorithm, which uses branch and bound techniques, is a variant of an algorithm developed by Schneeberger [7].
In the algorithm a decision tree is constructed. Each node corresponds to a set of stationary deterministic policies with fixed actions in certain subsets of the partition, in combination with a lower bound on the expected costs of the policies in that set. To describe a node, a K-dimensional vector d is used together with a lower bound L_d, where $d_s \in A \cup \{0\} = \{0, 1, \ldots, M\}$ for all s. The component d_s prescribes the action to be taken in all the states of the s-th set of the partition, E_s. When $d_s = 0$, it means there is not yet decided about the action in E_s. Now the set of possible policies in a node corresponding to a vector d is $\{R = f^\infty \mid f(i) = d_s \text{ if } d_s \neq 0 \text{ and } i \in E_s\}$. Starting with one node, where $d = 0$, the set of policies is stepwise reduced by choosing a set E_s for which $d_s = 0$, and defining M new nodes with $d_s = 1$, $d_s = 2$, ..., $d_s = M$, respectively. When this procedure is repeated often enough, nodes with $d_s \neq 0$ for all s are created, corresponding to precisely one stationary deterministic policy.

For each node, a lower bound of the optimal costs of the policies in that node, can be computed. This can be done using policy iteration, while keeping fixed the actions in the sets E_s where $d_s \neq 0$. The corresponding algorithm goes in the following way.

Algorithm to compute a lower bound.

1. *Choose a deterministic admissible decision rule f^1 for which $f^1(i) = d_s$ if $d_s \neq 0$ and $i \in E_s$;*
   $m = 1$.

2. *$v^1 := v^m$;*
   Define for $n = 1, 2, 3, \ldots$:

   $$v^{n+1} = c(f^m) + \alpha P(f^m) v^n;$$

   Stop if $\|v^{n+1} - v^n\| < \epsilon$.
   Let $v^m = v^{n+1}$.

3. *Find f^{m+1}, such that*

   $$f^{m+1}(i) \in \text{argmin}_{a \in A} \left\{ c_{ia} + \alpha \sum_{j=1}^{N} p_{iaj} v_j^m \right\} \text{ for all } i \in E_s \text{ where } d_s = 0.$$

4. *If $f^{m+1} = f^m$ then $\beta^T v^m$ is a lower bound of the costs for the policies corresponding to d;*
   otherwise $m = m + 1$ and go to step 2.

By using an upper bound of the optimal costs, nodes can be excluded. The last value vector found in the computation of a lower bound, say v^*, can be used to find an upper bound in the following way. For each E_s, where no action is chosen yet, define for all $i \in E_s$

$$f(i) = \text{argmin}_{a \in A} \left\{ \sum_{i \in E_S} \left(c_{ia} + \alpha \sum_{j=1}^{N} p_{iaj} v_j^* \right) \right\}. \tag{3}$$

Then compute for the resulting admissible policy f the discounted cost vector $v^\alpha(f)$ using step 2 of the 'Algorithm to compute a lower bound'. Then, an upper bound of the optimal costs is given by $\beta^T v^\alpha(f)$.

Now, N_d denotes the node corresponding to d, consisting of the pair (d, L_d). The global variable U will denote at each moment an upper bound on the optimal costs and the policy $(f^0)^\infty$ denotes the admissible policy corresponding to the upper bound at that moment. The algorithm is then as follows.

Algorithm 3.

1. *Compute with policy iteration the optimal policy and the corresponding discounted cost vector v^* for the model with full state information and define $L_0 = \beta^T v^*$ (use the 'Algorithm to compute a lowerbound' with $d = 0$).*
   Define an admissible policy f^0 as in formula (3) using v^* found by computing the lower bound L_0.
   Compute the discounted cost vector v^0 corresponding to f^0 using step 2 of the 'Algorithm to compute a lower bound', and let U, the upper bound, be defined as $\beta^T v^0$.
   $N_0 = \{0, L_0\}$;
   Nodeset $= \{N_0\}$.

2. *Choose $N_d \in$ Nodeset with minimal L_d.*
   If $L_d \geq U$, go to step 4,
   otherwise :

   Nodeset = Nodeset $- N_d$;
   Choose $s \in \{1, 2, \ldots, K\}$ such that $d_s = 0$.
   For $a = 1, 2, \ldots, M$:

   $d_s = a$;
   Compute the lower bound L_d.
   If $L_d \leq U$:

   compute a new upper bound U^* using the discounted cost vector corresponding to L_d and compute the admissible policy f^* corresponding to U^*.
   If $U^* \leq U$ then $f^0 = f^*$ and $U = U^*$.
   Nodeset = Nodeset $\cup \{N_d\}$.

3. *Go to step 2.*

4. *Stop : policy f^0 is optimal in the class of deterministic stationary admissible policies with discounted costs U.*

Because of the huge cardinality of the state space for relatively small values of N_1 and N_2, it is not possible to store transition matrices. Therefore, we compute the non-zero entries in these matrices again in each iteration.

4. Markov decision model

To apply the algorithms, we have to describe the Markov decision chain corresponding to the model of Section 2 (cf. [5]). The state space is the set

$$E = \left\{ (i_1, i_2, i_3, i_4) \in \mathbb{N}^4 \;\middle|\; \begin{array}{l} i_1 + i_2 \leq N_1 \\ i_3 + i_4 \leq N_2 \end{array} \right\}.$$

Because we are interested in routing policies for customers in the first centre, only depending on the numbers of customers in the queues of the first centre, we use the following partition of the state space: E is partitioned into sets $E_{(i_1,i_2)}$ such that

$$E_{(i_1,i_2)} = \{(i_1, i_2, i_3, i_4) \in \mathbb{N}^4 \mid i_3 + i_4 \leq N_2\}, \text{when } i_1, i_2 \in \mathbb{N} \text{ and } i_1 + i_2 \leq N_1.$$

The total number of states in this model is

$$\sum_{i_1=0}^{N_1} \sum_{i_2=0}^{N_1-i_1} \sum_{i_3=0}^{N_2} \sum_{i_4=0}^{N_2-i_3} 1 = \frac{1}{4}(N_1+1)(N_1+2)(N_2+1)(N_2+2).$$

The action space is $A = \{1, 2\}$. Action i means that if a customer arrives in centre 1 and is accepted, he is routed to queue i. In states where the customer is blocked, the actions are dummy actions.

First, we consider a cost function in which each customer in the system has a holding cost of one per time unit. However, when we applied the algorithms, we found that this cost function causes a lot of side effects because of the finite buffers. In states with a rather full unbalanced first centre, the policies found by the algorithms route customers to the longest queue in the first centre. In this way the probability of blocking future customers is increased (because of the joint buffer for the two queues). Hence, the total expected holding costs are decreased.
We tried to make the buffers so large that no side effects are caused by their finiteness. However, in this model, the state space is very large and therefore, the buffers have to be relatively small and blocking effects are caused. These effects can be avoided by giving a reward for each customer leaving the system. Because we study models with cost functions instead of rewards, we define costs as negative rewards. Then, the algorithms maximize the discounted number of departures.

Note that in an infinite system (system with no restricting buffers), minimizing the discounted holding cost in the system is equivalent to maximizing the discounted number of departures. Namely, when the cost rate is 1 per customer per time unit, the holding cost in the system at time t is equal to to the number of customers present at time t.
These costs have to be discounted by the discount factor at time t, α^t, which can be denoted by e^{-bt} with $b = -\ln(\alpha)$. Indeed, if we denote the number of arrivals up to time t

by $A(t)$ and the number of departures up to time t by $D(t)$, then minimizing the expected discounted holding cost is equivalent to minimizing

$$\mathbb{E}\int_0^\infty e^{-bt}(A(t)-D(t))\ dt. \tag{4}$$

When there are no finite buffers, $\mathbb{E}\int_0^\infty e^{-bt}A(t)\ dt$ is independent of the policy, because $\mathbb{E}A(t) = \lambda t$ for all t and $\int_0^\infty \mid e^{-bt}\lambda t \mid\ dt < \infty$. Thus, minimizing (4) is the same as maximizing

$$\mathbb{E}\int_0^\infty e^{-bt}D(t)\ dt.$$

For each realization $T_1, T_2, T_3, \ldots$ with T_k the k-th departure time, it holds that

$$\begin{aligned}\int_0^\infty e^{-bt}D(t)\ dt &= \sum_{k=1}^\infty \int_{T_k}^\infty e^{-bt}\ dt \\ &= \frac{1}{b}\sum_{k=1}^\infty e^{-bT_k}.\end{aligned}$$

This is exactly $\frac{1}{b}$ times the reward we would get for receiving a discounted reward of 1 for each departure, and this is just the reward function we consider.

In this model, the transition times (times between decision epochs) in different states may have different distributions. For example, in the empty system there is only one possible event, namely the arrival of a customer in the first centre. The transition time, the time until this event occurs, has an exponential distribution with rate λ. In the state where the system is full, the only event which can occur, is the departure of a customer in centre 2. This transition time has an exponential distribution with rate μ_2 or $2\mu_2$, respectively when there are customers in only one queue or in both queues.
When the transition times have different distributions, the expected one step costs until the next decision epoch have to be calculated separately. Therefore, we use a uniformization technique to get a model in which the transition times are exponentially distributed with the same parameter for all the states (cf. [8]). The subsequent states are described by a Markov decision chain, to which we will apply the algorithms. In this case, we choose the parameter of the uniform exponential distribution equal to the maximal parameter in the original model, namely $\lambda + 2\mu_1 + 2\mu_2$. Thus, in each state all events are possible, namely an arrival in the first centre and departures in both queues of both centres. If an event occurs which is not possible in reality, a dummy transition takes place (transition to the same state).

Now, some examples of transition probabilities are

$$\begin{aligned}p_{(i_1,i_2,i_3,i_4)a(i_1,i_2+1,i_3,i_4)} &= \frac{\lambda}{\lambda+2\mu_1+2\mu_2}\ \delta(a=2,\ i_1+i_2<N_1);\\ p_{(i_1,i_2,i_3,i_4)a(i_1-1,i_2,i_3+1,i_4)} &= \frac{\mu_1}{\lambda+2\mu_1+2\mu_2}\ \delta(i_1>0,\ i_3+i_4<N_2,\ i_3\le i_4)\\ &\quad (1-\frac{1}{2}\ \delta(i_3=i_4)).\end{aligned}$$

Here, $\delta(\text{condition } 1, \text{condition } 2, \ldots, \text{condition } n)$ is equal to 1 if all conditions are true and equal to 0 otherwise.

Finally, we have to define our expected one step cost. As described before, we have a cost of -1 when a customer leaves the system. To compute the one step cost, we have to take the discount factor $\alpha \in [0,1)$ into account. Suppose the discount factor in the original continuous model to be equal to α. Costs at time t are discounted by a factor $\alpha^t = e^{-bt}$ with $b = -\ln(\alpha)$. Now, for a state $i \in E$ with a positive number of customers in the first queue of centre 2

$$\begin{aligned} c &= \int_0^\infty e^{-bx} \frac{\mu_2}{\lambda + 2\mu_1 + 2\mu_2} (\lambda + 2\mu_1 + 2\mu_2) e^{-(\lambda+2\mu_1+2\mu_2)x} \, dx \\ &= \frac{\mu_2}{\lambda + 2\mu_1 + 2\mu_2 + b} \end{aligned}$$

is the expected discounted number of departures from this queue.

When there are customers present in both queues of centre 2, the expected discounted number of departures from centre 2 until the next decision epoch will be twice the expression above. Hence,

$$c_{(i_1,i_2,i_3,i_4)a} = -c\left(\delta(i_3 > 0) + \delta(i_4 > 0)\right).$$

Because the factor $\mu_2/(\lambda + 2\mu_1 + 2\mu_2 + b)$ is a constant, we take it for simplicity equal to 1. This will make no difference in the policy found; the total discounted cost of each policy will be multiplied by $(\lambda + 2\mu_1 + 2\mu_2 + b)/\mu_2$.

5. Numerical results

In this section we present numerical results found by the algorithms in the Section 3.

In the model with full information, the optimal routing policy is almost equal to the SQP for many different parameter values. Therefore, we expected the optimal routing policy in the partial information model to be equal to the SQP. For many parameter values, the SQP is indeed optimal. However, in [3] some parameters are given for which the SQP is not optimal. We also found this with the algorithms of Section 3. For example, the routing policy found by these algorithms deviates from the SQP, when the initial state is $(1,0,10,10)$ and the parameters are the following : $N_1 = 20$, $N_2 = 25$, $\lambda = 0.01$, $\mu_1 = \mu_2 = 1$ and $\alpha = 0.8991$ (this is the discount factor in the discrete time Markov decision chain). Indeed, the policy found, turns out to route arriving customers to the first queue in states in the partition set $E_{(1,0)}$. Heuristically, this can be explained as follows. Because of the discounting, the initial state has a big influence on the total discounted costs. In this state, there is one customer in the first queue of centre 1 and the second centre is rather full and balanced. By routing an arriving customer to the first queue in centre 1 instead of routing him to the second queue, the arrival of this customer in the second centre will be delayed more. Therefore, the second centre will probably be less full and less balanced when he finally arrives there, which results in a better routing in the second centre.

For the initial policy in Algorithm 2, we choose the SQP to reduce the computing times. In the cases considered, the SQP is (almost) the optimal policy. Hence, not many policy improvements need to be made by the algorithm and the computing time is small. However, for other initial policies the computing times can become very large. This can be seen from Table 1 (cf. [5]).

For several values of the buffer sizes, the computing times are given for three different initial policies; the SQP, the policy assigning all customers to the first queue and the policy that assigns customers with probability $\frac{1}{2}$ to the first and with probability $\frac{1}{2}$ to the second queue (our computer code of Algorithm 2 allows for an initial randomized policy). The policies are denoted as 'SQP', 'Queue 1' and 'Bernoulli' respectively.

N_1	N_2	Initial policy	Computing time
20	20	SQP	3 hours
15	15	SQP	1.5 hours
10	10	SQP	11 minutes
8	8	SQP	7 minutes
6	4	SQP	1.5 minutes
20	20	Queue 1	138.5 hours
15	15	Queue 1	43 hours
10	10	Queue 1	3 hours
8	8	Queue 1	1 hour
6	4	Queue 1	11 minutes
15	15	Bernoulli	84.5 hours
10	10	Bernoulli	6 hours
8	8	Bernoulli	1.75 hours
6	4	Bernoulli	21 minutes

Table 1. Computing times for different initial policies in Algorithm 2.

The results were found, using a slower computer than in Tables 2 up to 5, thus the computing times are not comparable to the other computing times in this section. However, it is clear that the computing time becomes very large if the initial policy is not close to the optimal one. Algorithm 1 is not influenced by the choice of the initial policy. Furthermore, the choices of x^1 and v^1 have very little impact on the computing times. This means that the computing time is virtually independent of the initialization of the algorithm.
Hence, in general models where we have no good guess for the optimal policy, Algorithm 1 is much faster than Algorithm 2.

In our implementation of the third algorithm, we use the discounted costs of the SQP (computed by iteration of the v-vector) as initial upper bound on the costs. This seems a good choice, because we expect the optimal policy to be close to the SQP.
We also implement the algorithm, such that the order of the sets E_s with $d_s = 0$, chosen in step 2 of the algorithm, is the following (if possible) : $E_{(0,1)}$, $E_{(1,0)}$, $E_{(0,2)}$, $E_{(2,0)}$, $E_{(1,2)}$, $E_{(2,1)}$, $E_{(0,3)}$, $E_{(3,0)}$, $E_{(1,3)}$, $E_{(3,1)}$. When the policy is already fixed for the partition sets mentioned, the choice is made by choosing the state with the lowest number. We choose

to fix the policy first in these states, because these states are close to the initial state and are likely to have the biggest influence on the costs. Furthermore, when an action has to be chosen in step 2, the action deviating from the SQP is chosen first. This, in the hope that the corresponding lower bound is so high that the node is not appended to the tree. The computing time for the Algorithm 3 depends very much on the order in which the subsets of the partition are chosen to fix the policy there. In general, this algorithm will be slowest.

Numerical results for the three algorithms can be found in the Tables 2 – 5.
In the tables, we give results for parameters close to the parameters where we found the policy to be different from the SQP. The policy SQP′ in the tables is defined as the policy which chooses action 1 in sets $E_{(i,j)}$ with $i \leq j$ and chooses action 2 otherwise. Note, that in contrast to the SQP, in sets $E_{(i,i)}$ action 1 is chosen instead of randomizing between the two actions. We choose this initial policy because we want to allow only deterministic policies in our implementations.
In all tables, the (discrete time) discount factor α is equal to $(\lambda + 2\mu_1 + 2\mu_2)/(\lambda + 2\mu_1 + 2\mu_2 + b)$.
In the column 'Sets not SQP′ ', the entry (i, j) means that the policy found by the algorithm considered, deviates from the SQP′ in the subset $E_{(i,j)}$.
For ϵ, we use the value 10^{-10}.

In Table 2, the results for different λ can be found. The other parameters are $N_1 = 20$, $N_2 = 25$, $\mu_1 = \mu_2 = 1$ and $b = 0.5$. The initial distribution vector β is chosen as $\beta_i = 1$ if $i = (1, 0, 10, 10)$ and zero otherwise. Thus, the initial state is $(1, 0, 10, 10)$.

	λ	Sets not SQP′	Cost per time unit
Algorithm 1	0.005	(0,0), (1,0), (1,1)	-17.7721
Algorithm 2	0.005	(0,0), (1,0), (1,1)	-17.7721
Algorithm 3	0.005	(0,0), (0,3), (1,0), (1,2) (1,3), (3,0), (5,7)	-17.7721
Algorithm 1	0.010	(0,0), (1,0), (1,1), (2,2)	-17.7941
Algorithm 2	0.010	(0,0), (1,0), (1,1), (2,2)	-17.7941
Algorithm 3	0.010	(0,0), (1,0), (1,3), (3,0), (5,4) (5,5), (6,6), (7,7), (8,8), (9,9)	-17.7941
Algorithm 1	0.015	(0,0), (1,0), (1,1), (2,2)	-17.8161
Algorithm 2	0.015	(0,0), (1,0), (1,1), (2,2)	-17.8161
Algorithm 3	0.015	(0,0), (0,3), (1,0), (1,1), (1,3) (3,0), (3,1), (5,4), (5,5), (6,6) (7,7), (8,8), (9,9)	-17.8161
Algorithm 1	0.020	(0,0), (1,0), (1,1), (2,2)	-17.8381
Algorithm 2	0.020	(0,0), (1,0), (1,1), (2,2)	-17.8381
Algorithm 3	0.020	(0,0), (0,3), (1,0), (1,1), (1,3) (3,0), (3,1), (5,4), (5,5), (6,6) (7,7), (8,8), (9,9)	-17.8381

	λ	Sets not SQP′	Cost per time unit
Algorithm 1	0.025	(0,0), (1,0), (1,1), (2,2)	-17.8601
Algorithm 2	0.025	—	-17.8600
Algorithm 3	0.025	(0,0), (0,3), (1,0), (1,1), (1,3) (3,0), (3,1), (5,4), (5,5), (6,6) (7,7), (8,8), (9,9)	-17.8601
Algorithm 1	0.030	(0,0), (1,0), (1,1), (2,2)	-17.8820
Algorithm 2	0.030	—	-17.8820
Algorithm 3	0.030	(0,0), (0,3), (1,0), (1,1), (1,3) (3,0), (3,1), (5,4), (5,5), (6,6) (7,7), (8,8), (9,9)	-17.8820
Algorithm 1	0.035	(0,0), (1,0), (1,1), (2,2)	-17.9040
Algorithm 2	0.035	—	-17.9039
Algorithm 3	0.035	(0,0), (1,0), (1,1), (3,0), (5,4) (5,5), (6,6), (7,7), (8,8), (9,9)	-17.9040
Algorithm 1	0.040	—	-17.9259
Algorithm 2	0.040	—	-17.9259
Algorithm 3	0.040	(0,0), (0,3), (1,0), (1,1), (1,3), (5,4)	-17.9259
Algorithm 1	0.045	—	-17.9478
Algorithm 2	0.045	—	-17.9478
Algorithm 3	0.045	(0,0), (1,0), (1,1), (5,4), (5,5), (6,6) (7,7), (8,8), (9,9)	-17.9478
Algorithm 1	0.050	—	-17.9697
Algorithm 2	0.050	—	-17.9697
Algorithm 3	0.050	(0,0), (1,0), (1,1), (2,2), (3,0), (5,4) (5,5), (6,6), (7,7), (8,8), (9,9)	-17.9698

Table 2. Results for varying λ.

In Table 3, the results for different μ_2 can be found. The other parameters are $N_1 = 20$, $N_2 = 25$, $\lambda = 0.005$, $\mu_1 = 1$ and $b = 0.5$. The initial state is $(1, 0, 10, 10)$.

	μ_2	Sets not SQP′	Cost per time unit
Algorithm 1	1.25	(0,0), (1,1)	-19.4482
Algorithm 2	1.25	—	-19.4482
Algorithm 3	1.25	—	-19.4482
Algorithm 1	1.5	—	-20.9654
Algorithm 2	1.5	—	-20.9654
Algorithm 3	1.5	—	-20.9654
Algorithm 1	1.75	—	-22.3289
Algorithm 2	1.75	—	-22.3289
Algorithm 3	1.75	—	-22.3289
Algorithm 1	2	—	-23.5509
Algorithm 2	2	—	-23.5509
Algorithm 3	2	—	-23.5509

Table 3. Results for varying μ_2.

In Table 4, the results for different μ_1 can be found. The other parameters are $N_1 = 20$, $N_2 = 25$, $\lambda = 0.005$, $\mu_2 = 1$ and $b = 0.5$. The initial state is $(1, 0, 10, 10)$.

	μ_1	Sets not SQP′	Cost per time unit
Algorithm 1	1.25	(0,0), (1,0), (1,1), (2,0)	-19.7434
Algorithm 2	1.25	(0,0), (1,0), (1,1), (2,0)	-19.7434
Algorithm 3	1.25	(0,0), (0,2), (1,0), (1,3) (2,0), (3,1), (9,9)	-19.7434
Algorithm 1	1.5	(0,0), (1,0), (1,1), (2,0)	-21.7148
Algorithm 2	1.5	(0,0), (1,0), (1,1), (2,0)	-21.7148
Algorithm 3	1.5	(0,0), (0,2), (0,3), (1,0) (1,3), (2,0), (2,1), (13,6)	-21.7148
Algorithm 1	1.75	(0,0), (1,0), (1,1), (2,0), (3,0)	-23.6861
Algorithm 2	1.75	(0,0), (1,0), (1,1), (2,0), (3,0)	-23.6861
Algorithm 3	1.75	*	
Algorithm 1	2	(0,0), (1,0),(1,1),(2,0), (3,0), (4,0)	-25.6575
Algorithm 2	2	(0,0), (1,0), (1,1), (2,0), (3,0), (4,0)	-25.6575
Algorithm 3	2	*	

Table 4. Results for varying μ_1.

* These entries are omitted because the computing times were too large.

The results in Tables 2, 3 and 4 are found on a SGI Challenge R4400SC with 128 Mbyte memory and 500 Mbyte swap space. The total of computing times (in hours) needed per algorithm for all the problem instances in the tables are

	Table 2	Table 3	Table 4
Algorithm 1 :	4 : 03	4 : 33	1 : 45
Algorithm 2 :	2 : 42	2 : 59	2 : 25
Algorithm 3 :	596 : 31	5 : 48	$> 800 : 00$.

In Table 5, the results for different b, thus for different discount factors, can be found. The other parameters are $N_1 = 20$, $N_2 = 25$, $\lambda = 0.005$ and $\mu_1 = \mu_2 = 1$. The initial state is $(1, 0, 10, 10)$.

	b	Sets not SQP′	Cost per time unit
Algorithm 1	0.6	(0,0), (1,0), (1,1)	-15.2436
Algorithm 2	0.6	(0,0), (1,0), (1,1)	-15.2436
Algorithm 3	0.6	(0,0), (0,3), (1,0), (1,2) (1,3), (3,0), (5,6)	-15.2436
Algorithm 1	0.7	(0,0), (1,0), (1,1)	-13.3940
Algorithm 2	0.7	(0,0), (1,0), (1,1)	-13.3940
Algorithm 3	0.7	(0,0), (0,3), (1,0), (1,2), (1,3) (2,1), (3,0), (3,1), (4,6)	-13.3940
Algorithm 1	0.8	—	-11.9887
Algorithm 2	0.8	—	-11.9887
Algorithm 3	0.8	—	-11.9887
Algorithm 1	0.9	—	-10.8879
Algorithm 2	0.9	—	-10.8879
Algorithm 3	0.9	—	-10.8879
Algorithm 1	1	—	-10.0036
Algorithm 2	1	—	-10.0036
Algorithm 3	1	—	-10.0036

Table 5. Results for varying b.

The results are found on a Silicon Graphics Indy R4000PC with 32 Mbyte memory and 100 Mbyte swap space. The total of computing times (in hours) needed per algorithm for all the problem instances in the table are

Algorithm 1 : 2 : 55;
Algorithm 2 : 3 : 11;
Algorithm 3 : 108 : 56.

Note, that the policies found by Algorithm 3, are globally optimal. From the tables, we see that the policies found by the three different algorithms are almost the same. Hence, the policies found by Algorithm 1 and 2 are (nearly) optimal too. In Table 2, for $\lambda = 0.025$ and $\lambda = 0.035$, we have that Algorithm 1 and 3 find better policies than Algorithm 2 and for $\lambda = 0.050$ Algorithm 3 finds a better policy than the other two algorithms. However, the difference in costs is very small.
From all this, we can conclude that Algorithm 1 finds good, if not optimal, policies in this model in a (relatively) short time.

References

[1] Hordijk, A. & Koole, G. (1992). On the assignment of customers to parallel queues. *Probability in the Engineering and Informational Sciences* 6: 495–511.

[2] Hordijk, A. & Loeve, J.A. (1994). Undiscounted Markov decision chains with partial information; an algorithm for computing a locally optimal periodic policy. *ZOR-Mathematical Methods of Operations Research* 40: 163–181.

[3] Koole, G. (1992). Stochastic scheduling and dynamic programming. Ph.D. thesis, University of Leiden.

[4] Kulkarni, V.G. & Serin, Y. (1995). Optimal implementable policies : discounted cost case. *Computations with Markov chains*, W. Stewart (editor), 283–307. Kluwer Academic Publications.

[5] Loeve, J.A. & Pols, M.A.H. (1990). Optimal assignment of customers in queueing models with two centres (in Dutch). Master thesis, University of Leiden.

[6] Puterman, M.L. (1994). *Markov Decision Processes.* Wiley, New York.

[7] Schneeberger, S. (1992). Markov-Entscheidungs-Prozesse mit abhängigen Aktionen für optimale Reparaturmaßnahmen bei unvollständiger Information. *OR Spektrum* 14: 71–78.

[8] Tijms, H.C. (1988). *Stochastic modelling and analysis : a computational approach.* Wiley, Chichester.

Analysis of a Generalized Shortest Queue System by Flexible Bound Models

G.J. van Houtum
Faculty of Mechanical Engineering
University of Twente
E-mail: g.j.j.a.n.vanhoutum@wb.utwente.nl

Abstract

Motivated by a practical situation for the production/assembly of Printed Circuit Boards, we study a generalized shortest queue system. This system consists of parallel servers, which all have their own queue. The system serves several types of jobs, which arrive according to Poisson processes. Because of technical reasons, most or all types of arriving jobs can only be served by a restricted set of servers. All jobs have the same exponential service time distribution, and, in order to minimize its own service time, each arriving job joins (one of) the shortest queue(s) of all queue(s) where the job can be served. The behavior of the resulting queueing system may be described by a multi-dimensional Markov process. Since an analytical solution for this Markov process is hard to obtain, we present flexible bound models in order to find the most relevant performance measures, viz. the waiting times for each of the job types separately and for all job types together. The effectiveness of the flexible bound models is shown by some numerical results.

1 Introduction

To show the relevance of the queueing system studied in this paper, we first describe a queueing situation stemming from a flexible assembly system consisting of a group of parallel insertion machines, which have to mount vertical components on Printed Circuit Boards. We start the description with explaining how an *insertion machine* operates. An insertion machine mounts vertical components, such as resistors and capacitators, on a Printed Circuit Board (PCB) by the *insertion head*. The components are mounted in a certain sequence, which is prescribed by a Numerical Control program. The insertion head is fed by the *sequencer*, which picks components from tapes and transports them in the right order to the insertion head. Each tape contains only *one* type of components. The tapes are stored in the *component magazine*, which may contain 80 tapes, say. Each PCB needs, on average, 60 different types of components. If a machine has to mount components on a PCB, then all the components need to be available on that machine. That means that for all those components a tape must be placed in the magazine. So the set of components available on the machine completely determines which types of PCBs can be handled.

In general we have a group of parallel insertion machines which have to process a number of different types of PCBs at the same time. Each insertion machine has its own queue, and

the PCBs are transported to the insertion machines by an Automatic Conveyor System. In Figure 1, we have depicted a system which consists of three insertion machines and which has to process three different types of PCBs. The machines are basically similar, but due to the fact that they may be loaded with different types of components, the classes of PCB-types that can be handled by the machines may be different. In the situation depicted in Figure 1, machine M_1 can handle PCBs of the types A and B, machine M_2 can handle the types A and C, and machine M_3 can handle the types B and C.

In fact, there are two decision problems: the *assignment problem* and the *routing problem.* We first describe the assignment problem, which is the major problem. The assignment problem concerns how the tapes with components have to be divided among the machines. One should try to allocate the tapes with components to the machines such that, for example, the waiting times (and/or sojourn times) of the PCBs are minimized. There would be no problem if the magazines were big enough to contain all components needed to process all types of PCBs. However, in general they can only contain the components needed for a small subset of the different types of PCBs.

In order to solve the assignment problem, we must be able to evaluate the performance characteristics of a *given assignment* of the components to the machines. These performance characteristics depend on how the second decision problem, i.e. the routing problem, is handled. This problem concerns to which machines the PCBs must be sent upon arrival. For an arriving PCB, we must select one of the machines which can handle that PCB. If for all types the mounting times are roughly the same, then it is reasonable to select the machine with the *shortest queue* (let ties be broken with equal probabilities); this at least (roughly) minimizes the waiting time of the arriving PCB itself, and it may be expected that this also roughly minimizes the average waiting time for all PCBs together, provided that we are in a balanced situation (i.e. a situation in which each server will have to handle the same amount of work on average). Assume that the shortest queue routing is used by the Automatic Conveyer System, and that, once arrived in a queue, the PCBs are served in a First-Come-First-Served (FCFS) manner. Then we have the following problem:

> *Given the shortest queue routing and the FCFS service discipline at each machine, we want to have an efficient method for the determination of the performance characteristics of the flexible assembly system for a given assignment of the components to the machines.*

The main performance characteristics we are interested in, are the waiting times for each type of PCBs separately and for all PCBs together. It is obvious that an efficient method for determining these measures can be exploited for selecting the best possible assignment of the components to the machines.

The assembly of PCBs is often characterized by relatively few job types, large production batches and small processing times (see Zijm [13]). Therefore, a queueing model approach seems natural. The flexible assembly system can be modeled as a queueing system consisting of parallel servers, each with a own queue, and serving several types of jobs, where each job upon arrival joins the shortest queue of all queues that can handle this job. We call this system a *Generalized Shortest Queue System (GSQS).*

Apart from the situation described above, the GSQS is also relevant for many other practical situations; for example, in a job shop with a group of identical, parallel machines which are loaded with different sets of tools, in a computer system where each information

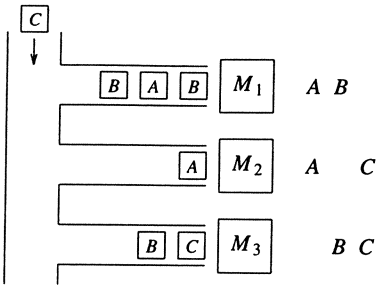


Figure 1: A flexible assembly system consisting of three parallel insertion machines, on which three types of PCBs are made.

file is available on a restricted set of a number of parallel disks and requests for information files have to be handled by only one disk, and at a banking office where each clerk is able to carry out a restricted set of tasks. Nevertheless, the GSQS has hardly been studied in the literature. To our knowledge, the only contribution is made by Adan, Wessels and Zijm [3], who, for a simplified situation (see the next paragraph), present rough approximations for the waiting times. Further, closely related systems have been studied by Schwartz [10] (see also Roque [9]), Green [5] and Hassin and Haviv [6].

In this paper, we make the following assumptions for the GSQS (cf. [3]): (i) all jobs arrive according to Poisson streams; (ii) the service times are exponentially distributed; (iii) the service times are job-independent; (iv) all insertion machines work equally fast. The assumptions (ii)-(iv) imply that all service times are exponentially distributed with the same parameter. Even under these assumptions, the GSQS constitutes a hard problem. The behavior of the GSQS is described by a continuous-time Markov process with multi-dimensional states where each component denotes the queue length at one of the servers. However, because of the shortest queue routing, the structure of the transitions is rather complicated and hence an analytical solution seems hard to obtain in general. In fact, an analytical solution is only known for the special case with two parallel servers and one type of jobs that can be handled by both servers; in this case the GSQS reduces to the two-dimensional symmetric shortest queue system, for which a generalized product-form solution has been derived by using a compensation approach (see [4]). For all other cases, even a standard numerical method is not available. Therefore, for the general case of the GSQS, we propose to use truncation models which: (i) have a truncated state space with a flexible size (i.e. depending on one or more truncation parameters); (ii) can be solved efficiently; (iii) lead to upper/lower bounds for the waiting times. Such models are called *solvable flexible bound models*. We shall define one lower bound and one upper bound model. By solving these two models for increasing sizes of the truncated state space, we can determine the waiting times of the original GSQS as accurately as desired. Numerical results for two series of instances will show that this method may work quite well. It is noted that flexible bound models previously have been successfully applied to the symmetric shortest queue system (with ≥ 2 servers), the symmetric longest queue system and an $M|M|c$ system with critical jobs (see [2, 12, 1]).

This paper is organized as follows. In Section 2, we give a precise description of our model for the GSQS. Next, in Section 3, we describe the flexible bound models that can be used

to determine the waiting times for the GSQS. Finally, in Section 4, we present numerical results in order to show the effectiveness of the flexible bound models. For simplicity and in order to save space, in the remaining part of this paper we shall restrict ourselves to the two-dimensional case, i.e. to a GSQS consisting of two servers. Nevertheless, the whole analysis can easily be generalized to the case with two or more servers; for this generalization the reader is referred to [11].

2 Model

We consider a GSQS consisting of two parallel servers. For this system we distinguish three types of jobs: jobs of type A, which can be served by both servers, jobs of type B, which can only be served by server 1, and jobs of type C, which must be served by server 2; see Figure 2. The jobs of the types A, B and C arrive according Poisson processes with intensities λ_A, λ_B and λ_C (all ≥ 0). The total arrival intensity is denoted by $\lambda = \lambda_A + \lambda_B + \lambda_C$. All service times are assumed to be exponentially distributed with parameter $\mu = 1$. Upon arrival, jobs of type B join the queue at server 2, jobs of type C join the queue at server 3, and jobs of type A join the shortest queue (if both queues have equal length, then each queue is chosen with probability $\frac{1}{2}$).

The behavior of the GSQS is described by a continuous-time Markov process with states (m_1, m_2), where m_i denotes the length of the queue at server i, $i = 1, 2$ (jobs in service are included). So, the state space is equal to

$$M = \{m \mid m = (m_1, m_2) \text{ with } m_i \in \mathbb{N}_0 \text{ for } i = 1, 2\} .$$

In order to obtain an irreducible Markov process, we assume that $\lambda_A + \lambda_B > 0$ and $\lambda_A + \lambda_C > 0$. The transition rates are denoted by $q_{m,n}$. These rates have been depicted in Figure 3.

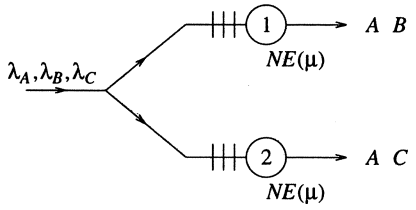


Figure 2: The GSQS with two servers and three job types.

The average workload per server is given by $\rho = \lambda/2$. The GSQS obviously can only be ergodic if $\rho < 1$ and if each of the servers can handle the job type that always has to be served by him, i.e. if

$$\lambda_B < 1, \quad \lambda_C < 1 \text{ and } \lambda < 2. \tag{1}$$

We *conjecture* that this condition is not only necessary, but *also sufficient* for the ergodicity. This conjecture is based on: (i) the idea that the *dynamic* shortest queue routing gives a better performance than a *static* routing; (ii) the property that if condition (1) is satisfied, then there exists a static routing under which the system is ergodic. The latter property

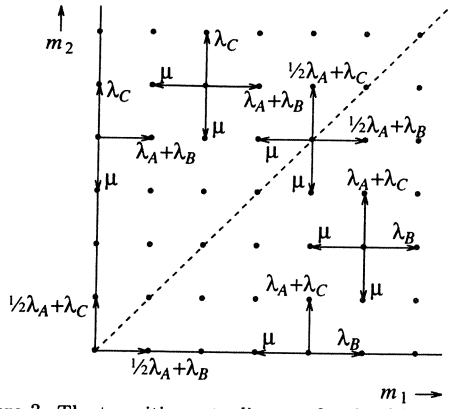


Figure 3: The transition rate diagram for the GSQS.

is seen as follows. Under a static routing, upon arrival, a job of type A joins the queue at server 1 with some given probability x, $0 \leq x \leq 1$, and it joins the queue at server 2 with probability $1 - x$. Then the two servers behave as two independent $M|M|1$ queues with workloads $x\lambda_A + \lambda_B$ and $(1-x)\lambda_A + \lambda_C$, respectively, and the system is ergodic if $x\lambda_A + \lambda_B < 1$ and $(1-x)\lambda_A + \lambda_C < 1$. It may be shown that this latter condition always can be satisfied for some choice of x if condition (1) is satisfied. From now on, we assume that condition (1) is satisfied.

The performance measures we are interested in are the mean waiting times $W^{(A)}$, $W^{(B)}$, $W^{(C)}$ for each of the job types A, B and C separately and the mean waiting time W for all job types together, which is equal to

$$W = \frac{\lambda^{(A)}}{\lambda} W^{(A)} \frac{\lambda^{(B)}}{\lambda} W^{(B)} \frac{\lambda^{(C)}}{\lambda} W^{(C)} .$$

It is easily seen that $W^{(B)}$ and $W^{(C)}$ are equal to the mean queue lengths L_1 and L_2 at the servers 1 and 2, respectively, and that $W^{(A)}$ is equal to the mean L_{sq} of the length of the shortest queue.

Finally, note that the GSQS is *symmetric* if $\lambda_B = \lambda_C$. For such a system, the ergodicity condition (1) reduces to $\rho < 1$ and the shortest queue routing used by the jobs of type A can be shown to minimize the total number of jobs in the system and hence also the mean waiting time W (this may be done by the technique used by Hordijk and Koole [7]).

3 Solution by flexible bound models

We now define two truncation models: one leading to lower bounds for the waiting times $W^{(A)}$, $W^{(B)}$, $W^{(C)}$ and W, and another one leading to upper bounds.

Since the shortest queue routing in general will cause a drift to the states with equal queue lengths, for both the lower and the upper bound model the truncated state space is

defined by

$$M' = \{m \in M \mid m = (m_1, m_2),\ m_1 \leq m_2 + T_1 \text{ and } m_2 \leq m_1 + T_2\} ,$$

where $T_1, T_2 \in \mathbb{N}$ are so-called threshold parameters. For this choice of the truncated state space, there are four types of transitions pointing from states inside M' to states outside M':

(i) for the states $m = (m_1, m_1 + T_1) \in M'$ with $m_1 > 0$, a service completion at server 1 occurs with rate μ and leads to a transition from m to state $n = m - e_1 \notin M'$;

(ii) for the states $m = (m_2, m_2 + T_2) \in M'$ with $m_2 > 0$, a service completion at server 2 occurs with rate μ and leads to a transition from m to state $n = m - e_2 \notin M'$;

(iii) for the states $m = (m_1, m_1 + T_1) \in M'$ with $m_1 \geq 0$, an arrival of a job of type C occurs with rate λ_C and leads to a transition from m to state $n = m + e_2 \notin M'$;

(iv) for the states $m = (m_2, m_2 + T_2) \in M'$ with $m_2 \geq 0$, an arrival of a job of type B occurs with rate λ_B and leads to a transition from m to state $n = m + e_1 \notin M'$.

In the lower bound model, these transitions are redirected from the states n to states n' which correspond to situations with a smaller number of jobs at one of the two servers. With respect to waiting times and queue lengths these states are more attractive. In the upper bound model, redirections are made to less attractive states corresponding to situations with a larger number of jobs at one of the two servers.

In the lower bound model, the transitions described under (i) and (ii) are redirected to the states $n' = n - e_2 = m - e_1 - e_2 \in M'$ and $n' = n - e_1 = m - e_1 - e_2 \in M'$, respectively. The physical interpretation of these redirections is that a departure of a job at a non-empty shortest queue is accompanied by a destruction or killing of one job at the other queue. Further, the transitions described under (iii) and (iv) are redirected to the states $n' = n - e_2 = m \in M'$ and $n' = n - e_1 = m \in M'$, i.e. to the states m itself. The physical interpretation of these redirections is that a new job arriving at one of the servers is rejected. Because of the physical interpretations, the lower bound model is called the *Threshold Killing and Rejection (TKR) model.*

In the upper bound model, the transitions described under (i)-(iv) are redirected to $n' = n + e_1 = m$, $n' = n + e_2 = m$, $n' = n + e_1 = m + e_1 + e_2$ and $n' = n + e_2 = m + e_1 + e_2$, respectively. The meaning behind the first two types of redirections is that if for one queue the difference with respect to the shortest queue has already reached its maximum value, then a service completion at the other queue is not accompanied by a departure, and the job in service has to be served once more; this is equivalent to saying that then the other server is blocked. The meaning behind the latter two types of redirections is that an arrival of a new job at a queue for which the difference with respect to the shortest queues has already reached its maximum value, is accompanied by the addition of one extra job at each of the shortest queues. Hence, the upper bound model is called the *Threshold Blocking and Addition (TBA) model.*

In Figure 4, we have depicted the redirections for both the lower and upper bound model.

The TKR model leads to stochastically smaller lengths for the queue at server 1, the queue at server 2 and the shortest queue, and hence also to smaller means than obtained for the original model. Further, it may be shown that the larger the values of T_1 or T_2 the

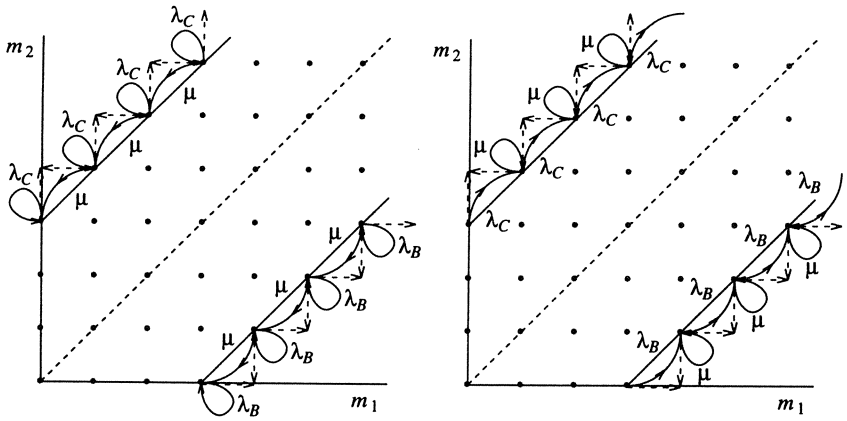


Figure 4: The redirections for the TKR and TBA model. For both models, T_1 and T_2 have been taken equal to 3.

smaller the difference between the queue lengths in the TKR model and the queue lengths in the original model. The lower bounds for the mean queue lengths immediately lead to lower bounds $W^{(A)}_{TKR}(\hat{T})$, $W^{(B)}_{TKR}(\hat{T})$, $W^{(C)}_{TKR}(\hat{T})$ and $W_{TKR}(\hat{T})$ for the mean waiting times; here $\hat{T} = (T_1, T_2)$. Similarly, the TBA model leads to larger queue lengths and waiting times. The upper bounds for the mean waiting times are denoted by $W^{(A)}_{TBA}(\hat{T})$, $W^{(B)}_{TBA}(\hat{T})$, $W^{(C)}_{TBA}(\hat{T})$ and $W_{TBA}(\hat{T})$. A formal proof of all these monotonicity results may be given by using the *precedence relation method.* This method is based on Markov reward theory and has been developed in [11].

For both the TKR and TBA model, the steady-state distribution can be determined by the *matrix-geometric approach*, as described in [8]. This enables an efficient computation of the corresponding lower and upper bounds for the waiting times; see [11] for appropriate matrix formulae that can be used for this computation.

4 Numerical results

In this final section, numerical results for two series of instances are presented in order to show how well the waiting times of the original GSQS can be determined by using the bound models. The instance with

$$\rho = 0.9, \quad \lambda = 2\rho, \quad \lambda_A = p\lambda \text{ with } p = \frac{1}{2}, \quad \lambda_B = \lambda_C = \frac{1}{2}(1-p)\lambda$$

has been chosen as a basic instance. In the first series, we have varied the value of the workload ρ. In the second series, we have varied the value of the fraction p of jobs that can be handled by both servers.

Since all instances concern symmetric cases, we can take $T_1 = T_2 = T$ and the waiting times $W^{(A)}$, $W^{(B)}$, $W^{(C)}$ and W can be determined by solving the TKR and TBA model

ρ	T	$W^{(A)}$	$\Delta^{(A)}(\hat{T})$	$W^{(B)}$	$\Delta^{(B)}(\hat{T})$	W	$\Delta(\hat{T})$
0.1	2	0.0146	0.0006	0.1059	0.0007	0.0603	0.0006
0.2	3	0.0558	0.0006	0.2282	0.0008	0.1420	0.0007
0.3	3	0.1281	0.0034	0.3746	0.0043	0.2514	0.0038
0.4	4	0.2351	0.0030	0.5577	0.0038	0.3964	0.0034
0.5	5	0.3966	0.0034	0.7977	0.0042	0.5971	0.0038
0.6	7	0.6468	0.0018	1.1337	0.0021	0.8902	0.0019
0.7	8	1.0723	0.0039	1.6532	0.0044	1.3628	0.0041
0.8	11	1.9222	0.0027	2.6142	0.0029	2.2682	0.0028
0.9	15	4.4516	0.0032	5.2782	0.0033	4.8649	0.0033
0.95	18	9.4729	0.0048	10.3800	0.0048	9.9265	0.0048
0.98	23	24.4883	0.0032	25.4495	0.0032	24.9689	0.0032
0.99	26	49.4939	0.0031	50.4742	0.0031	49.9841	0.0031

Table 1: The mean waiting times $W^{(\cdot)}$ and W determined within an absolute accuracy of $\epsilon_{abs} = 0.005$ for increasing values of ρ and with $\lambda = 2\rho$, $\lambda_A = \lambda$, $\lambda_B = \lambda_C = \lambda$.

for increasing values of T. Here, for each T, the values of $(W^{(A)}_{TKR}(\hat{T}) + W^{(A)}_{TBA}(\hat{T}))/2$ and $\Delta^{(A)}(\hat{T}) = (W^{(A)}_{TBA}(\hat{T}) - W^{(A)}_{TKR}(\hat{T}))/2$, where $\hat{T} = (T, T)$, are used as an approximation for $W^{(A)}$ and an upper bound for the corresponding absolute inaccuracy; and, similarly for $W^{(B)}$, $W^{(C)}$ and W. For each instance, we have determined the smallest value of T for which each of the waiting times was determined within an absolute accuracy $\epsilon_{abs} = 0.005$.

The numerical results have been gathered in the Tables 1 and 2. The first column of Table 1 denotes the chosen values for ρ, while the second column depicts the value found for T. In the third, fifth and seventh column, we have listed the approximations which for this T have been obtained for $W^{(A)}$, $W^{(B)} = W^{(C)}$ (because of the symmetry, also the waiting times for the types B and C are equal) and W; and, in the fourth, sixth and eighth column, we have listed the upper bounds $\Delta^{(A)}(\hat{T})$, $\Delta^{(B)}(\hat{T}) = \Delta^{(C)}(\hat{T})$ and $\Delta(\hat{T})$ for the corresponding absolute inaccuracies. Table 2 consists of the same columns, except that in this table the first column denotes the chosen values of p.

The results in Table 1 show that, as expected, the threshold parameter T which is needed to approximate the mean waiting times within the desired absolute accuracy, is increasing as a function of the workload ρ. Further, the results in the Table 2 show that the required value for T strongly depends on the strength of the drift to the states with equal queue lengths, i.e. to the states on the diagonal. In this table, a smaller value for p corresponds to a weaker drift to the states on the diagonal. It follows that the weaker the drift to the diagonal, the larger the required value for T. In the extreme case with $p = 0.0$, in which the corresponding SQS-JDP consists of 2 independent $M|M|1$ queues, T has to be equal to 85 in order to reach the desired accuracy, while in the other extreme case with $p = 1.0$, in which we have a pure symmetric shortest queue system, T only has to be equal to 8.

From the values found for T, it may be concluded that the TKR and TBA model only lead to tight bounds, if the drift to the states with equal queue lengths is sufficiently strong. This will also hold for GSQSs with more than two servers. It is noted that the existence of a certain drift to the states with equal queue lengths has been a point of departure when we constructed the TKR and TBA model. So, if there is only a weak drift to the states with equal queue lengths, then the probability mass will not be concentrated around these states,

p	T	$W^{(A)}$	$\Delta^{(A)}(\hat{T})$	$W^{(B)}$	$\Delta^{(B)}(\hat{T})$	W	$\Delta(\hat{T})$
0.0	85	4.2648	0.0024	8.9976	0.0046	8.9976	0.0046
0.1	43	4.3594	0.0038	6.8002	0.0046	6.5561	0.0045
0.2	29	4.4027	0.0041	6.0435	0.0045	5.7154	0.0044
0.3	22	4.4266	0.0040	5.6619	0.0042	5.2913	0.0041
0.4	18	4.4414	0.0033	5.4320	0.0034	5.0357	0.0034
0.5	15	4.4516	0.0032	5.2782	0.0033	4.8649	0.0033
0.6	13	4.4589	0.0027	5.1682	0.0028	4.7426	0.0028
0.7	11	4.4645	0.0034	5.0856	0.0035	4.6509	0.0034
0.8	10	4.4688	0.0025	5.0212	0.0025	4.5793	0.0025
0.9	9	4.4722	0.0021	4.9697	0.0021	4.5220	0.0021
1.0	8	4.4751	0.0022	4.9275	0.0022	4.4751	0.0022

Table 2: The mean waiting times $W^{(\cdot)}$ and W determined within an absolute accuracy of $\epsilon_{abs} = 0.005$ for the GSQS with $\rho = 0.9$, $\lambda = 2\rho$, $\lambda_A = p\lambda$, $\lambda_B = \lambda_C = (1-p)\lambda$, and varying p.

and one should focus on bound models with alternative truncated state spaces.

The values presented in the Tables 1 and 2 for the mean waiting times itself, also deserve some attention. The results in Table 1 show that only a small difference between the waiting times for the types B and C and the waiting time for type A is obtained, even for high workloads. From the results in Table 2, it follows that the mean waiting time W for all job types together is more than proportionally decreasing as a function of the fraction p of jobs which can be served by both servers. In fact, a small fraction p of jobs that can be handled by both servers, already leads to a considerable reduction for W, compared to the situation with $p = 0$. From this, we can draw the following important conclusion for the production of Printed Circuit Boards by the flexible assembly system, as described in Section 1: *In order to obtain small mean waiting times for the given total workload, the assignment of the components to the insertion machines should be such that for the resulting GSQS a strong drift to the states with equal queue lengths is obtained.* Note that these assignments are precisely the ones for which our bound models work well. Hence, after having selected a small number of assignments which are expected to have the strong drift to the states with equal queue lengths, the bound models can be well used to compute the performance for each of the selected assignments, and subsequently the best assignment can be easily determined.

References

[1] ADAN, I.J.B.F., AND HOOGHIEMSTRA, G., The $M|M|c$ with critical jobs. Memorandum COSOR 96-20, Eindhoven University of Technology, Dept. of Math. and Comp. Sci., 1996.

[2] ADAN, I.J.B.F., VAN HOUTUM, G.J., AND VAN DER WAL, J. Upper and lower bounds for the waiting time in the symmetric shortest queue system. *Annals of Operations Research 48* (1994), 197–217.

[3] ADAN, I.J.B.F., WESSELS, J., AND ZIJM, W.H.M. Queueing analysis in a flexible assembly system with a job-dependent parallel structure. In *Operations Research Proceedings 1988*, Springer-Verlag, Berlin, 1989, pp. 551–558.

[4] ADAN, I.J.B.F., WESSELS, J., AND ZIJM, W.H.M. Analysis of the symmetric shortest queue problem. *Stochastic Models 6* (1990), 691–713.

[5] GREEN, L. A queueing system with general-use and limited-use servers. *Operations Research 33* (1985), 168–182.

[6] HASSIN, R., AND HAVIV, M. Equilibrium strategies and the value of information in a two line queueing system with threshold jockeying. *Stochastic Models 10* (1994), 415–435.

[7] HORDIJK, A., AND KOOLE, G. On the assignment of customers to parallel queues. *Probability in the Engineering and Informational Sciences 6* (1992), 495–511.

[8] NEUTS, M.F. *Matrix-Geometric Solutions in Stochastic Models.* Johns Hopkins University Press, Baltimore, 1981.

[9] ROQUE, D.R. A note on "Queueing models with lane selection". *Operations Research 28* (1980), 419–420.

[10] SCHWARTZ, B.L. Queueing models with lane selection: a new class of problems. *Operations Research 22* (1974), 331–339.

[11] VAN HOUTUM, G.J. *New Approaches for Multi-Dimensional Queueing Systems.* Ph.D. Thesis, Eindhoven University of Technology, Eindhoven, 1995.

[12] VAN HOUTUM, G.J., ADAN, I.J.B.F., AND VAN DER WAL, J. The symmetric longest queue system. *Stochastic Models 13* (1997), 105–120.

[13] ZIJM, W.H.M. Operational control of automated PCB assembly lines. In *Modern Production Concepts: Theory and Applications*, G. Fandel and G. Zaepfel, Eds. Springer-Verlag, Berlin, 1991, pp. 146–164.

Stochastic scheduling with event-based dynamic programming

Ger Koole
Faculty of Mathematics and Computer Science
Free Universiteit
E-mail: koole@cs.vu.nl

Abstract

In this paper event-based dynamic programming is applied to stochastic scheduling models. This allows us a unified treatment of many different models, among which are multiple and single server models (with and without feedback), discrete and continuous time models, models with controlled and uncontrolled arrivals, etc.

1 Introduction

Structural results for optimal policies for queuing models are usually derived in the following way. After having formulated the dynamic programming (dp) value function for a particular model, it is shown inductively that this value function satisfies certain properties, from which the results are derived. This value function represents either a discrete time model, or a continuous time model (through the well-known uniformization technique, first applied in Lippman [14]). In this paper we use event-based dynamic programming. Event-based deals with event operators, which can be seen as building blocks of the value function. Typically we associate an operator with each basic event in the system, such as an arrival at a queue, a service completion, etc. Event-based dp focuses on the underlying properties of the value and cost functions, and allows us to study many models at the same time. It is explained in the next section.

In this paper we apply event-based dp to stochastic scheduling problems. First we consider single and multiple server models without feedback. Results from Chang et al. [6], Hordijk & Koole [7], and Koole & Vrijenhoek [13] are generalized.

After that we study a single server model with feedback to the other queues. The continuous time version has already been studied in Nain [15] and Koole [10], the discrete time version is the subject of Weishaupt [20]. Their results are slightly generalized; in the cited papers only feedback to queues with a lower priority were allowed, we allow also feedback to the next higher queue. This model has the following interesting application. One of the problems of dynamic programming is the difficulty of working with general distributions. A possible solution is the use of phase-type distributions, as in Koole [10]. DFR and IFR distributions are characterized in terms of phase-type distributions, and using the feedback result limiting results for G/DFR/1 and G/IFR/1 queues are derived. The single class results can also be found in Righter & Shanthikumar [16].

2 Dynamic Programming

2.1 Event-based dynamic programming

In this section we formulate the dynamic programming value function in general terms and prove some theorems which form the basis of our method.

We take $x \in X = \mathbb{N}_0^{m+1}$ to be our state space. Define operators $T_0, \ldots, T_{k-1}$, as follows:

$$T_i(f_1, \ldots, f_{l_i})(x) = \min_{a \in A_i(x)} \Big\{ c_i(x,a) + \sum_{j=1}^{l_i} \sum_{y \in X} p_i^j(x,a,y) f_j(y) \Big\},$$

with $f_1, \ldots, f_{l_i} : X \to \mathbb{R}$. $A_i(x)$ is called the action set, $c_i(x,a)$ the direct costs and $p_i^j(x,a,y)$ the transition probabilities. We often take $l_i = 1$, as we will see in the next Sections. An important exception however is the uniformization operator for which we need $l_i > 1$. If $l_i = 1$ then T_i is the standard dp operator, given that $p^1(x,a,y) \geq 0$ for all x, a, y, and that $\sum_y p^1(x,a,y) = 1$ (or α, the discount factor) for each pair x and a. (In this case we omit the superscript of p.) In applications we choose the event operators as simple as possible, by associating one with every possible event in the system.

The value function V_{n+1} is constructed from V_n and the operators T_i as follows. Assume that V_0 is given. Define, for $n = 0, 1, \ldots$, $V_n^{(0)}, \ldots, V_n^{(k)}$ by taking $V_n^{(k)} = V_n$, for $j = 0, \ldots, k-1$ $V_n^{(j)} = T_j(V_n^{(k_1)}, \ldots, V_n^{(k_{l_j})})$, for some $j < k_1, \ldots, k_{l_j} \leq k$ (where the assumption that $k_1, \ldots, k_{l_s} > j$ is made to avoid circularity), and $V_{n+1} = V_n^{(0)}$.

Although this definition is notationally quite burdensome, the intuition is simple: each step of the dp consists of the parallel and/or consecutive execution of several events. If $l_i > 1$ for some i, then also the determination of which events are to be executed depends on the state, the realization, or the action. The central ideas are summarized in the following (trivial) theorem.

Theorem 2.1 *Let $\mathcal{F}$ be some class of functions from $\mathbb{N}_0^m$ to $\mathbb{R}$, and $V_0 \in \mathcal{F}$. If, for all i, for $f_1, \ldots, f_{l_i} \in \mathcal{F}$ holds that $T_i(f_1, \ldots, f_{l_i}) \in \mathcal{F}$ then $V_n \in \mathcal{F}$ for all n.*

In what follows we consider special event operators and show that $T_i f \in \mathcal{F}$ for all i. This proves that the value function $V_n \in \mathcal{F}$ for all models which can be constructed with the T_i. We choose $\mathcal{F}$ such that certain structural properties of the optimal control policies can be derived from it.

On the other hand, it is possible to show that V_n as defined can be rewritten in the standard MDP formulation, given by

$$W_{n+1}(x) = \min_{a \in A(x)} \Big\{ k(x,a) + \sum_y q(x,a,y) W_n(y) \Big\}.$$

This allows us to use techniques and results from the theory of MDP's.

Finally let us consider optimality criteria. Normally we assume that all $p_i^j(x,a,y) \geq 0$ and that for all i, x and a $\sum_{y,j} p_i^j(x,a,y) = 1$: then V_n represents the total minimal n-stage costs, and under certain conditions the policy minimizing V_n as $n \to \infty$ is average optimal. These conditions however are non-trivial and should be checked for each model separately, unless the state space is finite. If we make the exception that $\sum_{y,j} p_0^j(x,a,y) = \alpha$, then V_n converges (again, under certain conditions) to the minimal discounted costs. Our focus in this paper is on the properties of the value function, not on the existence of limiting policies. In the case of a finite state space existence is guaranteed for all models. For discounting

results from Schäl [17] often provide the necessary existence result; for average costs some useful conditions are summarized in Cavazos-Cadena & Sennott [5].

Other optimality criteria are also possible. For discrete-time models there are no problems to expect when minimizing total finite-stage costs. Other choices are also possible. Take for example $V_0 = C$ and no further costs in the definition of V_n, $n > 0$. Then V_n gives the minimal expected final costs after n time. Also for continuous time models we can consider the costs at say T. This can simply be done by conditioning on the number of jumps of the uniformization process (see Ch. 5 of Koole [9] or Koole & Liu [12]).

2.2 The value function

In this Subsection we present the operators.

Notationally we make use of the following conventions: e_i denotes the ith unity vector, $1 \leq i \leq m$, while e_0 is the 0 vector, each vector (in)equality is taken componentwise, $I\{\ldots\}$ is the indicator function, $x^+ = x$ if $x \geq 0$, 0 otherwise, increasing and decreasing are used in the non-strict sense.

The operators related to the service process are:

- $T_{SS}f(x) = \min_{0 \leq i \leq m}\{\mu(i)f(x-e_i)+(1-\mu(i))f(x)\}$, where the minimization ranges over those i with $x_i > 0$. This models a single server that services m parallel queues. Idleness, action 0, is always allowed.
- $T_{MS}f(x) = \min_{i_1,\ldots,i_s}\{\sum_{k=1}^{s}(\mu(i_k)f(x-e_{i_k})+(1-\mu(i_k))f(x))\}$ where $\sum_k I\{i_k = j\} \geq x_j$, i.e., no more servers can work on a queue than that there are customers in that queue. This models s parallel servers.
- $T_{SSFB}f(x) = \min_{1 \leq i \leq m}\{\sum_{k=0}^{m}\mu(i,k)f(x - e_i + e_k)\}$ if $x \neq 0$, $f(x)$ if $x = 0$. Action i models again serving queue i. This operator allows feedback to other queues. Note that we do not allow idleness if there are customers available. With $\mu(i,k)$ we denote the probability that a customer in queue i which is being served moves to queue k. Queue 0 means leaving the system. (Recall that e_0 was the 0 vector.) We assume that $\sum_{k=0}^{m}\mu(i,k) = 1$ for all i.

Operators related to arrivals are:

- $T_{A(i)}f(x) = f(x + e_i)$, $1 \leq i \leq m$. This operator models an arrival at queue i.
- $T_{FS}f(x) = \sum_i \lambda_i \min\{f((x + e_i), 1\}$, with $\sum_i \lambda_i = 1$, and the minimization componentwise. This models a finite source queue: with probability λ_i the single class i customer in the system arrives, assuming that it is not yet there.

The direct costs and the discounting are represented by the following operators:

- $T_{costs}f(x) = C(x) + \alpha f(x)$. Here α is the discount factor, thus we often take $\alpha \in (0, 1]$, with $\alpha = 1$ representing total costs, but we only need $\alpha \geq 0$. We often write $T_{costs}(C, f)$ to indicate that the conditions for f must also hold for C.

Remember that we had an $m + 1$ dimensional state space. Component 1 up to m are used for the queues, as we saw in the model above, the 0th state component will be used for the environment. This environment allows us to model general arrival streams, server vacations, etc. For examples see the following subsection.

- $T_{env(0)}(f_1, \ldots, f_l)(x) = \sum_{y \in \mathbb{N}_0}\lambda(x_i, y)\sum_{j=1}^{l} q^j(x_i, y)f_j(x^*)$, where x^* is equal to x with the 0th component replaced by y. This operator models Markov Arrival Processes, which are discussed in the next section.

A special case of $T_{env(0)}$ is T_{unif}, the uniformization operator, given by

- $T_{unif}(f_1, \ldots, f_l)(x) = \sum_j p(j)f_j(x)$ with $p(j) > 0$ for all j. This is a convex combination of the f_j. The value function of a continuous time model typically has this form, due to the uniformization. This technique was first introduced in Lippman [14], and further developed in Serfozo [19]. It is the basis of the analysis of most continuous-time Markovian models.

An interesting extension of $T_{env(0)}$ is

- $T_{Cenv(0)}(f_1, \ldots, f_l)(x) = \min_a \sum_{y \in \mathbb{N}_0} \lambda(x_i, a, y) \sum_{j=1}^{l} q^j(x_i, a, y) f_j(x^*)$. This operator allows for control in the environment. We will not go into details here, but we refer to the discussion of MDAP's in [7].

2.3 Examples of value functions

Here we give some examples of value functions. A simple model with arrivals and service independent of some environment state (i.e., Poisson arrivals and constant service rates) has value function

$$V_{n+1} = T_{costs}(C, T_{unif}[T_{A(1)}V_n, \ldots, T_{A(m)}V_n, T_{SS}V_n]).$$

We could let the arrivals (and also the server completion times) depend on some environment state. This can be done with a Markov Arrival Process, which has the property that the class of all MAP's is dense in the class of all arrival processes (Asmussen & Koole [1]). An MAP consists of a Markov process on the environment states (with transition rates $\lambda(x, y)$), and event probabilities: if the environment moves from x to y then with probability $q^j(x, y)$ an event of type j (which can be an arrival in a certain class, or a possible service completion) occurs. Uniformization of such an MAP leads in a continuous time setting to a value function of the form

$$V_{n+1} = T_{costs}(C, T_{env(0)}[T_{A(1)}V_n, \ldots, T_{A(m)}V_n, T_{SS}V_n]),$$

where the MDAP is modeled by the operator $T_{env(0)}$. This is the model of Buyukkoc et al. [4], for which they show the optimality of the μc rule. If we replace T_{SS} by T_{SSFB} then we get the model of Section 3 of Nain [15]. If we take T_{MS} then we find the model of Chang et al. [6].

Also discrete time models can be modeled with event-based dynamic programming. The crucial property of discrete time models is that events occur after each other. Choices can be made here; we give below the value function of the model of Weishaupt [20]. It is given by:

$$V_{n+1} = T_{SSFB}(T_{costs}[C, T_{unif}(T_{A(1)}^{b(1,1)} \cdots T_{A(m)}^{b(1,m)} V_n, \ldots, T_{A(1)}^{b(l,1)} \cdots T_{A(m)}^{b(l,m)} V_n)]).$$

The uniformization operator represents the arrivals. Thus with probability $p(j)$ a batch of customers arrives, with $b(j, i)$ arrivals in queue i.

3 Models without Feedback

In this section we study models with service operators T_{SS} or T_{MS}. We define the class of functions $\mathcal{F}$ as follows. $f \in \mathcal{F}$ if the following two inequalities hold:

$$\mu(i)f(x - e_i) + (1 - \mu(i))f(x) \leq \mu(j)f(x - e_j) + (1 - \mu(j))f(x)$$

for all x such that $x_i > 0$ and $x_j > 0$, $i < j$, and

$$f(x) \leq f(x + e_i)$$

for all x and i.

The first inequality has a simple interpretation. The terms involved can be found in T_{SS}. Thus if $f \in \mathcal{F}$, then the minimizing action for $T_{SS}f$ consists of serving the customer of the lowest class number available, which is called the Smallest Index Policy (SIP) in [7]. The

second inequality shows that idling is not optimal. It is also readily seen that for T_{MS} the servers should be assigned to the group of s servers with the lowest indices. The following lemma is the basis for our results, it shows under what conditions $V_n \in \mathcal{G}$.

Lemma 3.1 *The following hold:*

$$f \in \mathcal{F} \Longrightarrow T_{SS}f \in \mathcal{F},\ T_{A(i)}f \in \mathcal{F},$$

$$f \in \mathcal{F},\ \mu_1 \leq \cdots \leq \mu_m \Longrightarrow T_{MS}f \in \mathcal{F},$$

$$f \in \mathcal{F},\ \lambda_1 \leq \cdots \leq \lambda_m \Longrightarrow T_{FS}f \in \mathcal{F},$$

$$C, f \in \mathcal{F} \Longrightarrow T_{costs}(C, f) \in \mathcal{F},$$

$$f_1, \ldots, f_l \in \mathcal{F} \Longrightarrow T_{env(0)}(f_1, \ldots, f_l) \in \mathcal{F},$$

$$f_1, \ldots, f_l \in \mathcal{F},\ \mu_1 \leq \cdots \leq \mu_m \Longrightarrow T_{Cenv(0)}(f_1, \ldots, f_l) \in \mathcal{F}.$$

Proof The proof follows by induction, following arguments in [11], [7] and [13]. □

Our results are summarized in the following theorem.

Theorem 3.2 *For value functions consisting of the operators $T_{A(1)}, \ldots, T_{A(m)}$, T_{FS} (with $\lambda_1 \leq \cdots \leq \lambda_m$), T_{SS}, T_{MS} (with $\mu_1 \leq \cdots \leq \mu_m$), T_{costs}, $T_{env(0)}$ and/or $T_{Cenv(0)}$ (with $\lambda_1 \leq \cdots \leq \lambda_m$), the SIP is optimal if $C \in \mathcal{F}$.*

An extensive study of allowable cost functions can be found in [7]. One of the results is that for C of the form $C(x) = \sum_i c_i x_i$, $C \in \mathcal{F}$ is equivalent to $c_i \geq 0$ and $\mu_1 c_1 \geq \cdots \geq \mu_m c_m$. Thus the SIP serves in decreasing order of $\mu_i c_i$. This is called the μc rule. If $\mu_1 \leq \cdots \leq \mu_m$ then the SIP serves the customers with the least expected processing times, which is called LEPT.

In the following corollary we summarize the results for the main models.

Corollary 3.3 *In the model with:*
independent arrivals and a single server the μc rule is optimal;
controlled arrivals and multiple servers the μc rule is optimal if it coincides with LEPT;
a finite source and a single server the μc rule is optimal if $\lambda_1 \leq \cdots \leq \lambda_m$;
a finite source and multiple servers the μc rule is optimal if it coincides with LEPT and if $\lambda_1 \leq \cdots \leq \lambda_m$.

The result for the first model can also be found in Buyukkoc et al. [4] or Baras et al. [2]. The second result can be found in Hordijk & Koole [7], the result for uncontrolled arrivals can also be found in Chang et al. [6]. The third result is that of Koole & Vrijenhoek [13], the fourth is a generalization of it.

4 Single-Server Model with Feedback

We continue with the inequality that we consider for the operator T_{SSFB}. We define the class of functions $\mathcal{G}$ as follows:

$$f \in \mathcal{G} \Longleftrightarrow \sum_k \mu(i,k) f(x - e_i + e_k) \leq \sum_k \mu(j,k) f(x - e_j + e_k),$$

for all x such that $x_i > 0$, $x_j > 0$, and $1 \leq i < j \leq m$.

The terms in the inequality can be found in T_{SSFB}. Thus if $f \in \mathcal{G}$, then the minimizing action for $T_{SSFB}f$ consists again of serving the customer of the lowest class number available, the SIP. The following lemma is the basis for our results, it shows under what conditions $V_n \in \mathcal{G}$.

Lemma 4.1 *If, in the definition of T_{SSFB}, μ is such that, for $1 \leq i \leq m$, $\mu(i,k) = 0$ for $0 < k < i-1$, then the following hold:*

$$f, C \in \mathcal{G} \Longrightarrow T_{A(i)}f \in \mathcal{G},\ T_{SSFB}f \in \mathcal{G},\ T_{costs}(C,f) \in \mathcal{G},\ T_{env(0)}f \in \mathcal{G}.$$

Proof The proof follows by induction, following arguments in [11] and [10]. We pay attention to the inductive step for the operator T_{SSFB} as existing results are extended for this operator.

We have to show, given that $f \in \mathcal{G}$, that

$$\sum_k \mu(i,k) T_{SSFB}f(x - e_i + e_k) \leq \sum_k \mu(j,k) T_{SSFB}f(x - e_j + e_k).$$

Consider the minimizing actions at the right hand side. For each of the states $x - e_j + e_k$ there are two possibilities: either action i is optimal in each state (note that $x_i > 0$), or there is a $l < i$ with $x_l > 0$ optimal. Note that because of our conditions on μ we have that $k \geq j$. For the rest of the proof we refer to [10]. □

Our result can be summarized as follows.

Theorem 4.2 *For value functions consisting of the operators $T_{A(1)}, \ldots, T_{A(m)}$, T_{SSFB}, T_{costs} and/or $T_{env(0)}$, the SIP is optimal under the following conditions:*
(i) Feedback in T_{SSFB} to queues with a lower index number should be restricted to the next higher queue;
(ii) The direct costs C should be such that $C \in \mathcal{G}$.

Let us compare this result with those obtained in the literature. As we saw already in the previous section we can deal at the same time with the continuous time model of Nain [15] (in [10] an equivalent result is obtained) and the discrete time model of Weishaupt [20]. Compared to their results there is a second difference: we allow not only feedback to lower indexed queues, but also to the next higher queue. This small difference will allow us in the next section to extend substantially the limiting results of Koole [10] for G/IFR/1 and G/DFR/1 queues.

Finally let us look at some special cases of the theorem. First assume that only $\mu(i,0)$ and $\mu(i,i)$ are non-zero, and that C has the following special form: $C(x) = \sum_i c_i x_i$. Then $C \in \mathcal{G}$ is equivalent to:

$$\mu(i,0)C(x - e_i) + (1 - \mu(i,0))C(x) \leq \mu(j,0)C(x - e_j) + (1 - \mu(j,0))C(x),$$

which is, due to the special form of C, equivalent to $\mu(i,0)c_i \geq \mu(j,0)c_j$. Thus we find again the well known μc rule.

Now consider cost functions which are only functions of the total number of customers in the system. Obvious functions are $C(x) = |x|$ or $-|x|$, corresponding respectively to minimizing and maximizing the number of customers in the system, but also $C(x) = I\{|x| > s\}$ and $I\{|x| < s\}$ are of interest: they correspond to the probability that there are more or less than s customers in the system. Thus for increasing cost functions (such as $|x|$ and $I\{|x| > s\}$) $C \in \mathcal{G}$ is equivalent to $\mu(i,0)$ decreasing in i, and vice versa. We will use this in the next section.

5 Phase-Type Distributions and Limit Results

Based on results in Schassberger [18] it is shown in [10] that any distribution function F can be approximated with distributions F_n of the form

$$F_n(x) = \sum_{k=1}^{\infty} \beta_k E_n^k(x),$$

with $\beta_1 = F(1/n)$ and $\beta_k = F(k/n) - F((k-1)/n)$ for $k > 1$. E_n^k is the density function of a gamma distributed r.v. with k phases and intensity n.

Thus β_k is the probability of exactly k phases. Another way to construct this r.v. is as follows. Define

$$\alpha_m = \begin{cases} \frac{\beta_m}{1-\sum_{k=1}^{m-1}\beta_k} & \text{if } \sum_{k=1}^{m-1}\beta_k < 1, \\ 1 & \text{if } \sum_{k=1}^{m-1}\beta_k = 1. \end{cases}$$

Let the r.v. G_n be the time until absorption of the following Markov process. The initial state is 1. The process rests in each state k an E_n^1 (exponentially) distributed amount of time, after that absorption takes place with probability α_n, or the system moves to state $k+1$ with probability $1-\alpha_n$. This is a special case of a Cox distribution. Because $\beta_m = (1-\alpha_1)\cdots(1-\alpha_{m-1})\alpha_m$ we see that $F_n \stackrel{\mathrm{d}}{=} G_n$.

Now we discuss DFR and IFR distributions, and their relation to phase-type distributions of the form G_n. We use the following definition of Barlow & Prochan [3], which is only in terms of $\bar{F}(t) = 1 - F(t)$ (thus the failure rate itself does not need to exist).

Definition 5.1 *(DFR and IFR) A non-negative distribution function is:*
DFR if $\bar{F}(t+s)/\bar{F}(t)$ is increasing in $t \geq 0$ with $\bar{F}(t) > 0$, for each $s \geq 0$;
IFR if $\bar{F}(t+s)/\bar{F}(t)$ is decreasing in $-\infty < t < \infty$ with $\bar{F}(t) > 0$, for each $s \geq 0$.

Examples of IFR (DFR) distributions are distribution with a non-decreasing (non-increasing) failure rate (defined by $f(t)/1 - F(t)$, with f the density), assuming that it exist for all t. But also $F(t) = I\{t \geq x\}$, the deterministic distribution, is IFR, although its failure rate does not exist.

Now we can formulate the result on phase-type distributions:

Theorem 5.2 *If F is DFR (IFR) then α_m is decreasing (increasing) in m, for all n.*

The proof of this theorem can be found in [10]. Hordijk & Ridder [8] proved the DFR part of this theorem for general Cox distributions.

A disadvantage of this method is that we need an infinite number of states. We can make the state space finite by changing the approximation into

$$F_n(x) = \sum_{k=1}^{n^2} \beta_k E_n^k(x) + (1 - \sum_{k=1}^{n^2} \beta_k) \sum_{k=1}^{\infty} (1-\beta_{n^2})^{k-1} \beta_{n^2} E_n^{n^2+k}(x).$$

It is easily checked that the approximation result also holds for this F_n, and that α_k becomes constant from $k = n^2$ on, so that we only need a finite number of states for the representation with the α's.

Now assume that we have a single class of customers, and identify the queues of theorem 4.2 with the stages of the phase-type distribution. We let customers arrive in queue 1, and we take $\mu(i,0) = \alpha_i$ and $\mu(i,i+1) = 1-\alpha_i$, $i < m$, and for $m = n^2$ we take $\mu(i,i) = 1-\alpha_i$,

using the finite state approximation given above. At the end of the last section it was shown that if $C(x) = I\{|x| > s\}$, then $\mu(i,0) = \alpha_i$ should be decreasing in i, i.e., F_n is DFR. Thus to minimize the number of customers, all having the same DFR distribution, the customers with the least attained service times (LAST) should be served. Similarly, to maximize the number of customers when they have IFR distributions LAST should also be used. Note that LAST, in the limit, leads to processor sharing between all the customers which have the same minimal attained service time.

Another possibility is arrivals at queue $m = n^2$, with $\mu(i,0) = \alpha_{n^2+1-i}$ and $\mu(i,i-1) = 1 - \alpha_{n^2+1-i}$, $i > 1$, and $\mu(1,1) = 1 - \alpha_{n^2}$. The optimal policy serves the customer which has the most attained service time (MAST). Note that this is equivalent to non-preemptive service.

All these results hold for all F_n, and therefore also for their limit F. This gives the following corollary. These results can also be found in Righter & Shanthikumar [16].

Corollary 5.3 *The number of customers at any time T in a G/G/1 queue is:*
minimized (maximized) by LAST in case of DFR (IFR) service times;
minimized (maximized) by MAST in case of IFR (DFR) service times.

Generalizations can be obtained to different customer classes having different service times. As long as feedback occurs to lower numbered queues no problems are to be expected. This is the case for DFR (IFR) distributions with positive (negative) holding times; see [10] for details. For the other two cases the situation is less clear. As different queues can belong to different customer classes, we have no guarantee that a change in phase means a transition to the next higher queue. For each case separately the conditions should be checked.

References

[1] S. Asmussen and G.M. Koole. Marked point processes as limits of Markovian arrival streams. *Journal of Applied Probability*, 30:365–372, 1993.

[2] J.S. Baras, D.-J. Ma, and A.M. Makowski. K competing queues with geometric service requirements and linear costs: The μc-rule is always optimal. *Systems and Control Letters*, 6:173–180, 1985.

[3] R.E. Barlow and F. Proschan. *Statistical Theory of Reliability and Life Testing.* Holt, Rinehart and Winston, 1975.

[4] C. Buyukkoc, P. Varaiya, and J. Walrand. The $c\mu$ rule revisited. *Advances in Applied Probability*, 17:237–238, 1985.

[5] R. Cavazos-Cadena and L.I. Sennott. Comparing recent assumptions for the existence of average optimal stationary policies. *Operations Research Letters*, 11:33–37, 1992.

[6] C.S. Chang, X. Chao, M. Pinedo, and R.R. Weber. On the optimality of LEPT and $c\mu$-rules for machines in parallel. *Journal of Applied Probability*, 29:667–681, 1992.

[7] A. Hordijk and G.M. Koole. On the optimality of LEPT and μc rules for parallel processors and dependent arrival processes. *Advances in Applied Probability*, 25:979–996, 1993.

[8] A. Hordijk and A. Ridder. Stochastic inequalities for an overflow model. *Journal of Applied Probability*, 24:696–708, 1987.

[9] G.M. Koole. *Stochastic Scheduling and Dynamic Programming.* PhD thesis, Leiden University, 1992. On request available from the author.

[10] G.M. Koole. Optimal server assignment in the case of service times with monotone failure rates. *Systems and Control Letters*, 20:233–238, 1993.

[11] G.M. Koole. Structural results for the control of queueing systems using event-based dynamic programming. Technical Report WS-461, Vrije Universiteit Amsterdam, 1996.

[12] G.M. Koole and Z. Liu. Stochastic bounds for queueing systems with multiple on-off sources. Technical Report WS-471, Vrije Universiteit Amsterdam, 1997.

[13] G.M. Koole and M. Vrijenhoek. Scheduling a repairman in a finite source system. *ZOR - Mathematical Methods of Operations Research*, 44:333–344, 1996.

[14] S.A. Lippman. Applying a new device in the optimization of exponential queueing systems. *Operations Research*, 23:687–710, 1975.

[15] P. Nain. Interchange arguments for classical scheduling problems in queues. *Systems and Control Letters*, 12:177–184, 1989.

[16] R. Righter and J.G. Shanthikumar. Scheduling multiclass single server queueing systems to stochastically maximize the number of successful departures. *Probability in the Engineering and Informational Sciences*, 3:323–333, 1989.

[17] M. Schäl. Conditions for optimality in dynamic programming and for the limit of n-stage optimal policies to be optimal. *ZOR - Mathematical Methods of Operations Research*, 32:179–196, 1975.

[18] R. Schassberger. *Warteschlangen.* Springer-Verlag, 1973.

[19] R.F. Serfozo. An equivalence between continuous and discrete time Markov decision processes. *Operations Research*, 27:616–620, 1979.

[20] J. Weishaupt. Optimal myopic policies and index policies for stochastic scheduling problems. *ZOR - Mathematical Methods of Operations Research*, 40:75–89, 1994.

Performance of queues with 'worst case input'

Michel Mandjes*
KPN Research
E-mail: m.r.h.mandjes@research.kpn.nl

Abstract

This paper focuses on the performance evaluation of queues fed by the worst case traffic that can leave the policer. The existing literature is reviewed, and an easily computable upper bound on the loss probability is derived. Some numerical examples are provided in order to show the strength of this bound.

1 Introduction

In recent years, telecommunication networks that use the asynchronous transfer mode (ATM) have appeared to be an emergent technology. In ATM systems, contracts are negotiated between network and customer. In such a contract the network promises a certain Quality of Service (loss, delay), as long as the customer satisfies a number of traffic characteristics. In order to enforce that the traffic entering the network meets these requirements, so-called Usage Parameter Control ('policing') is performed on the edge of the network. A (stylized version of a) typical contract of customer i consists of on the one hand a maximum allowed loss fraction ϵ, on the other hand peak p_i and mean rate r_i of the regulated traffic, as as well as the maximum allowed burst period T_i.

One node in the ATM network can be seen as a queue fed by a number of sources, all of them characterized by their 'triple' (r_i, p_i, T_i). We assume that the system is characterized by a (constant) link rate C and a buffer of size B.

It is generally believed that traffic with deterministic on-periods (or bursts, with length T_i) and off-periods, peak rate p_i and mean rate r_i is the worst case traffic that can depart from the ith UPC regulator, allthough not formally proven [2]. With 'worst case' we mean that it essentially maximizes the loss probability, given the parameters (r_i, p_i, T_i). From this point of view, it seems logical to base the Call Acceptance Control routine (i.e., the check whether or not a new request can be admitted without violating the loss constraint) on customers supplying this kind of traffic.

In view of the above, it is clear that it is essential to have a method for performance evaluation of a queue fed by 'worst case input'. Notice that the only stochastic effect is the 'random start' of the on-period within the period of the source. However, the model has appeared to be notoriously hard to analyze. A number of procedures have already been

*The author was PhD student at the Vrije Universiteit Amsterdam, during the period 1993-1996, where he was supervised by Ad Ridder and Henk Tijms. He followed courses and attended workshops organized by LNMB. He currently works at KPN Research, Leidschendam. This work considerably benefited from discussions with Maurits de Graaf and Hans van den Berg. Pim van der Stoel was helpful in providing numerical results.

proposed in literature, but they all have their particular deficiencies. The contribution of this paper is to develop a technique that copes with all these drawbacks.

The structure of this paper is as follows. First we evaluate a number of methods that are known from literature. Then we found a general, uniform, easily computable bound. We end up by giving some numerical results.

2 Review of proposed methods

This section reviews and evaluates the existing literature in the field of queues fed by 'worst case traffic', and states the properties which a good approximation or bound must satisfy.

Kvols and Blaabjerg

Kvols and Blaabjerg [4] consider the discrete-time version of the model, where the sources are homogeneous. Then the model is equivalently given, by on-time T, off-time T' and cell spacing D (i.e., during bursts, every D time units a cell arrives; in other words: '$D = p^{-1}$'). The well-known Beneš formula [5] then states (assuming that at time t stationary behavior is reached)

$$\mathrm{P}(Q_t > r) = \sum_{s=1}^{\rho(T+T')-r} \mathrm{P}(N(t-s,t) = r+s)\,\mathrm{P}(Q_{t-s} = 0 | N(t-s,t) = r+s).$$

Here ρ is the load of the multiplexer $(N/D) \times T(T+T')^{-1}$, $N(s,t)$ is the number of arrivals from all N sources in time interval $[s,t]$. It is immediately seen that the following upper bound holds:

$$\mathrm{P}(Q_t > r) \leq \sum_{s=1}^{\rho(T+T')-r} \mathrm{P}(N(t-s,t) = r+s).$$

Kvols and Blaabjerg numerically show that this upper bound is not very accurate. Apart from the upper bound, also an approximation is developed:

$$\mathrm{P}(Q_t > r) \approx \sum_{s=1}^{\rho(T+T')-r} \left\{\mathrm{P}(N(t-s,t) = r+s) - \rho \mathrm{P}(N(t-s,t) = r+s | X = 1)\right\},$$

where X is the stochastic number of cells arriving in interval $[t-s-1, t-s]$. In order to execute this approximation, some $N-$fold convolutions have to be calculated.

The drawbacks of the method presented by Kvols and Blaabjerg are the following: (i) only one type of sources is involved, (ii) the method seems to be computationally not very easy (convolutions have to be determined), (iii) it is not clear in what region the approximation is accurate (for large N for instance?), (iv) only the exceedance probability for infinite buffer model is examined, instead of the – for telecommunications more relevant – cell loss ratio in the finite buffer model.

Garcia, Barceló, and Casals

Garcia, Barceló, and Casals [3] find an algorithm that gives an upper bound for the survivor probability, above refered to as $\mathrm{P}(Q_t > r)$. As Kvols and Blaabjerg [4] did, the starting point of the analysis is the Beneš result. However, now the continuous-time (i.e., *fluid*) version of the model is considered. In fact, Garcia, Barceló, and Casals come up with a numerical

scheme to calculate the (continuous-time equivalent of the) upper bound found by Kvols and Blaabjerg (see above).

The method has a number of disadvantages: (i) only one type of traffic is assumed, (ii) the calculation scheme is computationally demanding with many complicated recursive steps to be made, (iii) it provides only an upper bound, without any information about the error incurred, (iv) it assumes that the on-period is smaller than the off-period: $T < T'$.

Botvich and Duffield

As can be learnt from the above studies, it seems to be impossible to find closed-form expressions for the relevant performance measures; ingeneous numerical schemes providing approximations is the best that is achieved up to now. An other angle from which the problem can be considered is the development of asymptotic analysis. One key paper in this field is by Botvich and Duffield [1].

Assume without loss of generality that there are two types of sources; let the total number of sources in the model be N, where a fraction $a \in [0,1]$ is of type 1; $N_1 := aN$ and $N_2 := (1-a)N$. Then they scale B and C by the number of sources N to Nb and Nc. Then Large Deviations analysis can be performed in order to get the asymptotic relation

$$\lim_{N\to\infty} \frac{1}{N} \log Q(N,B,C) = -I(b,c),$$

where $Q(N,B,C)$ is the exceedance probability of level B.

The way $I(b,c)$ has to be calculated is the following. $A_i(t) = pX_i(t)$ is the fluid generated by one single source of type i during $[0,t]$, where $X_i(t)$ is the on-time in this interval $[0,t]$. Furthermore:

$$M_{t,i}(\theta) = \mathrm{E}\exp(\theta A_i(t)) \quad \text{and} \quad J(t,x) := \sup_{\theta}[\theta x - a\log M_{t,1}(\theta) - (1-a)\log M_{t,2}(\theta)].$$

Then it can be shown that

$$\lim_{N\to\infty} \frac{1}{N} \log Q(N,B,C) = -\sup_{t\geq 0} J(t, b+ct) =: -I(b,c).$$

It should be noted that this method is not only appropriate for deterministic on-off sources, but under much more general conditions, see the hypothesis mentioned in [1].

The distribution of $X(t)$ can be given as follows. First notice that the off-period is given by $T' := (p-r)r^{-1}T$. Here we assume that $T < T'$, but an analogous reasoning applies to the case that $T > T'$. Three cases can be distinguished:

- $t < T$.

$$X(t) \stackrel{d}{=} \begin{cases} t-U & U \in [0,t] \\ 0 & U \in [t,T'] \\ U-T' & U \in [T',t+T'] \\ t & U \in [t+T',T+T'] \end{cases}$$

- $T \leq t < T'$.

$$X(t) \stackrel{d}{=} \begin{cases} T & U \in [0,t-T] \\ t-U & U \in [t-T,t] \\ 0 & U \in [t,T'] \\ U-T' & U \in [T',T+T'] \end{cases}$$

- $t \geq T'$.

$$X(t) \stackrel{d}{=} \begin{cases} T & U \in [0, t-T] \\ t-U & U \in [t-T, T'] \\ t-T' & U \in [T', t] \\ U-T' & U \in [t, T+T'] \end{cases}$$

Here U is uniformly distributed on $[0, T+T']$. This enables the calculation of the moment generating functions $M_{t,i}(\cdot)$ in a straightforward way.

This method has disadvantages as well: (i) the result obtained is asymptotic in N (consequently, it is not clear whether they hold for relatively small N), (ii) the method only yields the asymptotic decay rate of the exceedance probability, (iii) there are no results for the loss fraction, allthough it is likely that the same asymptotic result also holds for this performance measure. Advantages are that the calculations are not very complex and are valid for multiple types of traffic.

Simonian and Guibert

Simonian and Guibert [6] consider the same asymptotic regime as Botvich and Duffield [1], but they consider the case of one type of sources. Besides the establishment of the above asymptotic relation, they find an asymptotic upper bound. This asymptotic bound is again based on the Beneš result, as follows.

The probability of interest $Q(N, B, C)$ is bounded by some probability $Q^*(N, B, C)$ (which Garcia, Barceló, and Casals approximate). Then it is shown that

$$Q^*(N, B, C)\sqrt{N}\exp[NI(b, c)] \to d \text{ as } N \to \infty,$$

for some computable number d. The drawbacks of this method are the following: (i) a single class is considered, where in practice the multiple types of traffic case is more relevant, (ii) the method gives the asymptote of an upper bound, which does not mean that

$$Q(N, B, C) \leq \frac{d\exp[-NI(b, c)]}{\sqrt{N}}$$

uniformly in N, (iii) again, no loss rates but exceedance probablities are examined.

Elwalid, Mitra, and Wentworth

In Elwalid, Mitra, and Wentworth [2], as a first step, a worst-case analysis of this model is treated: they assume that all bursts arrive simultaneously. Then it is easy to verify that if the solution of the following linear program has a solution smaller than C there cannot be any loss:

$$\min \sum_{i=1}^{N} c_i, \quad \text{subject to } \sum_{i=1}^{N} (p_i - c_i)T_i \leq B,$$

where $r_i \leq c_i \leq p_i$. The correctness of this statement is an immediate consequence of the implication

$$\left\{\sum_{i=1}^{N} c_i \leq C\right\} \quad \Rightarrow \quad \left\{\text{Buffer contents} \leq \sum_{i=1}^{N} (p_i - c_i)T_i\right\} \tag{1}$$

Elwalid *et al.* [2] then assume an additional requirement, namely that the ratio of the contribution to the buffer contents by a particular source ($b_i = (p_i - c_i)T_i$) and its bandwidth

consumption (c_i) is equal to B/C. Then the problem can be solved explicitly, and yields that there is no loss if the sum of the effective bandwidths is not larger than C, where the effective bandwidth c_i of a source with parameters (r_i, p_i, T_i) is given by

$$c_i := \max\left(r_i, p_i\left(1+\frac{B}{CT_i}\right)^{-1}\right).$$

This is, since its solution satisfies the extra proportionality constraint, an underestimate of the real acceptance region.

In practice, it is not required that there is zero loss, but some small loss fraction ϵ is allowed, typically between 10^{-6} and 10^{-9}. In order to allow for such a small loss probability, Elwalid *et al.* [2] propose a statistical multiplexing routine. First they approximate the loss probability in the model with buffer B, by the probability of the buffer contents exceeding B in the infinite buffer model. Then they show that the a source of type i requires c_i only a fraction r_i/c_i of time. So, at a random moment in time the required bandwidth by a type i source amounts to c_i with probability r_i/c_i and is zero else. Reasoning in this way, they are able to calculate the probability of the required bandwidth exceeding link rate C; this probability is – consequently – an upper bound for the probability of the queueing model exceeding level B.

The great advantages of this method are its intuitive nature, its low complexity, and its applicability for the case of multiple types of traffic. However, (i) the model sometimes gives a very inaccurate upper bound – orders of magnitude larger than the real probability, as we will see in section 4, (ii) again, the buffer level exceedance probability is considered instead of the loss rate.

3 The upper bound

In this section we shall derive an easily computable upper bound to the loss rate. We first examine the homogeneous case: there a cycle has length $T+T'$. Consider the probability of loss during a cycle $\alpha(B)$. Then there must be an s, t (with $s < t < s+T+T'$) such that

$$\sum_{i=1}^{N}(A_i(t)-A_i(s)) \geq C(t-s)+B.$$

Without loss of generality, we can assume that $s=0$. Notice that if there is a $t \in [0, T+T']$ with $\sum A_i(t) \geq N(ct+b)$, there must also be an epoch of a *burst end* S_j (with $j=1,\ldots,N$) such that $\sum A_i(S_j) \geq N(cS_j+b)$. More formally:

$$\begin{aligned}
\alpha(B) &= \mathrm{P}\left(\exists S_j, j=1,\ldots N, \sum_{i=1}^{N} A_i(S_j) \geq N(cS_j+b)\right) \\
&\leq \sum_{j=1}^{N} \mathrm{P}\left(\sum_{i=1}^{N} A_i(S_j) \geq N(cS_j+b)\right) \\
&\leq N \max_{j=1,\ldots,N} \mathrm{P}\left(\sum_{i=1}^{N} A_i(S_j) \geq N(cS_j+b)\right) \\
&\leq N \max_{t\in[0,T+T']} \mathrm{P}\left(\sum_{i=1}^{N} A_i(t) \geq N(ct+b)\right).
\end{aligned}$$

For all positive θ we have, with the Markov inequality,

$$\mathrm{P}\left(\sum_{i=1}^{N} A_i(t) \geq N(ct+b)\right) \leq \left(\frac{M_t(\theta)}{\exp[\theta(ct+b)]}\right)^N.$$

But since this holds for all $\theta > 0$, we can also take the infimum over θ in the right hand side. We finally arrive at

$$\begin{aligned}\alpha(B) &\leq N \max_{t\in[0,T+T']} \inf_{\theta>0} \exp\left[-N(\theta(ct+b) - \log M_t(\theta))\right] \\ &= N \exp\left[-N \inf_{t\in[0,T+T']} \sup_{\theta>0} (\theta(ct+b) - \log M_t(\theta))\right].\end{aligned}$$

Now consider the heterogeneous case. Then it can be verified easily that an analogous reasoning eventually yields

$$\alpha(B) \leq f(N) \exp\left[-N \inf_{t\in[0,T+T']} \sup_{\theta>0} (\theta(ct+b) - a\log M_{t,1}(\theta) - (1-a)\log M_{t,2}(\theta))\right],$$

where $f(N)$ is the number of bursts arriving during a cycle of this queue. The calculation of $f(N)$ involves an easy calculation with an lowest common divisor.

Now the fraction $\eta(B)$ of fluid lost (during a cycle) can be bounded as follows. With 'renewal reward':

$$\eta(B) = \frac{\text{E(fluid lost during a cycle)}}{\text{E(fluid arrived during a cycle)}};$$

conditioning yields:

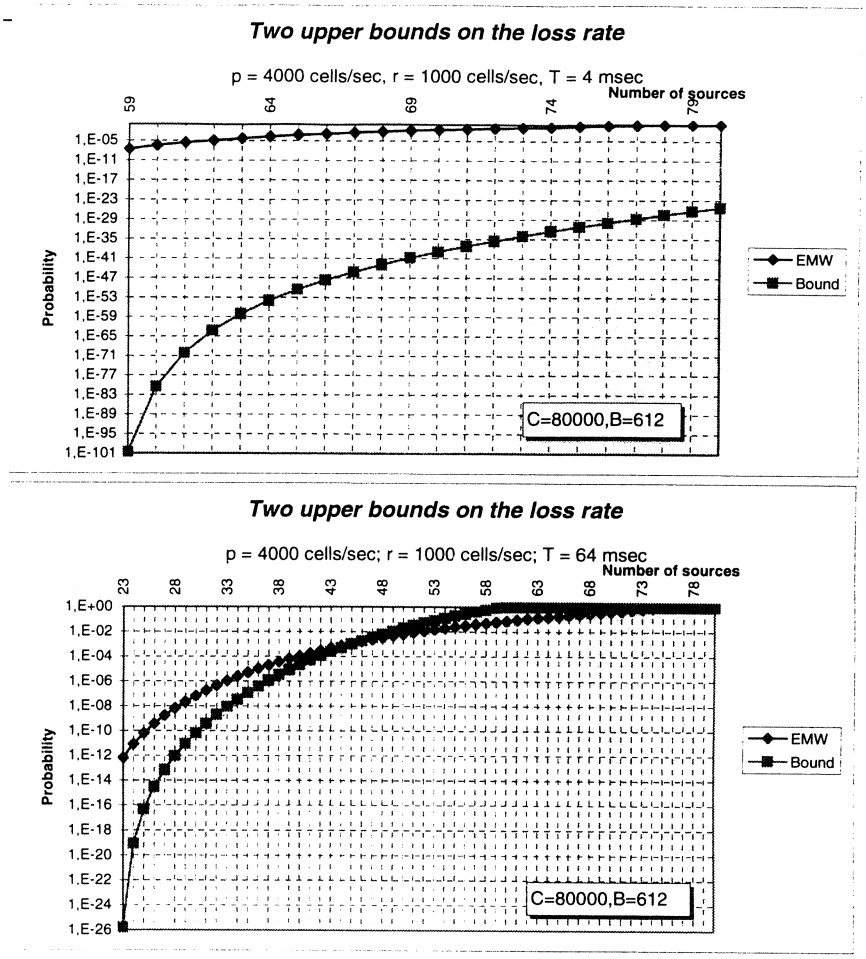
$$\begin{aligned}\text{E(fluid lost during a cycle)} &= \text{E(fluid lost during an overflow cycle)}\alpha(B) \\ &\leq \text{E(fluid arrived during a cycle)}\alpha(B),\end{aligned}$$

implying that $\eta(B) \leq \alpha(B)$. We already found an upper bound to $\alpha(B)$, so this bound also applies to the loss rate $\eta(B)$.

4 Numerical results

In this section we consider the following examples. In both cases $r = 1000$ cells/second, $p = 4000$ cells/second. In the first example however, $T = 4$ ms, whereas in the second $T = 64$ ms is used. Notice that we in fact consider two homogeneous cases.

We compare the results of our method (and particularly the number of sources that can be connected in order to guarantee a loss rate of 10^{-6}) with the results that are provided by the method that is most commonly used, namely the computation scheme provided by [2]. The figures display the loss probability as a function of the number of sources, using the 'EMW-method' and using the upper bound of section 3. The most striking feature is that the difference in sources that can be admitted. In the first example, with target loss 10^{-6}, 60 sources can be accepted according to EMW, where even with 80 sources (which is of course the maximum number, due to the stability requirement) the loss rate is still in the order of 10^{-25}. The other example is less extreme: 37 instead of 33 can be accepted.



References

[1] D. Botvich and N. Duffield [1994]. Large deviations, the shape of the loss curve and economies of scale in large multiplexers. *Queueing Systems,* 293-320.

[2] A. Elwalid, D. Mitra, and R. Wentworth [1995]. A new approach for allocating buffers and bandwidth to heterogeneous, regulated traffic in an ATM Node. *IEEE Journal on Selected Areas in Communications,* 1115-1127.

[3] J. Garcia, J. Barceló, and O. Casals [1995]. An exact model for the multiplexing of worst case traffic sources. *Sixth International Conference on Data Communication Systems and their Performance.*

[4] K. Kvols and S. Blaabjerg [1992]. Bounds and approximations for the periodic on-off queue with application to ATM traffic control. *Infocom 1992.*

[5] J. Roberts *et al.* [1996]. *Methods for the performance evaluation and design of broadband multiservice networks.* Springer.

[6] A. Simonian and J. Guibert [1995]. Large deviation approximation for fluid queues fed by a large number of on/off sources. *IEEE Journal on Selected Areas in Communications,* 1017-1027.

Periodic Polling Systems in Heavy Traffic

R.D. van der Mei
AT&T Labs
Holmdel
USA
E-mail: rob@buckaroo.ho.att.com

Abstract

We consider polling systems in heavy traffic, with general mixtures of exhaustive and gated service, and in which the server visits the queues according to a general service order table. We derive exact expressions for the expected delay at each of the queues (under heavy-traffic scalings), requiring the solution of a set of only $M - N$ linear equations, where M is the length of the service order table and N is the number of queues. The results lead to new insights into the behavior of polling systems in heavy traffic and moreover, suggest simple and fast-to-evaluate approximations for the expected delays at each of the queues. Numerical experiments show that the approximations are very accurate in practical heavy-load scenarios.

1 Introduction

The basic polling system consists of a number of queues attended by a single server which visits the queues in cyclic order to render service to the customers waiting at the queues. Polling models find many applications in computer-communication systems, and are also widely applicable in the areas of maintenance, manufacturing and production. The reader is referred to [12, 15] for overviews of the applicability of polling models, and to [14, 16] for overviews of the state-of-the-art in the analysis of polling models. In many applications the order in which the server visits the queues is not necessarily cyclic. A natural extension to cyclic server routing is the so-called periodic server routing, in which the order in which the server visits the queues is prescribed by a general service order table (polling table) of finite length.

A detailed exact analysis of polling models is only possible in special cases, and even then usually numerical techniques have to be used to determine performance measures of interest, like expected waiting times. The ultimate goal of performance modeling and analysis is to obtain the 'best' possible system performance. The proper operation of the system is particularly critical when the system is heavily loaded. However, the efficiency of the numerical algorithms may degrade significantly when the system reaches saturation. Moreover, numerical analysis can only to a limited extent contribute to gaining insight into the system behavior. Exact expressions provide much more insight into the dependence of the performance measures with respect to the system parameters. These observations raise the importance of an exact asymptotic analysis of the performance of polling models in heavy traffic.

In the literature, several papers have been devoted to the determination of the expected delay in periodic polling models with N queues and a general polling table of length M. Eisenberg [7] and Alford and Muntz [1] use the so-called buffer-occupancy method to derive a set of $O(M^3)$ linear equations to obtain the expected waiting times for models with exhaustive and gated service at all queues, respectively. Baker and Rubin [3] use the so-called station-time technique to obtain the expected delays in models with exhaustive service at all queues, requiring the solution of a set of $O(M^2)$ linear equations. Recently, an iterative numerical technique based on the concept of descendant sets (cf. [11, 6]) has been proposed to obtain the moments of the delay. Eisenberg [8] analyzes the waiting times in periodic polling systems with exhaustive service at each queue under a variety of stop-and-resume rules when the system gets empty. For models with limited-type service disciplines, Blanc [4] shows how the so-called power-series algorithm (PSA) may be applied to obtain the main performance measures. Van der Mei and Borst [17] show how periodic polling models with multiple servers can be analyzed by means of the PSA. The main drawback of each of the numerical techniques is that their efficiency degrades significantly when M is large and the system is heavily loaded.

In this paper we show how the expected delays in heavily loaded systems can be obtained more efficiently, by exploring heavy-traffic assumptions. We obtain exact expressions for the expected delay under heavy-traffic scalings, requiring the solution of a set of only $M - N$ linear equations. The asymptotic results suggest approximations for the expected delays in heavily loaded systems. Numerical results are presented to show that the approximations are accurate in practical heavy-traffic scenarios. For compactness of the presentation, details of the proofs of the various results are omitted.

In section 2 the model is described and some notation is introduced. In section 3 we give some preliminary results. In section 4 we obtain exact expressions for the expected delay under heavy-traffic scalings, and discuss some implications of the results. In section 5 we propose and test a simple and fast-to-evaluate mean waiting-time approximation.

2 Model Description

Consider a system consisting of N infinite-buffer queues, $Q_1, \ldots, Q_N$. Customers arrive at Q_i according to a Poisson arrival process with rate λ_i. The total arrival rate is denoted by $\Lambda = \sum_{i=1}^{N} \lambda_i$. The service time of a customer at Q_i is a random variable B_i, with finite first and second moments b_i and $b_i^{(2)}$. The first two moments of an arbitrary service time are denoted by $b = \sum_{i=1}^{N} \lambda_i b_i / \Lambda$ and $b^{(2)} = \sum_{i=1}^{N} \lambda_i b_i^{(2)} / \Lambda$. The load offered to Q_i is $\rho_i = \lambda_i b_i$, and the total offered load is equal to $\rho = \sum_{i=1}^{N} \rho_i$. A single server inspects the queues periodically according to a general polling table of finite length M, described by a mapping $T : \{1, \ldots, M\} \to \{1, \ldots, N\}$, which is used such that the server visits the queues periodically in the order $T(1), T(2), \ldots, T(M), T(1), T(2), \ldots$. Following the approach in [3], a unique pseudo-queue will be associated with each entry in the polling table. Denote by PQ_k the pseudo-queue associated with the k-th entry in the polling table; its corresponding queue has index $T(k)$. Customers which arrive at $Q_{T(k)}$ and are served at PQ_k are referred to as type-k customers. The moments at which the server arrives at PQ_k are referred to as the polling instants at PQ_k. Define a service period at PQ_k as the time between a polling instant at PQ_k and its successive departure from PQ_k. The service at each pseudo-queue is either according to the gated policy or the exhaustive policy. Under the gated policy only the type-k customers present at PQ_k at the polling instant at PQ_k are served; customers which arrive at PQ_k while it is being served are not served during the current visit period.

Under the exhaustive policy the server visits PQ_k until it is empty. We allow mixed service policies, e.g. exhaustive service at PQ_1 and PQ_4, and gated service at PQ_2 and PQ_3. For ease of the discussion, we assume that pseudo-queues corresponding to the same queue have the same service strategy. Define $E := \{i : Q_i \text{ is served exhaustively}\}$ and $G := \{i : Q_i \text{ is served according to the gated policy}\}$. At each queue the customers are served on a FIFO basis. After completing service at PQ_i the server proceeds to PQ_{i+1}, incurring a switch-over period whose duration is an independent random variable R_i. The first two moments of R_i are denoted by r_i and $r_i^{(2)}$. Denote the first moment of the total switch-over time in a cycle by $r = \sum_{i=1}^{N} r_i$, and the second moment by $r^{(2)} = \sum_{i=1}^{M} r_i^{(2)} + \sum_{i,j=1}^{M} \sum_{i \neq j} r_i r_j$. It is assumed throughout that $r > 0$.

All interarrival times, service times and switch-over times are assumed to be mutually independent and independent of the state of the system. A necessary and sufficient condition for the stability of the system is $\rho < 1$ [9]. In the sequel, it is assumed that this condition is satisfied, and that the system is in steady state, unless indicated otherwise.

Denote by W_k the delay incurred by an arbitrary customer at Q_k. Our main interest is in the behavior of $E[W_k]$, the expected delay at Q_k, in heavy traffic. Throughout, $E[W_k]$ will be considered as function of ρ; to be specific, we assume that the arrival rates are parametrized as $\lambda_i = a_i \rho$, where relative arrival rates a_i remain fixed. It is known that when $\rho \uparrow 1$, all queues become instable and hence, $E[W_k]$ tends to infinity for all k (cf. [9]). Although a rigorous proof has not been found in the literature, we assume that $E[W_k]$ has a first-order pole at $\rho = 1$: for $k = 1, \ldots, M$,

$$E[W_k] = \frac{\omega_k}{1-\rho} + o((1-\rho)^{-1}), \quad (\rho \uparrow 1), \tag{1}$$

where $o((1-\rho)^{-1})$ stands for a function of ρ which becomes negligible compared to $(1-\rho)^{-1}$ when $\rho \uparrow 1$. Based on equation (1), the analysis will be oriented towards the determination of

$$\omega_k = \lim_{\rho \uparrow 1} (1-\rho)\ E[W_k],\ k = 1, \ldots, M, \tag{2}$$

the scaled expected delay at Q_k, also referred to as the heavy-traffic residue of $E[W_k]$ at $\rho = 1$. In words, ω_k indicates the rate at which $E[W_k]$ tends to infinity as $\rho \uparrow 1$. We denote by W_k^{PQ} the waiting time of a customer which is served at PQ_k. Similar to the definitions above, denote by ω_k^{PQ} the heavy-traffic residue of $E[W_k^{PQ}]$.

Finally we introduce some notation. Let τ_{ij} be the entry in the polling table corresponding to the next visit to Q_j after a departure from PQ_i, and let σ_{ij} be the entry corresponding to the last visit to Q_j prior to an arrival of the server at PQ_i $(i = 1, \ldots, M,\ j = 1, \ldots, N)$. Moreover, let $z_{ij} := 1$ if $PQ_{i+1}, \ldots, PQ_j$ do not correspond to $Q_{T(i)}$ and $j \neq i$, while $z_{ij} := 0$ in all other cases. I_E stands for the indicator function on the event E.

3 Preliminaries

Let X_k be the steady-state number of customers at PQ_k at a polling instant at PQ_k and let V_k be the steady-state duration of a visit period of the server to PQ_k $(k = 1, \ldots, M)$. For a customer served at PQ_k, we define the waiting time at PQ_k to be the time between its arrival in the system (at $Q_{T(k)}$) and the moment at which the customer starts service at

PQ_k. The expected waiting time at PQ_k can be expressed in terms of the first two moments of X_k as follows (cf. [13]):

$$E[W_k^{PQ}] = \frac{Var[X_k] + (E[X_k])^2 - E[X_k]}{2\lambda_{T(k)}E[X_k]}(1 + \rho_{T(k)}), \quad (T(k) \in G), \tag{3}$$

and

$$E[W_k^{PQ}] = \frac{Var[X_k] + (E[X_k])^2 - E[X_k]}{2\lambda_{T(k)}E[X_k]} + \frac{\lambda_{T(k)}b_{T(k)}^{(2)}}{2(1 - \rho_{T(k)})}, \quad (T(k) \in E). \tag{4}$$

Thus, to obtain the expected waiting time at PQ_k, we need to quantify $E[X_k]$ and $Var[X_k]$. In 3.1 we discuss how $E[X_k]$ can be obtained by solving a set of linear equations. In 3.2 we discuss how the Descendant Set Approach (DSA) can be used to obtain $Var[X_k]$.

3.1 Determination of $E[X_k]$

To obtain an expression for $E[X_k]$, it is convenient to relate $E[X_k]$ to $E[V_k]$, the expected duration of a visit to PQ_k. Simple balancing arguments show that: for $k = 1, \ldots, M$,

$$E[V_k] = \varphi_{T(k)}E[X_k], \text{ where } \varphi_i := b_i \ \ (i \in G), \quad \varphi_i := \frac{b_i}{1 - \rho_i} \ \ (i \in E). \tag{5}$$

The variables $E[V_k]$, and hence $E[X_k]$, can be obtained by solving the following set of linear equations (cf. also [5]): for $k = 1, \ldots, M$,

$$E[V_k] = \lambda_{T(k)}\varphi_{T(k)}\left[\sum_{j=l+1}^{k-1}(r_j + E[V_j]) + r_l + E[V_l]I_{\{T(l)\in G\}}\right], \tag{6}$$

where $l := \sigma_{k\ T(k)}$, the entry corresponding to the last visit to $Q_{T(k)}$ prior to PQ_k. Balancing the flow of customers at Q_k in and out of the system during one cycle of the server along the queues implies: for $i = 1, \ldots, N$,

$$\sum_{m:T(m)=i} E[V_m] = \rho_i \frac{r}{1 - \rho}. \tag{7}$$

One may verify that equations (6)-(7) lead to a set of $M - N$ simultaneous linear equations for $E[V_k]$ and hence for $E[X_k]$. Since the analysis will be devoted to heavy-traffic behavior of the system, we note that equations (6)-(7) imply that $E[V_k]$ and $E[X_k]$ have a first-order pole at $\rho = 1$. Therefore, we define: for $k = 1, \ldots, M$,

$$v_k := \lim_{\rho \uparrow 1}(1 - \rho)E[V_k], \quad x_k := \lim_{\rho \uparrow 1}(1 - \rho)E[X_k], \tag{8}$$

referred to as the heavy-traffic residues of $E[V_k]$ and $E[X_k]$, respectively. Using equations (6)-(8) it follows that the variables v_k $(k = 1, \ldots, M)$ are (uniquely) determined by the following set of equations: for $k = 1, \ldots, M$,

$$v_k = \lambda_{T(k)}\varphi_{T(k)}\left[\sum_{j=l+1}^{k-1} v_j + v_l I_{\{T(l)\in G\}}\right], \quad \sum_{m:T(m)=i} v_m = \rho_i r \ (i = 1, \ldots, N). \tag{9}$$

Note that in (9) the parameters $\lambda_{T(k)}$, $\varphi_{T(k)}$ and ρ_i have to be evaluated at $\rho = 1$. The computation of the variables v_k $(k = 1, \ldots, M)$, requires the solution of a set of $M - N$ linear equations. Using (5) it follows that once v_k is known, x_k is given by $x_k = v_k/\varphi_{T(k)}$.

3.2 The Descendant Set Approach

The Descendant Set Approach (DSA) classifies the customers in a polling system into two classes. An *originator* is a customer which arrives during a switch-over period, and a *non-originator* is a customer which arrives during the service of another customer. For a tagged customer C, the so-called children set of C is the set of customers arriving to the system during the service of C; the descendant set of C is recursively defined to consist of C, its children (if any) and the descendants of its children. To determine the moments of the delay of customers arriving at a fixed PQ_k, the DSA concentrates on the determination of $X_k(P)$, defined as the number of customers at PQ_k present at an arbitrary, but fixed, polling instant P at PQ_k. P is referred to as the *reference point* at PQ_k. The main idea of the DSA is the observation that each of the $X_k(P)$ type-k customers present at PQ_k at P is a descendant of exactly one originator, and that the descendant sets of the originators evolve independently. Therefore, the DSA concentrates on an arbitrary type-i customer served at PQ_i and on calculating the number of type-k descendants it has at P.

The DSA considers the Markov process embedded at the polling instants of the system. To this end, we number the successive polling instants as follows. Let $P_{M,0}$ be an arbitrary polling instant at PQ_M, and for $i = M-1, \ldots, 1$, let $P_{i,0}$ be recursively defined as the polling instant at PQ_i prior to $P_{i+1,0}$. In addition, for $c = 1, 2, \ldots$, we define $P_{i,c}$ to be the last polling instant at PQ_i prior to $P_{i,c-1}$, $i = 1, \ldots, M$. We consider a tagged customer $C_{i,c}$ which is present at PQ_i at $P_{i,c}$. Define $A_{(i,c),k}$ to be the number of type-k descendants $C_{i,c}$ has at $P_{k,0}$. In this way, $A_{(i,c),k}$ can be viewed as the contribution of $C_{i,c}$ to $X_k(P_{k,0})$. If we define $\alpha_{(i,c),k} = E[A_{(i,c),k}]$ and $\alpha^{(2)}_{(i,c),k} = E[A_{(i,c),k}(A_{(i,c),k} - 1)]$, then $E[X_k]$ and $Var[X_k]$ can be expressed as follows (cf. [11]):

$$E[X_k] = \sum_{i=1}^{M} r_i \sum_{c=0}^{\infty} \left[\sum_{j:\tau_{ij}>i} \lambda_j \alpha_{(\tau_{ij},c),k} + \sum_{j:\tau_{ij}\leq i} \lambda_j \alpha_{(\tau_{ij},c-1),k} \right], \tag{10}$$

and

$$\begin{aligned} Var[X_k] &= \textstyle\sum_{i=1}^{M} (r_i^{(2)} - r_i^2) \sum_{c=0}^{\infty} \left[\sum_{j:\tau_{ij}>i} \lambda_j \alpha_{(\tau_{ij},c),k} + \sum_{j:\tau_{ij}\leq i} \lambda_j \alpha_{(\tau_{ij},c-1),k} \right]^2 \\ &\quad + \textstyle\sum_{i=1}^{M} r_i \sum_{c=0}^{\infty} \left[\sum_{j:\tau_{ij}>i} \lambda_j \alpha^{(2)}_{(\tau_{ij},c),k} + \sum_{j:\tau_{ij}\leq i} \lambda_j \alpha^{(2)}_{(\tau_{ij},c-1),k} \right]. \end{aligned} \tag{11}$$

The variables $\alpha_{(i,c),k}$ and $\alpha^{(2)}_{(i,c),k}$ can be computed recursively. To this end, note that the contribution to $X_k(P_{k,0})$ of the tagged customer $C_{i,c}$ is equal to the total contribution to $X_k(P_{k,0})$ of its children. This observation leads to the following set of recursive relations (for the case of gated service at PQ_i): for $k = 1, \ldots, M$, $c = 0, 1, \ldots$,

$$\alpha_{(i,c),k} = b_{T(i)} \left[\sum_{j:\tau_{ij}>i} \lambda_j \alpha_{(\tau_{ij},c),k} + \sum_{j:\tau_{ij}\leq i} \lambda_j \alpha_{(\tau_{ij},c-1),k} \right] \quad (T(i) \in G), \tag{12}$$

and

$$\alpha^{(2)}_{(i,c),k} = \frac{b^{(2)}_{T(i)}}{b^2_{T(i)}} \alpha^2_{(i,c),k} + b_{T(i)} \left[\sum_{j:\tau_{ij}>i} \lambda_j \alpha^{(2)}_{(\tau_{ij},c),k} + \sum_{j:\tau_{ij}\leq i} \lambda_j \alpha^{(2)}_{(\tau_{ij},c-1),k} \right] \quad (T(i) \in G). \tag{13}$$

For the case of exhaustive service at PQ_i, similar expressions can be obtained in a straightforward manner. The initial conditions are given by $\alpha_{(k,0),k} := 1$; $\alpha_{(i,0),k} := 0$ $(i = k+1, \ldots, M)$; $\alpha_{(i,-1),k} := 0$ $(i = 1, \ldots, k-1)$. Starting with these initial values, all coefficients $\alpha_{(i,c),k}$ and $\alpha^{(2)}_{(i,c),k}$ can be recursively determined according to (12) and (13).

For the analysis, we also need to conduct the recursion in a different way: rather than carrying it out via the children of a given customer, we carry it out via the so-called *immediate parents* of the descendants. We consider again a tagged customer $C_{i,c}$ present at PQ_i at $P_{i,c}$, and try to find the contribution of $C_{i,c}$ to $X_k(P_{k,0})$. To this end, we consider the most recent polling instants of PQ_j prior to $P_{k,0}$, denoted by $P_j^*, j = 1, \ldots, M$; thus, $P_j^* = P_{j,0}$ for $j = 1, \ldots, k-1$, and $P_j^* = P_{j,1}$ for $j = k, \ldots, M$. We recursively derive the contribution of $C_{i,c}$ to $X_k(P_{k,0})$ as function of the contribution of $C_{i,c}$ to $X_j(P_j^*)$. A crucial observation is that the distribution of the contribution of $C_{i,c}$ to $X_j(P_j^*)$ is identical to that of $A_{(i,c),j}$ for $j = 1, \ldots, k-1$, and to that of $A_{(i,c-1),j}$ for $j = k, \ldots, M$. To relate the number of descendants of $C_{i,c}$ at P_j^* to the number of type-k descendants of $C_{i,c}$ at $P_{k,0}$, we observe that each type-k customer $C_{k,0}$ at $P_{k,0}$ has arrived during the sub-busy period generated by exactly one customer C_j^* present at P_j^* for some $j = l, \ldots, k-1$, where $l = \sigma_{k\ T(k)}$, the entry corresponding to the last visit to $Q_{T(k)}$ prior to a visit of the server to PQ_k. C_j^* is referred to as the immediate parent of $C_{k,0}$. Note that if $T(j) \in G$, then the sub-busy period generated by C_j^* is just the service of C_j^* itself, and that in case $T(j) \in E$, the sub-busy period generated by C_j^* is an M/G/1 busy period (with arrival rate and service-time distribution corresponding to those of $Q_{T(j)}$). To relate the number of descendants of $C_{i,c}$ at P_j^* to the number of type-k descendants of $C_{i,c}$ at $P_{k,0}$, we observe that C_j^* $(j = l, \ldots, k-1)$ is the immediate parent of on the average $\lambda_{T(k)} d_{jk}$ type-k customers at $P_{k,0}$, where $d_{jk} := b_j$ for $j \in G$, and $d_{jk} := I_{\{T(k) \neq j\}} b_j/(1-\rho_j)$ for $j \in E$ $(k = 1, \ldots, M)$. These observations lead to the following recursive relations: for $c = 0, 1, \ldots$,

$$\alpha_{(i,c),k} = \lambda_{T(k)} \sum_{j=l}^{k-1} \left[d_{jk} \alpha_{(i,c),j} I_{\{j<k\}} + d_{jk} \alpha_{(i,c-1),j} I_{\{j \geq k\}} \right], \tag{14}$$

where $l = \sigma_{k\ T(k)}$. The initial conditions are given by $\alpha_{(i,0),i} := 1$; $\alpha_{(i,0),k} := 0$ $(k = 1, \ldots, i-1)$; $\alpha_{(i,-1),k} := 0$ $(k = i+1, \ldots, M)$.

It is useful to express the recursive relation (12) in matrix notation. To this end, for $i = 1, \ldots, M$, $c = 0, 1, \ldots$, let $\mathbf{a}_{(\cdot,c),k}$ be the vector whose i-th element is $\alpha_{(i,c),k}$ for $i = k, \ldots, M$, and $\alpha_{(i,c-1),k}$ for $i = 1, \ldots, k-1$. For $i = 1, \ldots, M$, let $\mathbf{P}_i$ be the M by M matrix, whose (j,k)-th element equals $I_{\{j=k\}}$ for $j \neq i$, while the (i, τ_{ij})-th element equals $b_{T(i)} \lambda_j$ if $T(i) \in G$, and $I_{\{j \neq T(i)\}} \lambda_j b_{T(i)}/(1-\rho_{T(i)})$ if $T(i) \in E$, and all other components of the i-th row are 0. If we define for $k = 1, \ldots, M$, $\mathbf{M}_k = \mathbf{P}_k \cdots \mathbf{P}_M \mathbf{P}_1 \cdots \mathbf{P}_{k-1}$, then the recursive relations (12) can be expressed as follows: for $k = 1, \ldots, M$,

$$\mathbf{a}_{(\cdot,0),k} = \mathbf{e}_k, \quad \mathbf{a}_{(\cdot,c),k} = \mathbf{M}_k \mathbf{a}_{(\cdot,c-1),k} = (\mathbf{M}_k)^c \mathbf{e}_k, \quad c = 1, 2, \ldots. \tag{15}$$

To write equations (14) in matrix notation, for $i = 1, \ldots, M$, $c = 0, 1, \ldots$, let $\mathbf{a}_{(i,c),\cdot}$ be the vector whose k-th element is $\alpha_{(i,c),k}$ for $k = 1, \ldots, i$, and $\alpha_{(i,c-1),k}$ for $k = i+1, \ldots, M$. For $k = 1, \ldots, M$, let $\hat{\mathbf{P}}_k$ be the M by M matrix whose (i,j)-th element equals $I_{\{i=j\}}$ for $i \neq k$, while the (k,j)-th element of $\hat{\mathbf{P}}_k$ is given by $\lambda_{T(k)} d_{jk}$ for $j = l, \ldots, k-1$, and all other components of the k-th row are 0. If we define, for $i = 1, \ldots, M$, $\hat{\mathbf{M}}_i = \hat{\mathbf{P}}_i \cdots \hat{\mathbf{P}}_1 \hat{\mathbf{P}}_N \cdots \hat{\mathbf{P}}_{i+1}$, then relations (14) and the initial condition can be expressed as follows: for $i = 1, \ldots, M$,

$$\mathbf{a}_{(i,0),\cdot} = \mathbf{e}_i, \quad \mathbf{a}_{(i,c),\cdot} = \hat{\mathbf{M}}_i \mathbf{a}_{(i,c-1),\cdot} = (\hat{\mathbf{M}}_i)^c \mathbf{e}_i, \ c = 1, 2, \ldots. \tag{16}$$

4 Analysis

The variables $\alpha_{(i,c),k}$ are fully determined by both sets of relations (15) and (16). More precisely, both (15) and (16) constitute a set of homogeneous difference equations of the first

order. From the literature on difference equations it is well-known that the variables $\alpha_{(i,c),k}$ can be solved directly if the eigenvalues and eigenvectors of $\mathbf{M}_k$ ($\hat{\mathbf{M}}_i$) are known. In general, however, the eigenvalues and eigenvectors of $\mathbf{M}_k$ ($\hat{\mathbf{M}}_i$) are unknown for $\rho < 1$. However, to analyze the system behavior in the limiting case $\rho \uparrow 1$, there is no need to obtain all eigenvalues and eigenvectors of $\mathbf{M}_k$ (or $\hat{\mathbf{M}}_i$) and then let $\rho \uparrow 1$. More precisely, since $E[X_k]$ and $Var[X_k]$ are known to tend to infinity when $\rho \uparrow 1$, it follows from (10) and (11) that the heavy-traffic behavior of $E[X_k]$ and $Var[X_k]$ is determined by the *dominant tail behavior* of the sequences $\{\alpha_{(i,c),k}, c = 0, 1, \ldots\}$ and $\{\alpha^{(2)}_{(i,c),k}, c = 0, 1, \ldots\}$, which appears to be relatively simple.

The following property decomposes the matrices $(\mathbf{M}_k)^c$ and $(\hat{\mathbf{M}}_i)^c$ into two parts, one of these parts becomes dominant as c gets large (cf. [2]):

Lemma 1

The matrices $\mathbf{M}_k$ and $\hat{\mathbf{M}}_i$ have respective maximal eigenvalues γ and $\hat{\gamma}$ which are real-valued, positive, have multiplicity 1, and have associated right and left eigenvectors $\mathbf{u}_k$, $\mathbf{w}_k$, and $\hat{\mathbf{u}}_i$, $\hat{\mathbf{w}}_i$, respectively. If these are normalized so that $\mathbf{u}_k^\top \mathbf{w}_k = \mathbf{u}_k^\top \mathbf{1} = 1$, $\hat{\mathbf{u}}_i^\top \hat{\mathbf{w}}_i = \hat{\mathbf{u}}_i^\top \mathbf{1} = 1$, then

$$(\mathbf{M}_k)^c = \gamma^c \mathbf{u}_k \mathbf{w}_k^\top + (\mathbf{R}_k)^c, \quad (\hat{\mathbf{M}}_i)^c = \hat{\gamma}^c \hat{\mathbf{u}}_i \hat{\mathbf{w}}_i^\top + (\hat{\mathbf{R}}_i)^c, \tag{17}$$

where there exist $K < \infty$ and $\bar{\gamma}$ $(0 < \bar{\gamma} < \gamma, \hat{\gamma})$, such that all entries of $(\mathbf{R}_k)^c$ and $(\hat{\mathbf{R}}_i)^c$ are strictly smaller than $K\bar{\gamma}^c$ $(i, k = 1, \ldots, M)$.

Lemma 2

(1) *If $\rho < 1$, then $\gamma, \hat{\gamma} < 1$, and if $\rho = 1$ then $\gamma = \hat{\gamma} = 1$.*
(2) *If $\rho = 1$, then $\mathbf{u}_k$ is proportional to $\mathbf{b} = (b_{T(1)}, \ldots, b_{T(M)})$ for $k = 1, \ldots, M$.*
(3) *If $\rho = 1$, then $\hat{\mathbf{u}}_i$ is proportional to $\mathbf{x} = (x_1, \ldots, x_M)$ for $i = 1, \ldots, M$.*

Part 1 implies that the sequence $\{\alpha_{(i,c),k}, c = 0, 1, \ldots\}$ converges to 0 for $\rho < 1$, and to a (positive) constant for $\rho = 1$. It is readily verified that the sequence $\{\alpha^{(2)}_{(i,c),k}, c = 0, 1, \ldots\}$ converges to 0 for $\rho < 1$, and to a linearly increasing function for $\rho = 1$. Parts 2 and 3 give the ratios between the limiting values of the sequence $\{\alpha_{(i,c),k}, c = 0, 1, \ldots\}$ for $\rho = 1$ among different values of i and k.

Lemma 3

For $i, j, k, l = 1, \ldots, M$,

$$(1) \lim_{\rho \uparrow 1} \frac{\sum_{c=0}^{\infty} \alpha_{(i,c),k}}{\sum_{c=0}^{\infty} \alpha_{(j,c),l}} = \frac{b_{T(i)}}{b_{T(j)}} \frac{x_k}{x_l}; \quad (2) \lim_{\rho \uparrow 1} \frac{\sum_{c=0}^{\infty} \alpha^{(2)}_{(i,c),k}}{\sum_{c=0}^{\infty} \alpha^{(2)}_{(j,c),l}} = \frac{b_{T(i)}}{b_{T(j)}} \frac{x_k^2}{x_l^2}. \tag{18}$$

Part 1 follows directly from Lemmas 1 and 2, using the continuity of eigenvectors and eigenvalues as function of ρ [10]. Part 2 can then be obtained by straightforward manipulations. Lemmas 1-3 lead to the following result.

Theorem 1

$$(1) \lim_{\rho \uparrow 1} \frac{E[X_k]}{E[X_l]} = \frac{x_k}{x_l}; \quad (2) \lim_{\rho \uparrow 1} \frac{Var[X_k]}{Var[X_l]} = \frac{x_k^2}{x_l^2}. \tag{19}$$

Note that the first summation at the right-hand side of (11) has a first-order pole at $\rho = 1$, while the second summation has a second-order pole at $\rho = 1$. Hence, the second summation

dominates the first summation when the system reaches saturation. Combining this observation with equations (3) and (4) leads to the following e xpression for the ratios between the scaled expected waiting times at pseudo-queues.

Theorem 2 (Ratios between scaled expected waiting times at pseudo-queues)
For $k, l = 1, \ldots, M$,

$$\frac{\omega_k^{PQ}}{\omega_l^{PQ}} = \frac{x_k(1+\rho_{T(k)}I_{\{T(k)\in G\}})/\lambda_{T(k)}}{x_l(1+\rho_{T(l)}I_{\{T(l)\in G\}})/\lambda_{T(l)}} = \frac{v_k(1+\rho_{T(k)}I_{\{T(k)\in G\}})/\lambda_{T(k)}\varphi_{T(k)}}{v_l(1+\rho_{T(l)}I_{\{T(l)\in G\}})/\lambda_{T(l)}\varphi_{T(l)}}. \tag{20}$$

The waiting times at the pseudo-queues can be related to the waiting times at the queues by conditioning on π_j, defined as the fraction of customers which arrive at $Q_{T(j)}$ that are served at PQ_j $(j = 1, \ldots, M)$. Using this definition, it is readily seen that: for $j = 1, \ldots, M$,

$$\pi_j = \frac{v_j}{\sum_{k:T(k)=T(j)} v_k} = \frac{v_j}{\rho_{T(j)}r}, \tag{21}$$

where the second equality follows from (9). Hence, for $i = 1, \ldots, N$,

$$\omega_i = \sum_{j:T(j)=i} \pi_j \omega_j^{PQ} = \frac{1}{\rho_i r} \sum_{j:T(j)=i} v_j \omega_j^{PQ}. \tag{22}$$

Combining (22) and Theorem 2 leads to the following expression for the ratios between the scaled expected waiting times at the queues.

Theorem 3 (Ratios between scaled expected waiting times at queues)
For $i, j = 1, \ldots, N$,

$$\frac{\omega_i}{\omega_j} = \frac{\eta_i/\rho_i^2}{\eta_j/\rho_j^2} \frac{\sum_{k:T(k)=i} v_k^2}{\sum_{k:T(k)=j} v_k^2}, \quad \text{where } \eta_k := 1+\rho_k \ \ (k \in G), \quad \eta_k := 1-\rho_k \ \ (k \in E). \tag{23}$$

Based on Theorem 3, the scaled expected waiting times are known up to some unknown scaling factor. This scaling factor can easily be obtained by using the pseudo-conservation law for the model under consideration (cf. [5]), an exact expression for a specific weighted sum of the expected waiting times. By applying some straightforward algebraic manipulations, we obtain the following relation:

$$\sum_{i=1}^{N} \rho_i \omega_i = \frac{b^{(2)}}{2b} + \sum_{k=1}^{M} \rho_{T(k)} \sum_{m \neq k} \frac{r_m}{r} z_{km} \sum_{j=k}^{m-1} v_{j+1} + \sum_{j \in G} \rho_{T(j)} v_j \sum_{m=1}^{M} \frac{r_m}{r} z_{jm} + \sum_{m \in G} \rho_{T(m)} \frac{r_m}{r} v_m. \tag{24}$$

Theorem 4
For $i = 1, \ldots, N$,

$$\omega_i = \frac{(\eta_i/\rho_i^2) \sum_{k:T(k)=i} v_k^2}{\sum_{j=1}^{N} (\eta_j/\rho_j) \sum_{k:T(k)=j} v_k^2} C, \tag{25}$$

where C *is the right-hand side of* (24).

Note that all terms in Theorems 2-4 corresponding to arrival rates and loads at the queues have to be evaluated at $\rho = 1$.

In the special case of cyclic server routing, we have $M = N$, $v_k = \rho_k r$ $(k = 1, \ldots, N)$, and the right-hand side of (24) is given by $C = b^{(2)}/2b + r[1 - \sum_{i \in E} \rho_i^2 + \sum_{i \in G} \rho_i^2]$, so that

$\omega_i = C\eta_i / \sum_j \rho_j \eta_j$, which corresponds to the result obtained in [18].

Theorem 4 provides new insights into how the expected waiting times depend on the system parameters.

First, we observe that ω_k depends on the service time distributions only through $b^{(2)}/2b$, the ratio between the first two moments of the service time of an *arbitrary* customer, rather than on the (first two) moments of the individual service times.

Second, ω_k also depends on the switch-over times only through r, the *first* moment of the *total* switch-over time per cycle of the server along the queues. To this end, note that it follows from (9) that v_k is a linear function of r. Moreover, simple arguments show that in the limiting case $\rho \uparrow 1$ the quantity C in Theorem 4, which stands for the heavy-traffic residue of the expected amount of waiting work in the system, also depends on the individual switch-over times only through r. In fact, since v_k is linear in r, it is easy to verify that the right-hand side of (24) is the sum of a constant (i.e., $b^{(2)}/2b$) and a linear function in r.

The observed insensitivities are not generally valid for stable systems (i.e., $\rho < 1$). Apparently, the dependencies of the expected waiting times on the distributions of the individual service times and switch-over times vanish when the system reaches saturation.

5 Approximation

Theorem 4 implies that in heavy load the expected waiting times can be approximated by equation (25), divided by $1 - \rho$ (see equation (1)). This is equivalent to approximating the ratios between the expected waiting times by the right-hand side of (23). One may question the practicality of 'heavy traffic', in the sense of 'how heavy should the traffic be' to make the approximation (23) work well. To demonstrate this issue, we consider a system with the following parameters: $N = 4$; $M = 8$; $T = (1,2,1,3,4,2,1,3)$; $G = \{1,4\}$; $E = \{2,3\}$; the ratio between the arrival rates is $5:1:1:1$; the service times are exponentially distributed with means $b_1 = 1, b_2 = 1, b_3 = 5, b_4 = 1$; the switch-over times are exponentially distributed with means $r_i = 0.05$ for $i \in \{1,2,3,4,5,6,7\}$ and $r_8 = 5$. The system is therefore very asymmetrical in the arrival rates, service times and switch-over times.

Table 1 below shows the values of the ratio $\delta_k(\rho) := E[W_k]/E[W_1]$ $(k = 2,3,4)$ for different values of ρ. The 'limit' in Table 1 represents the results in the limiting case, obtained from (23) and (9). To measure the accuracy of the approximation, the relative difference between $\delta_k(\rho)$ and $\delta_k(\text{limit})$, i.e., the 'exact' ratio in the limiting case,

$$\text{abs}\left(\frac{\delta_k(\rho) - \delta_k(\text{limit})}{\delta_k(\text{limit})}\right) \times 100\%, \tag{26}$$

is indicated in parentheses. We used the numerical technique in [6] to determine the expected waiting times. Note that the computation time required to obtain the results for the higher values of ρ in Table 1 was in the order of several minutes, while the results for the 'limit' case were obtained in less than a second (on a SUN SPARC 4 work station).

ρ	$\delta_2(\rho)$	$\delta_3(\rho)$	$\delta_4(\rho)$
0.50	1.023 (9.9)	1.009 (75.5)	1.596 (25.3)
0.60	0.991 (6.4)	0.895 (55.7)	1.629 (23.8)
0.70	0.967 (3.9)	0.780 (35.7)	1.680 (21.4)
0.80	0.952 (2.3)	0.681 (18.4)	1.772 (17.1)
0.90	0.943 (1.3)	0.610 (6.1)	1.922 (10.1)
0.95	0.938 (0.8)	0.588 (2.3)	2.021 (5.4)
0.98	0.934 (0.3)	0.579 (0.7)	2.089 (2.2)
0.99	0.933 (0.2)	0.577 (0.3)	2.113 (1.1)
limit	0.931	0.575	2.137

Table 1. Ratios between the expected waiting times as function of the load.

The results in Table 1 shows that the accuracy of the approximations increases when the load increases, as expected. The accuracy of the approximation for given ρ varies from queue to queue. In the worst case, the relative error is less than 20% for $\rho = 0.80$ and is about 10% for $\rho = 0.90$. This implies that for most practical cases the approximation can be used with good confidence; this implication stems from the fact that in practice heavy load is the main region of interest. We emphasize that the example presented here is very asymmetrical in the arrival rates, the service rates and the switch-over times, and that the approximations are considerably more accurate in most cases.

Acknowledgements
The author wishes to thank O.J. Boxma for his useful comments on an earlier draft of this paper, and G.L. Choudhury for making the software program corresponding to [6] available for performing numerical experiments.

References

[1] Alford, M. and Muntz, R.R. (1975). Queueing models for polled multi-queues. In: *Fourth Texas Conf. Comput. Syst.*, 5B-2.1-2.5.

[2] Athreya, K.B and Ney, P.E. (1971). *Branching Processes* (Springer, Berlin).

[3] Baker, J.E. and Rubin, I.R. (1987). Polling with a general-service order table. *IEEE Trans. Commun.* **35**, 283-288.

[4] Blanc, J.P.C. (1993). Performance analysis and optimization with the power-series algorithm. In: *Performance Evaluation of Computer and Communication Systems*, eds. L. Donatiello and R. Nelson (Springer-Verlag, Berlin), 53-80.

[5] Boxma, O.J., Groenendijk, W.P. and Weststrate, J.A. (1990). A pseudoconservation law for service systems with a polling table. *IEEE Trans. Commun.* **38**, 1865-1870.

[6] Choudhury, G. and Whitt, W. (1994). Computing transient and steady state distributions in polling models by numerical transform inversion. *Perf. Eval.* **25**, 267-292.

[7] Eisenberg, M. (1972). Queues with periodic service and changeover time. *Oper. Res.* **20**, 440-451.

[8] Eisenberg, M. (1994). The polling system with a stopping server. *Queueing Systems* **18**, 387-431.

[9] Fricker, C. and Jaïbi, M.R. (1994). Monotonicity and stability of periodic polling models. *Queueing Systems* **15**, 211-238.

[10] Kato, T. (1966). *Perturbation Theory for Linear Operators* (Springer-Verlag, New York).

[11] Konheim, A.G., Levy, H. and Srinivasan, M.M. (1994). Descendant set: an efficient approach for the analysis of polling systems. *IEEE Trans. Commun.* **42**, 1245-1253.

[12] Levy, H. and Sidi, M. (1991). Polling models: applications, modeling and optimization. *IEEE Trans. Commun.* **38**, 1750–1760.

[13] Takagi, H. (1986). *Analysis of Polling Systems* (The MIT Press, Cambridge, MA).

[14] Takagi, H. (1990). Queueing analysis of polling models: an update. In: *Stochastic Analysis of Computer and Communication Systems*, ed. H. Takagi (North-Holland, Amsterdam), 267-318.

[15] Takagi, H. (1991). Applications of polling models to computer networks. *Comp. Netw. ISDN Syst.* **22**, 193-211.

[16] Takagi, H. (1994). Queueing analysis of polling models: progress in 1990-1993. In: *Frontiers in Queueing: Models, Methods and Problems*, ed. J.H. Dhsalalow (CRC Press).

[17] Van der Mei, R.D. and Borst, S.C. (1997). Analysis of multiple-server polling systems by means of the power-series algorithm. To appear in *Stoch. Mod.*

[18] Van der Mei, R.D. and Levy, H. (1996). Expected delay analysis of polling systems in heavy traffic. Submitted.

Heuristics for Complex Inventory Systems*

Erik van der Sluis
Vrije Universiteit Amsterdam
Bedrijfswiskunde & Informatica
Amsterdam
E-mail: eriks@cs.vu.nl

1 Introduction

Research in inventory theory is often focused on either deterministic or stochastic problems. For the deterministic problems optimal solutions, heuristics and complexity results are important issues. For the stochastic problems optimal strategies, subclasses of strategies, heuristics and bounds on average cost are of interest. From a practical point of view both types of models are too restrictive. Deterministic models lack the notion of uncertainty in demand, in supply or within the system itself. Stochastic models, on the other hand, are often complicated and therefore difficult to comprehend. Moreover, incorporating deterministic variables complicates the model even more. Dealing with real world problems a combination of deterministic and stochastic models is needed. In practice, one settles for deterministic models and incorporates uncertainty by, for example, creating safety stocks.

In my thesis I studied the behaviour of deterministic models in stochastic systems. Not only cost aspects are considered, but also the usefulness of the information obtained from these models. Unfortunately, an optimal or heuristic solution of a stochastic model generates only a decision given the current state of the system and gives no information about actions that will be taken in later periods. Also, additional information about future demand is neglected, for example, orders of special customers. To incorporate such information or to generate information on future actions deterministic models are needed. These models should take into account the stochastic nature of the system.

Common practice for solving a complex stochastic inventory system is to separate the problem into a deterministic lot-size problem using forecasts for the stochastic variables during a finite number of periods on one hand and a problem of determining safety stocks on the other hand, see for example Schneeweiss (1986). The procedure of updating forecasts and solving these problems periodically is referred to as a rolling horizon procedure and consists of the following three basic steps:

1. Forecast the outcomes of the relevant stochastic variables, usually the demand, during a planning horizon of N periods;
2. Determine an ordering schedule for the next N periods, and safety stocks;
3. Implement the decision for the first period in the stochastic system.

*This is a summary of a PhD thesis with the same title

At the time of the next decision new demand information is added to maintain a constant planning horizon and old information is updated. For each step different choices can be made, for example, we can use expected demand as a forecast or overestimate the demand; various procedures may exist to solve the deterministic problem and to determine safety stocks; and finally, we can analyse and adjust the decision for the first period before implementing it in the system. All in all, the behaviour of the rolling horizon procedure in a stochastic system can be influenced in numerous ways.

How can we evaluate the effectiveness of this approach? An important criterion is the average cost performance in the long-run. The average cost can be obtained by simulation experiments, or sometimes analytically, and can be compared with the cost of an optimal or an alternative strategy or with a lower bound on the minimal cost. However, these strategies are often based on the assumption of identical and independent distributed demand per period. Relaxation of this assumption, for instance, by introducing time-dependent demand, growing demand or dependency between demand for several products, often complicates a problem such that alternatives are (almost) impossible to obtain. In that case, applying a rolling horizon procedure is the only practical approach.

Another aspect that has received more and more attention in recent years is the stability of the generated schedules. Stability is especially important when information about future replenishments is used for production planning and personnel scheduling at the supplier. At the time of a decision a deterministic schedule is generated. Due to the stochastic nature of the demand and due to the fact that new demand information is incorporated the timing and sizes of future replenishments may be different from the previous schedule. Frequent adjustments of the schedules is undesirable and is referred to as system nervousness. A change in size is often regarded as less costly than a change in timing. Carlson et al. (1979) introduced penalty costs for changing the timing of replenishments in a rolling horizon approach for single-product problems; see Jensen (1992) for different measures of system nervousness.

In my thesis rolling horizon procedures are examined for three types of inventory systems: single- and multi-product and multi-retailer systems. Given a certain system, we examine some alternatives; based on cost aspects and system nervousness we choose those that suit best. Furthermore, the influence of the planning horizon is examined. Sometimes negative effects of the planning horizon can be eliminated by changing the solution method or by solving a different deterministic problem. In this outline we describe the approach for single-product systems.

2 Single-product systems

We considered the most elementary single-product problem under periodic review: given a fixed setup cost k, unit holding cost h for each unit in stock at the end of a period and stochastic demand per period $\mathbf{d}$ with known pdf, determine an optimal replenishment strategy, that minimises the long-run average cost. Orders arrive after a fixed lead time of L periods. Furthermore, we assume that a minimal inventory level s^- is required to maintain a certain service level and demand not satisfied directly from stock is backordered.

Order decisions are based on the economic inventory $\mathbf{s}$ at the start of a period, that is, the inventory on hand plus on order minus on backorder. If the economic inventory falls below s^- then it has to be replenished. That is, the reorder level is sufficient to satisfy demand in the next $L+1$ periods according to some service level constraint.

It is well known that for the described inventory systems an (s^-, S)-strategy is optimal.

Under such a strategy the economic inventory is only ordered up to level S when it has fallen below the reorder level s^-. So, the decision for the stochastic system $\mathbf{x} = S - \mathbf{s}$ for $\mathbf{s} < s^-$ and $\mathbf{x} = 0$ otherwise. For this strategy we can derive some useful properties based on renewal theory, see Tijms (1986). The expected order size $Q(s^-, S)$ and expected reorder interval $T(s^-, S)$ are given by

$$Q(s^-, S) = S - s^- + (c^2 + 1)\mu/2 \tag{1}$$

$$T(s^-, S) = Q(s^-, S)/\mu$$

with μ the expected demand and c^2 the coefficient of variation. Note that, although the strategy is applied in period review systems, the expected reorder interval is in general not an integral value.

2.1 Rolling horizon procedure

Given the current state of a single-product system $\mathbf{s}$, the rolling horizon procedure should generate a decision $\mathbf{x}$ for the stochastic system. When no additional information is available, a usual way of setting the deterministic demand is: $d_1 = s^-$ and $d_t = \mu$ for $t > 1$. The demand d_1 represents the total requirement in the first $L + 1$ periods, the lead time plus review time. Every review period the decision $\mathbf{x}$ is determined by solving

$$\begin{aligned} \min_x \quad & \sum_{t=1}^{N} \{k\delta(X_t) + hI_t\} \\ \text{such that} \quad & I_t = I_{t-1} + X_t - d_t, \ t = 1, ..., N, \\ & I_t, X_t \geq 0, \ t = 1, ..., N, \\ & I_0 = \mathbf{s}, \ \mathbf{x} = X_1, \end{aligned}$$

with X_t denoting the replenishment decision and I_t the ending inventory of period t. This problem is known as the Wagner-Whitin problem (1958), for which several solution procedures are developed.

This approach has some major advantages. First of all, it not only gives a decision for the current period ($\mathbf{x} = X_1$), but also a replenishment schedule for the next $N - 1$ periods. Secondly, all kind of information about future demand can be incorporated in the demand forecasts d_t. For example, orders that are placed in advance or seasonal demand. And finally, the procedure is very natural and requires no complex solution procedures.

When we look at the resulting strategy and the replenishment schedule for the next periods more carefully, than we notice some strange behaviour. The first drawback of this approach is that the choice of the planning horizon in combination of the solution method can influence the decision for the first period. Or in other words, the resulting strategy depends not only on the cost and demand structure, but also on how the deterministic problem is solved. An increase of the planning horizon can lead to an increase in the amount ordered in the first period. Suppose the deterministic problem is also solved for $I_0 \geq d_1$, that is $\mathbf{s} \geq s^-$. In advance, it is known that $X_1 = 0$. So, solving the deterministic problem only gives adjusted replenishments for next periods. But, even when demand is as expected these replenishment can differ from the last scheduled replenishments, in timing as well as in size. This effect is entirely due to the solution method in combination with the planning horizon.

Secondly, regardless of the solution method the resulting strategy is of the following type: if $\mathbf{s} < s^-$ then order $\mathbf{x} = (s^- - \mathbf{s}) + (r - 1)\mu$ with r denoting the scheduled reorder

interval, otherwise do not order. Only the reorder interval depends on the chosen solution method. The next scheduled replenishments are multiples of μ and are often of size $r\mu$. However, with (1) the expected reorder interval for this (s^-, S)-strategy $(S = s^- + (r-1)\mu)$ is only integral for $c^2 = 1$. So, in general the expected reorder interval is not equal to the scheduled reorder interval r. For distribution functions with small coefficient of variation $(c^2 < 1)$, the next replenishment is frequently triggered earlier than scheduled and the size of the replenishment is smaller than scheduled. As a consequence, schedules are frequently adjusted and are therefore unsuitable for further planning activities.

2.2 Reducing system nervousness

Both described effects can be considered as system nervousness. The first effect can be eliminated by choosing a proper solution method. With this respect, the optimal solution method is no longer suitable, but one can choose the Silver-Meal heuristic instead. The Wagner-Whitin procedure can only be used when the formulation the deterministic problem is changed. In van der Sluis (1993) we show that allowing for positive ending inventory in period N leads to a stable procedure. The essence is that the deterministic model only considers total cost in N periods, while the solution is implemented in a system that is focused on minimising long run average cost.

The second disadvantage of the rolling horizon procedure, the difference in expected and scheduled reorder interval, can be eliminated by adjusting the decision for the current period based on cost and stability considerations. One approach is to change order-up-to levels by specifying the deterministic demand in a different way. Note that the reorder interval r may change as well. Or alternatively, given the deterministic schedule X (e.g. order every r periods), the decision for the first period can be adjusted before implementing it in the stochastic system. With (1) is easy to see that an increase of X_1 by $(1 - c^2)\mu/2$ leads to an integral expected reorder interval, that is equal to the scheduled interval. The long run average cost of both the standard approach and the adjusted approach is within 2% of the cost of an optimal strategy. Based on cost considerations both approaches are suitable, but adjusting the order gives a more stable replenishment schedule.

3 Multi-product systems

The rolling horizon approach is often the only suitable approach for complex inventory systems. For example, multi-product systems where co-ordination of replenishments makes sense due to the cost structure: a joint setup cost k_0 is incurred when at least one product is ordered and an individual setup cost k_i for each product ordered. The joint setup cost can be seen as cost of the mode of transpor-tation, independent of the quantity ordered. It is well known that for such a cost structure co-ordinated replenishment leads to lower costs compared to independent replenishment of each product.

For deterministic problems with constant demand simple *cycle policies* are proposed. Such policies prescribe replenishments at constant intervals of time and the quantity ordered of each product is sufficient to last for an integer multiple of this *basic interval.* The structure of an optimal strategy can be very complex, even for systems with only a few products. When the basic interval is restricted to integral value, an optimal cycle policy can easily be found.

For deterministic problems with time-varying demand Arkin et al. (1989) proved that this joint replenishment problem is NP-complete. Several *single-pass* heuristics are suggested. These heuristics determine replenishment periods and lot sizes on a *period by period* basis.

The solution for an n-period problem are used to obtain a solution for the $(n+1)$-period problem by increasing the last replenishment or by triggering a new replenishment in period $n+1$. The heuristics schedule replenish-ments as if the problem continues forever. An important advantage of these methods is the linear running time in the number of planning periods and in the number of products. Furthermore, the decision for the first period is independent of the choice of the planning horizon. The quality of the solution, however, may be poor.

Better solutions can be found by using *multi-pass* heuristics. Given a replenishment schedule for N periods the schedule is iteratively improved by changing the timing of joint replenishments. For a large set of test problems the solutions found by multi-pass heuristics are within 2% of the minimal cost and for more than 80% of the test problems an optimal solution is found. However, the decision for the first period depends on the choice of the planning horizon.

For stochastic systems under continuous review research has been focused on finding an (S, c, s)-strategy, the so-called *can-order strategy*, first introduced by Balinthy (1964). Such a strategy is characterised by three parameters s_i, c_i and S_i for each product i with $s_i \leq c_i \leq S_i$. When the inventory level of a product falls to or below its *reorder level* s_i an order is placed and every other product with inventory level at or below its *can-order level* c_i is included. All products included in the order are replenished up to their *order-up-to level* S_i. However, an optimal strategy within the class of (S, c, s)-strategies does not have to be the one that minimises the average costs. In general finding this cost minimising strategy is almost impossible. In case of periodic review only for problems with two products an optimal strategy can be found. For larger problems no algorithm is known to obtain an optimal overall strategy or an optimal can-order strategy.

We have implemented a rolling horizon procedure in the same way as for the single-product systems. That means, for each product i the deterministic demand is defined as $d_{i1} = s_i^-$ and $d_{it} = \mu_i$ for $t > 1$, starting inventory I_{i0} is set at s_i, the economic inventory and decisions x_i are obtained by solving a joint replenishment problem using a single-pass heuristic. Simulating a stochastic system using this approach shows that the next replenish-ment is frequently triggered earlier than scheduled. This is no surprise. Suppose the second replenishment is scheduled in period r and m products are included. When for each product α is the probability of triggering a replenishment before period r, then the probability that the system replenishes in or after period r is $(1-\alpha)^m$. With for instance $\alpha = 0.2$ and $m = 10$, $(1-\alpha)^m = 0.1$. So, this probability is already small for reasonable values of α and m and replenishing in period $r-1$ or $r-2$ is more likely.

Early replenishments often cause all scheduled replenishments to be put forward and therefore (joint) setup costs are more frequently incurred than scheduled. The cost of an early replenishment, referred to as nervousness cost, can be estimated. Furthermore, the probability of an early replenishment can be decreased by increasing the current replen-ishment. This leads to higher inventory until the next scheduled replenishment, but lower nervousness cost. We developed an iterative procedure to increase the current replenishment based on a trade off between additional inventory cost and nervousness cost.

Computational results show that when in a rolling-horizon environment the adjustment procedure is used the long-run average costs are reduced considerably. Cost reductions between 1.8% and 7.6% have been obtained. In case of low joint setup cost the approach even outperforms the can-order strategy. For problems with higher joint setup cost there is a marginal cost difference between the two approaches, except for a few instances the can-order strategy gives better results, however, the cost difference is less than 2%.

The poor performance of the standard rolling-horizon approach is caused by the large number of early replenishments. In same cases more than 90% of all replenishments were put forward. This number reduced to 23% when the adjustment procedure was used. As a consequence the total number of replenishments per 100 periods for this specific instance reduced from 48 to 36 (in both cases 33 scheduled). This behavior is observed for all test problems; a reduction of the number of replenish-ments between 15% and 30%. So we can conclude that the adjustment procedure reduces the number of early replenishments and therefore reduces the system nervousness.

4 Concluding remarks

Applying a rolling horizon procedure in a stochastic system requires some caution. Using the standard approach, that is, using expected demand as demand forecast, leads to a nervous system. Scheduled reorder intervals are seldom realised and also order sizes are frequently adjusted. The schedules are therefore inconvenient for further planning. It is important to examine the cause of this nervousness. First of all, a solution method should be used that is independent of the planning horizon. This avoids build-in nervousness. Secondly, the consequences of an early replenishment should be determined. Often early replenishments lead to unused inventory and are almost never compensated by late replenishments. Finally, its is useful to have insight in the structure of the solution for deterministic problems with constant demand.

In my thesis this approach is also applied in multi-retailer systems. These systems are related to the multi-product systems, in the sense that co-ordination of the replenishments of the retailers leads to lower costs at the central depot. An additional complicating factor analysing these systems is the organisation of the distribution system: a push or pull system. In the former, retailers order independently at the depot. The depot only has insight in the inventory levels at all retailers and uses this information for its own replenishments. In the latter, replenishment decisions for the retailers are made at the depot simultaneously. Also, safety stock calculations for this type of system are more complicated. The main conclusion for these systems is that increasing the decision obtained by a standard rolling horizon procedure leads to lower average cost and less nervous systems and hence more useful replenishment schedules.

References

ARKIN, E., D. JONEJA AND R. ROUNDY (1989). Computational complexity of uncapacitated multi-echelon production planning problems. *Operations Research Letters* **8**, 61-66.

AXSÄTER, S., CH. SCHNEEWEISS AND E. SILVER, eds. (1986). Multi-Stage Production Planning and Inventory Control, *Springer Verlag, Berlin.*

BALINTHY, J.L. (1964). On a Basic Class of Multi-Item Inventory Problems. *Management Science* **10**, 287-297.

CARLSON, R.C, J.V. JUCKER AND D.H. KROPP (1979). Less Nervous MRP-systems: A Dynamic Economic Lot-Sizing Approach. *Management Science* **25**, 754-761.

JENSEN, T. (1992). Measurement of Planning Stability in Material Coordination Systems. In *Proceedings Seventh International Working Seminar on Production Economics*, Igls/Innsbruck.

SCHNEEWEISS, CH. (1986). Some Modelling Theoretic Remarks on Multi-Stage Production Planning. In Axsäter et al. (1986), 1-8.

SLUIS, E. VAN DER (1993). Heuristics for Complex Inventory Systems, *Thesis Publishers, Amsterdam.*

TIJMS, H.C. (1986). Stochastic Modelling and Analysis: A Computational Approach, *Wiley, New York.*

WAGNER, H.M. AND T.M. WHITIN (1958). Dynamic version of the Economic Lot Size Model. *Management Science* **5**, 89-96.

Maintenance policies for complex systems

S.G. Vanneste
Shell International Oil Products
Amsterdam
E-mail: s.g.vanneste@siop.shell.nl

1. Introduction

The reliability of production systems and the quality of products have become important issues in today's industry. It is realized that quality and reliability directly affect the competitiveness of the firm and that a good preventive maintenance program can have a major impact on them. While a few decades ago, maintenance was considered a necessary evil, today it has become an important business function. Factors that contributed to this growing importance are the increased levels of automation, the increased complexity of products and equipment, the trend towards Japanese management philosophies and the need for a stringent cost control.

In many firms, as well as in public works (roads, bridges, railways) and defense, the maintenance budget accounts for a significant portion of the total operating costs, thus necessitating an effective and efficient allocation. Moreover, the consequences of breakdowns in terms of opportunity cost (e.g. production losses) may be very large and even surpass the maintenance budget in magnitude.

In this thesis we analyse preventive maintenance policies for complex systems. For many systems, the costs of breakdowns are much higher than the maintenance or replacement costs in case the unit has not yet failed. In view of the fact that the risk of failure usually increases with age or use, one may wonder whether, and at what age or condition, it is economically justified to replace a particular component or group of components preventively. As it turns out that the optimal policy may have a fairly irregular structure, which makes it difficult to implement, we focus attention on policies which are easy to characterize. We analyse the (near-)optimality of this type of policies and develop algorithms and approximations to evaluate them and to find the best policy, within a selected class. In contrast with the early literature, which considered maintenance optimisation issues for single unit systems, we consider systems which are composed of multiple components, whether or not in relation to the production environment.

Along with the analysis of new and existing models, we develop an analytical framework to deal with practical maintenance situations. This framework includes models and serves as a concept for using them in practice. The framework is applied to several industrial cases.

2. The classical age replacement model

The classical age replacement model serves as a reference point for the models presented in this thesis. Consider a piece of equipment which is subject to (stochastic) failures. The

[1] This is a summary of a PhD thesis with the same title

unit can be replaced by a new one, either preventively or correctively (upon failure). The following assumptions are made:

1. The equipment is operating continuously during its lifetime
2. Repair times are negligible
3. The planning horizon is infinite
4. Every new piece of equipment has identical characteristics
5. Only replacement costs are considered
6. The objective is to minimize the long-run costs of replacement

Furthermore we assume that the lifetime of the unit is a random variable with known distribution function $F(\cdot)$. The following replacement policy is applied: the unit is replaced t units of time after the last replacement or upon failure, whichever occurs first. The cost of preventive replacement is c_p and corrective replacement c_f.

Define a cycle as the time between two consecutive replacements (either preventive or corrective). It follows from renewal theory that the expected costs per unit time equal the expected cost per cycle divided by the expected length of a cycle. Elaborating the last two quantities yields the following formula for the average costs $g(t)$:

$$g(t) = \frac{c_p(1 - F(t)) + c_f F(t)}{\int_0^t (1 - F(s))ds}.$$

The general shape of $g(t)$ as a function of t is shown in Figure 1. The function is decreasing on the left hand side of the minimum and increasing on the right hand side. A function with this property is called a unimodal function. The minimum of the function yields the optimal preventive age-limit. By imposing an appropriate cost structure, the model can also be used to study availability issues.

This model gives a simple and lucid description of a typical situation found in replacement. The model is fairly general, robust and easy to use. Due to these properties, it is a successful and widely used model.

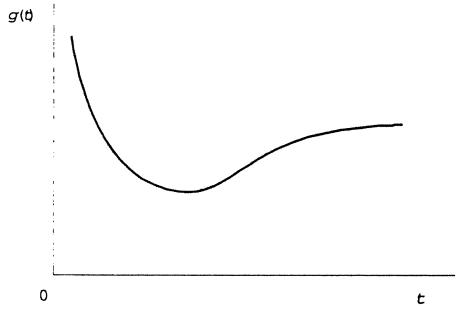


Figure 1.1 Shape of the average cost function

In the thesis we study systems consisting of one or several identical components in a series configuration. The components are subject to failures, and it is assumed that the

system cannot work satisfactorily if one of the components has failed. Therefore, a failure is followed by a corrective maintenance activity (CM), which restores the state of the failed component. To avoid a frequent occurrence of a failure there is the option of performing preventive maintenance (which is assumed to be less costly). The additional costs associated with preventive maintenance (PM) have to be balanced with the cost savings related to the prevented failures. When the system is composed of several components, additional savings may result from a so-called group (or system) replacement. It is usually cheaper to replace (or maintain) all components at one time than separately. The probabilistic nature of the failure behaviour under a certain preventive maintenance policy calls for a mathematical model, so that we can analyse critical performance measures and find the optimal maintenance policy in terms of minimal long term average costs.

When we speak of complex technical systems, composed of several parts, we can think of the maintenance of a highway or railway, the piers of a bridge or a couple of generators, providing a factory or hospital with energy, but also on critical components of equipment or vehicles (e.g. a series of bolts, see Chapter 6). The major characteristics are that the components deteriorate over time (in the sense that the probability of failure increases), and that there are costs associated with PM and CM. We use the terms 'unit' and 'component' interchangeably.

As an example, consider the maintenance of a particular part of the road. In practice it is convenient to divide this part up into several sections, which correspond to components in our system. When a serious crack in the asphalt arises in a particular section, we can speak of a failure, which calls for a reconstruction of that section. The probability of a crack increases with the age of the asphalt. Therefore, by timely adding a new layer, which is less costly than a complete reconstruction, the bursting can be prevented (this can be regarded as a preventive maintenance action in our model).

Thus we need information on the cost associated with the maintenance activity and on the deterioration process. Three cost parameters are required: the cost of preventive and corrective maintenance on component level and the cost of a group replacement. It is assumed that PM is less costly than CM and that a group replacement is less costly than the number of components times the cost of a separate replacement. With respect to the deterioration process we distinguish two cases: the age case and the condition case. The age case refers to the situation that the age of the component determines the probability of failure. In this case we need the lifetime distribution of the time to failure for our model. In the condition case we assume that the component can be in a number of states, representing its condition. From a state we may either move to a higher state or to the breakdown state (failure), after a stochastic amount of time. More specifically, we assume that the sojourn times in each state are exponentially distributed. What we then need is, for each state, the mean sojourn time in that state, and the failure rate associated with that state. We note that for our purpose, these two cases are mathematically closely related. The deterioration of different components is supposed to be stochastically independent.

3. Single-unit systems

In Chapter 2 we formulate a generalization of the standard age-replacement model, which we presented in the previous section. In practice, the repair facility is often not available at any time. Therefore, we consider the case that the repairs are restricted to opportunities generated by a Poisson process and compare this to the situation that the repair facility is continuously available. Furthermore, we allow for a very general type of repair. The repair times have a general distribution, and the repair is allowed to be imperfect, that is

the state after maintenance is not necessarily perfect, but may be an inferior state with a certain probability. Thus, we account for possible human errors or other factors that may lead to the (near) occurrence of failures as a result of PM.

The assumption of imperfect repair in combination with a repair facility which is not continuously available, gives rise to an interesting phenomenon. Contrary to intuition, the age limit for preventive maintenance under opportunistic maintenance is possibly *higher* than in case the repair facility is continuously available.

We model the system by a general Markov chain. Although we formulate the model as a single-unit model, the system may physically consist of several parts. We show for example how a two-component standby system can be analysed with this model. The general representation implies that the results apply as well to the case of condition-based deterioration. It is assumed that the unit can be in a finite number of states which either represent (possibly discretized) ages or conditions.

As in the standard age-replacement model, the decision problem can be formulated as follows: in which state (or age) should we start preventive maintenance? From a theoretical analysis of the model it follows that the optimal maintenance policy can be characterized by a single number (called the control limit), denoting the level above which PM is required, and below which the system is left alone. Moreover, it can be shown that the long-run average costs under a fixed control limit policy as a function of the control limit are unimodal (cf. Figure 1).

We present efficient procedures to calculate the average cost under a fixed policy and to compute the optimal policy. These procedures are easily implemented on a PC and yield quick solutions.

4. Two-component systems

In order to analyse multi-component systems we drop our generalizations for the single-unit case. Like in the standard age-replacement case, we assume that the repair times are negligible, that repairs are perfect and that the repair facility is continuously available. Also, our analysis of the two-component system in Chapter 3 is restricted to the age-case. For each age-combination we now have four options: replace component 1, replace 2, replace both and leave the system alone. One may wonder whether the optimal policy still has a nice structure, which is easy to characterize. This appears not to be the case. The optimal policy may have a rather complicated structure. An example of a counter-intuitive property is the following: it may happen that when component 1 has age 2, say, and component 2 age 6, it is optimal to replace component 2, but when both units are one time-unit older (3 and 7, respectively) it is optimal to replace none.

However, numerical investigations show that there is a class of policies, which is easy to characterize, and at the same time close to optimality. This is the class of so-called (n, N)- policies, which are a kind of generalized control-limit policies. A (n, N)-policy prescribes to replace a component preventively as soon as it reaches the age N, but also when the age n ($\leq N$) has been reached at an epoch at which the other component is due to replacement (because of failure or having reached the preventive age-limit N) The latter type of replacement is called an opportunistic replacement. From extensive numerical investigations with various lifetime-distributions and cost parameters, we may conclude that the best (n, N)-policy is in general very close to optimality (less than 1% deviation in average cost). For this reason, we further analysed the class of (n, N)-policies and developed an efficient procedure to evaluate the average cost under a fixed (n, N)-policy. This procedure was used as a building block in a heuristic procedure to find the "optimal" values of n and N.

5. Multi-component systems.

From the perspective of maintenance actions a component can be in four possible states. One is the failed state, which requires a corrective replacement. Another is the bad state, which calls for a preventive replacement. Yet another is the good state, where no maintenance is required. And finally a component can be in a doubtful (or intermediate) state, which does not call for a preventive replacement on itself, but if some maintenance activity is performed on other components in the system, it may be worthwhile to maintain the current one as well (since it is relatively cheap). This idea is used in our analysis of a system composed of arbitrarily many components.

One of the group replacement policies that we consider is the following: the whole system is replaced if, upon replacement of a single component, there are more than K components in the doubtful state, where K is a control parameter.

First we consider the case, that the transition behaviour among the four aggregated states is comparable to the behaviour of a condition based deterioration process as it was described in the beginning of this subsection. That is, the sojourn times in the states good and intermediate are exponentially distributed, and we can move from good to doubtful and from good to down, and from intermediate to bad and from intermediate to down (the states bad and down are instantaneous states, since a replacement is carried out upon entrance, and the required time is negligible). For this case, we derive explicit and easily computable formulas for the long run average cost per unit time, the fraction of preventive and corrective replacements as well as the mean time between two system replacements.

Approximations for these quantities are derived for the case that the components deteriorate with age. To match the ages with the aggregated states, we introduce two critical parameters r and R, and associate an age between 0 and r with good, between r and R with doubtful, and above R with bad. Combining the results for the exponential model with heuristics, decomposition arguments and results for the one-component system, we are able to develop an approximation for the average cost under a policy characterized by the parameters r, R and K. The resulting approximation is insightful and easily implementable on a PC, yielding quick solutions with an accuracy up to 5%.

6. Interaction with production

In production situations the preventive maintenance does not only depend on the deterioration of the machines, but also on the state of the production. In Chapter 2 we therefore allowed that repairs were performed (or started) at opportunities. Here we go a step further. We consider a system which is composed of a machine followed by a buffer and which is part of a flow-shop production line. The buffer is used to build up safety-stocks in case the machine is halted. The content of the buffer can be used as indicator whether or not to start preventive maintenance on a given machine, because it determines to what extent a machine stop can be covered by delivering from the buffer. We analyse the optimal policy as a function of both the age of the machine and the content of the buffer and present an efficient algorithm to compute several performance measures for a class of sub-optimal policies.

7. A decision making and analysis framework

In order to use models in a practical situation one needs an analytical framework in which they can take part. Such a framework is discussed in Chapter 6. In that chapter we discuss

a systematic framework to improve the maintenance function. We present an eight phase appproach in which effectiveness and efficiency are improved in a continual improvement program. Models are used in the efficiency phase. Furthermore we show how information procedures fit into the concept and discuss an application of the concepts, including some models, at a concrete factory.

Part III

Game Theory

Game Theory of the Past Decade, the Contribution of the Netherlands

S.H. Tijs[a] and O.J. Vrieze[b]
[a] Department of Econometrics
Tilburg University
E-mail: tijs@kub.nl

[b] Department of Mathematics
University of Maastricht
E-mail: oj.vrieze@math.unimaas.nl

1 Introduction

Game theory is a mathematical theory. Its aim is to model decision situations where two or more parties have strategic interaction, and to develop and to analyse solution concepts. A party can be a person, a firm, a political party, a nation, etc.. Many of the game theoretical models stem from situations of economic interaction. Both at the micro-economic level and at the macro-economic level abundant examples can be found where strategic behavior is essential. Game theory delivers the tools, comprising ideas about what is called a solution and about how the decision situation should be strategically played by the parties.
It is generally acknowledged that game theory started with the fundamental book 'Theory of Games and Economic Behavior' (1944) by John von Neumann and Oskar Morgenstern. Their main motivation comes from the following phrase (p. 31).

> We wish to find the mathematically complete principles which define rational behavior for the participants in a social economy and to derive from them the general characteristics of that behavior.

Already from the beginning of the development of game theory it was clear that two main branches should be discerned, namely noncooperative games and cooperative games. Obviously in noncooperative games the players have not the possibility or no incentive to cooperate with each other in order to derive joint profit. Noncooperative game theoretical models presume that the strategical moves of the players are taken simultaneously and independently. However each players outcome depends on the collection of strategical moves of all the players.
In cooperative games the assumption is made that any subset of players can form a coalition which has a certain strength or worth. The main question in cooperative game theory concerns a fair division of the worth of the grand coalition, taking into account the strength of all the subcoalitions.
Not long after the appearing of the book of Von Neumann and Morgenstern, the main lines of development of game theory were set by four recent Nobel prize winners, namely John Nash (1994), Reinhard Selten (1994), John Harsanyi (1994) and William Vickrey (1996) of

which the last one died from a heart attaque a few days after the announcement. Nash (1950) introduced the concept of equilibrium point as a solution method for noncooperative games and still nowadays Nash-equilibria are recognized as the most appropriate solution idea for such games. Selten (1975) extended the equilibrium concept to multi-move decision situations. He argued that it makes sense not only to anticipate on rational behavior of the opponents but on irrational behavior as well. He introduced the concepts of perfect equilibria and subgame perfectness which can be considered as refinement concepts with respect to Nash-equilibria. Harsanyi (1967), motivated by problems of arms control, proposed models for decision situations with incomplete information. Before that the rules of the games included that all players know all the data of the game and games with incomplete information were considered to be unsolvable. Harsanyi introduced for games with incomplete information the concept of Bayesian-Nash equilibrium, a method based on statistical techniques. The present information-economy in which crucial incomplete information or knowledge is continuously updated can be seen as being formed by the work of Harsanyi. Finally Vickrey approached game theory from the economic side. He was very much interested in incentives related to asymmetric information, like public economies with as central question how far should a government influence or control economic markets. Though less mathematically rigorous, Vickry introduced many game theoretical concepts in a qualitative sense, some of them even before they were "discovered" by the mathematicians.
Besides the roles of these four Nobel prize winners one should not underestimate the historical contributions of other researchers of which Lloyd Shapley and Robert Aumann should be especially mentioned.

2 Developments in the Netherlands

Game theory in the Netherlands started with Stef Tijs. In 1975 he finished his Ph.D. Thesis "Semi-infinite and infinite matrix games and bimatrix games" at the Catholic University at Nijmegen. Already in 1987 about 15 researchers in game theory were affiliated to Dutch universities and 12 of them straightforwardly branched from Tijs. The state of the art of game theory in the Netherlands in 1987 can be found in a book dedicated to the 50-th anniversary of Tijs in which contributions are presented by his students (Peters and Vrieze(1987)).
After 1987 a turbulent development in game theory took place. The number of workers in the field largely increased, not only in the Netherlands but all over the world. Partly this is due to a revolution in economic theory where more and more game theory is accepted as delivering the toolbox for explanations of economic behavior. Of course, the above mentioned Nobel prize to game theorists increased the status of the field. Presently in the Netherlands more than 30 researchers, among which 6 at a professor level, in game theory are active and moreover one can argue whether certain researchers in economics are in fact game theoretists. In the past 10 years more than 20 Ph.D. Theses in game theory appeared in the Netherlands. In the next section most of them will be mentioned. Another 8 till 10 dissertations will be finished in the near future. Numerous papers appeared in respected international journals.
At the organisational side we can mention the editorial activities for international journals by Eric van Damme, Hans Peters, Stef Tijs and Koos Vrieze. Further, Stef Tijs is the editor of a Kluwer book series on Theory and Decisions in which 15 books appeared during the past decade, either directly concentrated on fields in game theory or related via economic decision situations. Recently 3 international conferences were organized (Tilburg, 1991 (Van Damme), Maastricht, 1996 (Peters), Twente 1996 (Driessen, Hoede, Faigle)) and next year a fourth one is planned (Maastricht, 1998 (Vrieze)). Also since 1982 a monthly game theory

seminar is organized by Tijs.

3 Developments in Noncooperative Games

A noncooperative two-person game can be characterized by a four-tuple $< A, B, r_1, r_2 >$, where A and B are the nonempty strategy sets of player 1 and player 2 respectively and where $r_k : A \times B \rightarrow IR$ is the payoff function of player $k, k = 1, 2$. Generally, the players are allowed to randomize their strategies, that is they may select a strategy according to a probability measure on their available set of strategies. When $|A|$ is finite, such a probability measure is just a probability vector in a finite dimensional Euclidian space and when $|A|$ is infinite the probability measure is assumed to be defined on a given probability space. Let x denote a randomized strategy for player 1 and let y denote a randomized strategy for player 2. $r_k(x, y)$ will denote the expectation of $r_k(a, b)$ with respect to x and y.
The generally accepted solution method for non-cooperative games is the Nash-equilibrium. A Nash-equilibrium is defined as a pair of strategies (x, y), such that none of the players has a unilateral incentive for a deviation:

$$\begin{aligned} r_1(x, y) &\geq r_1(\tilde{x}, y), \forall \tilde{x} \\ r_2(x, y) &\geq r_2(x, \tilde{y}), \forall \tilde{y} \end{aligned}$$

In the early eighties the structure of the set of Nash equilibria for bimatrix games (the case where both (A) and (B) are finite) were extensively studied by Jansen (1981, 1987). Van Damme studied games in extensive form and bimatrix games and he succeeded in connecting most of the refinement concepts of Nash-equilibria (Van Damme, 1987). In the thesis of Borm (1990) some remaining gaps in the properties of the refinement concepts were filled and furthermore he derived specific results in case $|A| = 2$ and $|B| = n \in IN$. In the thesis of Jurg (1993) a thorough study is done for perfect and proper equilibria and refinements thereoff. Recently the long lasting question whether any bimatrix game possesses a quasi-strict equilibrium (already suggested by Harsanyi in 1973) was solved by Norde (1994).
Kohlberg and Mertens (1986) introduced a stability concept which leads to a selection of a set of equilibria which satisfy certain properties. Vermeulen, in his thesis (1996), studied extensively this stability concept and several variations of the original definition. He succeeded in defining the essential properties of the set-valued solution concepts for non-cooperative games.
It appears as if the attemps to refine the set of equilibria to one single point have temporarily reached an endpoint by the introduction of the stable set. Peleg and Tijs (1996) introduced the consistency principle for games in strategic form. Norde, Potters, Reijnierse and Vermeulen (1996) were able to prove that there do not exist consistent equilibrium selections and consistent refinements. It is by no means overdone to say that in the past 15 years Dutch researchers have had great influence on the developments of the general theory of equilibria of noncooperatieve-games and many of the characterizing properties were elucidated in our own surroundings.
In a thesis, which opens new horizons, Hurkens (1995) relaxed the rules of a noncooperative game situation. He allowed the players to make strategical moves like commitments and communications. He considered sets of solutions instead of one point solutions. Among other results he showed that so-called persistent sets or "closed under rational behaviour" sets are appropriate for games in an evolutionary context. Further he derived unexpected results where he allowed players to burn part of their fortune, which appeared to be equivalent of buying communication lines at a high prize.

4 Developments in Stochastic Games

Stochastic games are dynamic games that evolve along a discrete time path. At any decision moment the players have to play a game identical with a static noncooperative game with the extension that the strategy choices determine a probability distribution according to which the next state (again a noncooperatieve game) is chosen, etc.. Usually the infinite horizon model is studied and the evaluation criteria for an infinite stream of payoffs normally concentrate on discounted rewards or limiting average rewards defined respectively as

$$v_k^{\beta}(x,y) = \sum_{n=0}^{\infty} \beta^n r_k^n(x,y) \quad , \quad \beta \in (0,1)$$

$$a_k(x,y) = \liminf_{N\to\infty} \frac{1}{N} \sum_{n=0}^{N} r_k^n(x,y)$$

One should be aware of the fact that here x and y, rather than one shot strategies, denote rules that prescribe the players how to play the game along the infinite time path.
The first Dutch thesis came from Vrieze (1983) in which a complete study of the zero-sum case ($r_1 + r_2 = 0$) was elaborated. For non-zero sum stochastic games, especially for the limiting average criterion, the existence of Nash-equilibria is not yet known,and no doubt presently this is one of the most challenging open problems in game theory. In the thesis of Thuijsman (1989) a new idea based on threat stretegies is introduced. That idea is later on exploited and until now all existence results for Nash-equilibria for subclasses of stochastic games can be founded on the concepts of threat strategies. The fundamental result in this ares can be found in a paper of Vrieze and Thuijsman (1989). Stochastic games can be seen as extension of Markov Decision Problems to the competitive case. This viewpoint has recently been worked out in the first book on stochastic games by Filar and Vrieze (1997).
A promising new branche in the area of stochastic games is introduced in Joosten (1996), in which he studies stochastic games where the game parameters change over time. For instance a game with vanishing action is defined as a game where a player will forget an action in case he does not make use of it for a sufficiently long time. In Joosten, this one and related stochastic games with changing parameters are solved.
Passchier (1996) explored the application of stochastic games to light traffic control in which he knew to exploit optimally the similarities between stochastic games and Markov Decision Problems.

5 Developments in Cooperative Games

The theory of cooperative games is based on the idea of a characteristic function. Let N be the set of players, then any subset $S \subset N$ is called a coalition and a characteristic function is a function v:

$$v : 2^N \to IR,$$

that is, to every coalition S, a real number is associated reflecting the strength of that coalition. The main goal of the theory of cooperative games is to define and motivate solution concepts in which $v(N)$, the worth of the grand coalition, is divided among the players in a relatively fair way. Interesting work in this area is done by Driessen published in his book (1988) in which he connects several of the existing solution concepts and studies the properties of a new one, the τ-value, introduced by Tijs. In the thesis of Derks (1991) it

was shown how certain classes of games can be seen to be polyhedral cones, which enabled him to give easy proofs and more understanding of the different properties for many classes of cooperative games.
In the thesis of Otten (1995) several characterizations of solution concepts are given. Some of them apply to NTU-games, where the "utility" is no longer transferable between the players.
Finally, we want to mention the work of Curiel. In cooperation with Potters and Tijs she introduced the idea of combinatorial games. In combinatorial optimization several classical problems like travelling salesman, minimum cost spanning tree, assignment problems, etc., are studied in a one-person sense, that is, there is only one optimizer. In practice often more parties at least jointly influence the interaction and the outcome. When the worth of a coalition has to be computed, this coalition can be considered to be one unified player that has to optimize a certain type of combinatorial optimization problem. In the thesis of Curiel (1988) this idea is applied for the first time in a systematic way. In the next section we will see that several studies followed this approach. See also Curiel (1997).
Nowadays it is widely recognized that cooperative game theory has a broad range of application. Especially with respect to schemes of costs allocation cooperative game theory offers convincing tools as was already shown by Littlechild (?). More recently, applications to allocation of reduction measures in order to abate the environmental pollution were worked out (Van der Ploeg and De Zeeuw (1994), Tol (1996)). Further, in a paper of Van den Nouweland, they report about profit allocation in games constructed from cooperation of phoning in planes.
In the classical cooperative game model it is assumed that there is unrestricted communication between the players. Evidently, in practice this is not always the case and recently this phenomenon is tried to express in game theoretical terms in the thesis of Van den Nouweland (1993). She modelled communication restriction by a communication graph. As solution concepts she proposed the Myerson value and the position value. In general these two values are hard to compute, however Van den Nouweland developed elegant algoritm for several subclasses of communication games.
In the algorithmic sphere the work of Reijnierse (1995) should be mentioned. He has developed methods to compute the nucleolus. The emphasis of his work lies in TU-games and besides the algorithmic results he defined properties that make solution concepts unique and he showed all kind of relations between the different solutions.

6 Developments in Combinatorial Games

In a short period of time four dissertations appeared around this subject. In the preceding section we already motivated this area. In Kuipers (1994) numerous classical combinatorial optimization problems are studied in a game setting. For instance the travelling salesman games (especially routing games), the minimum cost spanning tree game (represented as an information graph game) and assignment games. The main results concern existence and properties of the anti-core and anti-nucleolus. "Anti" because the characteristic function has the meaning of costs and obviously players prefer the lesser costs. In the work of Kuipers the properties of the underlying combinatorial structure were fully exploited in computational schemes or other aspects of the solution concepts. Aarts (1994) mainly concentrated on minimum cost spanning tree games. For several special cases he derived new and useful results. Also Veltkamp (1995) concentrated on minimum cost spanning tree games. However in his thesis the emphasis lies on the axiomatic characterization of different solution concepts for different instances of the games. Obviously, a good characterization shows whether a

concept is applicable in the light of the combinatorial properties of the decision problem. Hamers (1995) explored sequencing games in which several agents want to use one machine that can only serve one agent at a time. The obvious question concerns the sequence of the agents that determine the order in which they are served, taking into account all kind of cost structures. Hamers compared several solution methods and derived many new properties. In 1993 a special issue of ZOR (now MMOR) was edited by Borm and Tijs, dealing with games, graphs and O.R..

7 Developments in Bargaining Games

Bargaining theory can be seen as part of cooperative game theory, where the utility is not transferable. When N players are involved an N-dimensional set is defined an element of which has to be appointed, where component k is associated the player $k, k = 1, 2, \ldots, N$. In the thesis of Peters (1986) the relations between several solution concepts for bargaining games were studied. In his book, Peters (1992) worked out the axiomatic characterizations for the different solution concepts, where the axioms refer to intuitive appealing properties. Especially he studied non-symmetrical extensions of the classical solutions like Nash solutions and Kalai-Smorodinski solutions.
Houba (1994) studied a few extensions of the classical bargaining model. For a three player situation the rules of the game allow just two players to cooperate. He derived many axiomatic characterizations for solutions of such games. Further he studied bargaining games motivated by policy making decision situations. The main feature of this type of games concerns the fact that the players can influence the outcome in case of a disagreement leading to additional strategical aspects.

8 Developments in Social Choice Theory

Social Choice is about making decision in a societal surrounding. The preference relations among the alternatives of the players determine the choice behavior. In Wakker (1986, 1989) the main assumption concerns the fact that the set of alternatives can be structured as a cartesian product. Any component refers to a specific aspect of the alternative. Wakker obtains many mathematical properties of this type of decision problems. In Storcken (1989) the ordering of the alternatives reflected in the preference relations, stand central. His goal is to derive possibility theorems for the different models related to different assumptions on the orderings. In fact, his work can be considered as the first cohesive approach for the study of possibility theorems. In Van der Stel (1993) voting schemes are studied in which all players call one alternative and next a voting rule determines the alternative chosen. He derives many new results with respect to the concept of strategy-proof choices, which are choices that are non-manipulative. Finally in Otten (1995) effectivity functions for social choices are studied. Effectivity functions define sets of alternatives that are reachable for coalitions of players in the sense that they are able to veto alternatives outside this set. Many interesting results about effectivity functions can be found in Otten.
Monsuur (1994) studied choice problems, where alternatives are ordered by pairwise comparison with respect to importance, preference, dominance, quality, etc.. The idea is to come to a priority list of the alternatives based on these pairwise comparisons. Further, Monsuur studied circularity measures for tournaments.

9 Conclusions

From the above it might be clear that the contribution of the Dutch researchers to the developments of game theory is overwhelming. Nearly all the fields of game theory are covered and many new and appealing ideas and results come from our country. Together with Israel and the United States, The Netherlands is one of the leading countries in game theory. Obviously, the researchers do the job and they need to possess the mathematical level to apply their creativity. It is in this respect that the activities of the National School of Operations Research (LNMB) tremendously contributed to the fundamental knowledge of our young bright newcomers. In the same spirit, we need to mention our counterpart in economics, the National School of Quantitative Economics (NAKE), that performs a similar excellent job as the LNMB. We are quite sure that without the continuous endeavours of these two Schools to maintain the outstanding level of our young researchers, the Netherlands would never have reached their present position.

10 References

Aarts H. (1994), *Minimum cost spanning tree games and set games, Ph.D. Thesis, University Twente.*

Borm P. (1990), On game theoretic models and solution concepts, Ph.D. Thesis, Catholic University Nijmegen.

Curiel I. (1988), *Cooperative game theory and applications, Ph.D. Thesis, Catholic University Nijmegen.*

Curiel I. (1997), Cooperative Game Theory and Applications, Kluwer Academic Publishers, Dordrecht.

Derks J. (1991), *On polyhedral cones of cooperative games, Ph.D. Thesis, Maastricht University.*

Driessen T. (1988), Cooperative games, solutions and applications, Kluwer Academic Publishers, Dordrecht.

Feltkamp V. (1995), *Cooperation in controlled network structures, Ph.D. Thesis, Tilburg University.*

Filar F. and O.J. Vrieze (1997), Competitive Markov Decision Processes, Springer Verlag, New York.

Hamers H. (1995), *Sequencing and Delivery Situations, a game theoretic approach, Ph.D. Thesis, Tilburg University.*

Harsanyi (1967), Games with incomplete information played by Baysian players, Managed Science **14**, *part I, II and III, p159-182, p320-334, p486-502.*

Houba H. (1994), Game theoretic models of bargaining, Ph.D. Thesis, Tilburg University.

Hurkens J.P.M. (1995), *Games, rules and solutions, Ph.D. Thesis, Tilburg University.*

Jansen M.J.M. (1981), Maximal Nash subsets for bimatrix games, Nav.Res. Log. Quart. **28**, *p147-151.*

Jansen M.J.M. (1987), Regular equilibrium points of bimatrix games, OR Spektrum **9**, *p87-92.*

Joosten R. (1996), Dynamics, equilibria and values, Ph.D. Thesis, Maastricht University.

Jurg P. (1993), *Some topics in the theory of bimatrix games, Ph.D. Thesis, Catholic University Nijmegen.*

Kohlberg E. and J.F. Mertens (1986), On Strategic stability of equilibria, Econometrica **54**, *p1003-1037.*

Kuipers J. (1994), Combinatorial methods in cooperative game theory, Ph.D. Thesis, Maastricht University.

Littlechild S.C. and G.F. Thompson (1977), Aircraft landing fees: A game theory approach, *Bell J. Economics* **8**, p186-204.

Nash J. (1950), Equilibrium points in n-person games, *Proc. Nat. Ac. of Science USA* **36**, p48-49.

Otten G.J. (1995), *On decision making in cooperative situations, Ph.D. Thesis, Tilburg University.*

Passchier O. (1996), The theory of Markov games and queueing control, Ph.D. Thesis, State University Leiden.

Peleg B. and S.H. Tijs (1996), The consistency principle for games in strategic form, *I.J.G.T* **25**, p13-34.

Peters H. (1986), *Bargaining game theory, Ph.D. Thesis, Catholic University Nijmegen.*

Peters H. (1992), Axiomatic bargaining game theory, Kluwer Academic Publishers, Dordrecht.

Peters H. and Vrieze O.J. (1987), Surveys in game theory and related topics, *CWI Tract* **39**, Amsterdam.

Reijnierse H. (1995), *Games, graphs and algorithm, Ph.D. Thesis, Catholic University Nijmegen.*

Selten R. (1975), Reexamination of the perfectness concept for equilibrium points in extensive games, I.J.G.T. **4**, *p25-55.*

Storcken T. (1989), Possibility theorems for social welfare functions, Ph.D. Thesis Tilburg University.

Thuijsman F. (1989), *Optimality and equilibria in stochastic games, Ph.D. Thesis, Maastricht University.*

Tol R. (1996), A Decision analytic treatise of the enhanced greenhouse effect, Ph.D. Thesis,

Free University Amsterdam.

Tijs S.H. (1975), *Semi-infinite and infinite matrix games and bimatrix games, Ph.D. Thesis, Catholic University Nijmegen.*

Van Damme E. (1987), Stability and perfection of Nash equilibria, Springer Verlag, Berlin.

Van den Nouweland (1993), *Games and graphs in economic situations, Ph.D. Thesis, Tilburg University.*

Van den Nouweland A., P. Borm, W. van Golstein Brouwers, R. Groot Bruinderink, S.H. Tijs (1996), A game theoritic approach to problems to telecommunicating, Management Science **42**, *p294-303.*

Van der Ploeg F. and A. de Zeeuw (1992), International aspects of pollution control, Environmental and Resource Economics **2**, *p117-139.*

Van der Stel H. (1993), Strategy-proof voting schemes on Euclidian spaces, Ph.D. Thesis, Maastricht University.

Vermeulen D. (1996), *Stability in non-cooperative game theory, Ph.D. Thesis, Catholic University Nijmegen.*

Von Neumann J. and O. Morgenstern (1944), Theory of Games and Economic Behavior, Princeton University Press, Princeton.

Vrieze O.J. (1983), *Stochastic games with finite state and action Spaces, Ph.D. Thesis, Free University Amsterdam.*

Vrieze O.J. and F. Thuijsman (1989), On equilibria in repeated games with absorbing states, I.J.G.T. **18**, *p293-310.*

Wakker P.P. (1986), Representations of choice functions, Ph.D. Thesis, Tilburg University.

Wakker P.P. (1989), *Additive representations of preferences, Kluwer Academic Publishers, Dordrecht.*

On "Dynamics, Equilibria, and Values"*

Reinoud Joosten
Maastricht University
E-mail: rjoosten@merit.unimaas.nl

The thesis forms a compilation of results on dynamics, evolution and (un)learning in economics and noncooperative game theory, as well as on a topic in cooperative game theory. The thesis consists of four chapters which can be read independently. Here, I will briefly describe the frameworks of the four chapters, the contributions in the theory, and possible connections between the chapters.

The chapter called 'Economic adjustment processes[1]', treats a model from mathematical economics in the theoretical framework of an exchange economy. An economic equilibrium is a special situation, where the demand equals the supply on the market of each commodity. The prices of the commodities play a prominent role in establishing both the level of the demand and the level of the supply of each commodity. The price vector associated with an economic equilibrium is called an equilibrium price vector. The following three questions are traditionally regarded as important. Does there exist an equilibrium price vector? Can an equilibrium price vector be reached by adjusting prices of commodities starting from a situation where the demand does not equal the supply for each commodity, and if so, how? Can an equilibrium price vector be computed or approximated, and if so, how? These questions motivate the contributions in this chapter.

Firstly, a new price adjustment process is presented, which reaches an equilibrium price vector from any given initial price vector in an arbitrary exchange economy. An excess demand function determines for each vector of prices the difference between the demand for and the supply of each commodity in the exchange economy. The so-called successive tâtonnement process of Walras [1874], the simultaneous tâtonnement process of Samuelson [1947] and large classes of generalizations thereoff (see e.g., Arrow & Hahn [1971]) are known to converge to an equilibrium price vector, if the excess demand function satisfies conditions such as Revealed Preferences or Gross Substitutability of all commodities. The problem is, however, that the class of excess demand functions is quite large and the conditions on the excess demand functions guaranteeing convergence to an equilibrium price vector of the price adjustment processes just mentioned, are generally regarded as rather strong.

A price adjustment process (or computational process in this framework) is called universally convergent if it converges to an equilibrium price vector for an *arbitrary* excess demand function. A price adjustment process is called globally convergent if it is universally convergent and if it may be started in an *arbitrary* price vector. The price adjustment process presented in this chapter is *globally convergent.* The path followed by the price adjustment process from the initial price vector to the equilibrium price vector, allows for an attractive economic interpretation. For any price vector on the path namely, it holds that if the price of a commodity is higher than its initial value, then the excess demand of this commodity is

*This is a summary of the PhD thesis Joosten [1996b] with the same title
[1]Chapter 2 is based on Joosten & Talman [1993] and Joosten & Talman [1995].

maximal, moreover if the price of a commodity is lower than its initial value, then the excess demand of this commodity is minimal.

Secondly, we contribute a new variable dimension restart algorithm to compute an equilibrium price vector in an arbitrary exchange economy. This computational process is globally convergent as well, in the sense that it terminates with *arbitrarily* accurate approximation of an equilibrium price vector for an *arbitrary* excess demand function, while being started in an *arbitrary* price vector. The problem of finding an equilibrium price vector for an exchange economy is a so-called zero-point problem. Hence, it may also be applied to solve other problems which can be (re)formulated as a zero-point problem. Furthermore, the variable dimension restart algorithm may be applied in order to approximate the path followed by the globally convergent price adjustment process presented in the same chapter.

Thirdly, we derive an 'intersection theorem'. Intersection theorems are frequently used to prove the existence of equilibrium points in game theory, economics, or to prove existence of solutions to mathematical programming problems. The intersection theorem derived in this chapter may be used to prove the existence of an economic equilibrium in an exchange economy.

The chapter called 'Evolution of populations and strategies[2]', contains two models. The first model describes the evolution of the composition of a population with several subgroups having (genetically) predetermined strategies, assuming that a process of 'Darwinian' selection takes place. Fitness, the crucial driving force of the evolutionary selection process, is determined by the predetermined strategies employed by all subgroups and the composition of the population. The composition of the population is (assumed to be) characterized by a vector of population shares, i.e., a vector consisting of the relative frequencies of the subgroups in the population . The evolutionary selection process favors the fitter subgroups over the less fit ones. This is formalized in the model as follows. Under the selection process any subgroup using a strategy giving a higher-than-average fitness, increases its share in the population, any subgroup using a strategy giving a lower-than-average fitness, decreases its share in the population, and any subgroup using a strategy giving average fitness, has its population share unchanged. At any point in time, the changes in the composition of the population are driven by differences in fitness levels among the subgroups given the strategic environment as mentioned, and this strategic environment changes in general when the composition of the population changes.

The second model in Chapter 3 describes the evolution of a population consisting of several subgroups as well, but instead of having genetically predetermined strategies for each subgroup, the strategies change in general in this new setup. The model describes namely how the subgroups adapt their strategies in order to improve their payoffs in a given strategic environment. We interpret the strategical adjustment processes of the subgroups as learning dynamics. The learning dynamics belong to the same large class of evolutionary dynamics as examined in the first model in this chapter. The second model in Chapter 3 may lead to rather high-dimensional and complex dynamics as it is necessary how to describe the evolution of the composition of the population, as well as the evolution of the strategies employed by the subgroups, given the strategic environment. Again, the strategic environment is not static, but changes in general as the composition of the population changes and the subgroups alter their strategies.

As equilibrium concepts in these evolutionary models, we use the concept of saturated equilibrium Hofbauer & Sigmund [1988], and two new 'evolutionary stability' concepts generalizing the evolutionarily stable strategy of Maynard Smith & Price [1973]. These general-

[2]Chapter 3 is based on Joosten [1995] and Joosten [1996a].

izations of the evolutionarily stable strategy (ESS) are the evolutionarily stable equilibrium and generalized evolutionarily stable state. The former equilibrium concept is to be regarded as a dynamic generalization of the ESS, the latter concept as a static generalization. For large classes of 'plausible' evolutionary dynamics examined in the chapter, the saturated equilibrium, the evolutionarily stable equilibrium and the generalized evolutionarily stable state are rest points.

The three evolutionary equilibrium concepts are examined with respect to dynamic (stability) properties, and relations to fixed point concepts known from the analysis of dynamical systems are shown. A fixed point is called *stable* if the dynamic system remains close to the fixed point for any sufficiently small perturbation of the dynamic system, forever. A fixed point is called *asymptotically stable* if it is stable and the dynamic system eventually returns to the fixed point for any sufficiently small perturbation. For given evolutionary dynamics, each evolutionarily stable equilibrium is an asymptotically stable fixed point by conception. The Euclidean distance forms a Lyapunov function with respect to the dynamics near the evolutionarily stable equilibrium, implying that the Euclidean distance to the evolutionarily stable equilibrium decreases monotonically in time on any trajectory sufficiently nearby. Each generalized evolutionarily stable state is an asymptotically stable fixed point of the replicator dynamics. Moreover, it is shown that any stable fixed point must be a saturated equilibrium, hence every fixed point which is *not* a saturated equilibrium is *unstable*. Furthermore, it is shown that if evolutionary dynamics converge from the interior of the state space, then they converge only to a saturated equilibrium.

For several standard models, each saturated equilibrium corresponds to a Nash equilibrium, and vice versa. In the same branch of models, each generalized evolutionarily stable state corresponds to an ESS, and vice versa. So, one may question the independence of the new evolutionary equilibrium concepts. However, we have established that the one-to-one relationships of the equilibrium conepts for the standard models just mentioned, do not extend to all models in an evolutionary framework. The main message is that evolutionary and learning processes lead to the saturated equilibrium and its refinements such as the evolutionarily stable equilibrium and the generalized evolutionarily stable state. These need not necessarily coincide with the Nash equilibrium and its refinements such as the evolutionarily stable strategy. Examples are provided highlighting discrepancies between these end-products of myopic evolutionary dynamics on one hand, and the equilibrium concepts known from (evolutionary) game theory on the other.

Connections between Chapters 2 and 3 are surprising. From a mathematical viewpoint, evolutionary dynamics may be interpreted as a price adjustment process in a pure exchange economy with normalized prices. In line with this analogy, several fixed point concepts for evolutionary dynamics correspond with fixed point concepts for price adjustment processes. For example, the aforementioned saturated equilibrium corresponds with the Walras equilibrium. Several algorithms to compute a Walras equilibrium in pure exchange economies, can also be used to compute a saturated equilibrium for a deterministic evolutionary model. For instance, the algorithm presented in Chapter 2 may be used to compute a saturated equilibrium if the evolutionary system fulfills a condition called permanence.

The chapter called 'Changing payoffs or action sets[3]', contains two models where the payoffs to the players, or their action sets, may change during the play of the game. In the first model of Chapter 4, two-player repeated games with vanishing actions are studied. In such a game, the number of actions which each player possesses, may decrease as a direct

[3]Chapter 4 is largely based on Joosten, Peters & Thuijsman [1994a], and Joosten, Peters & Thuijsman [1995].

result of how the play develops. If namely, an action has not been used by a player for a certain number of consecutive stages of the play, then this action is removed from this player's action set. This phenomenon of losing an action is referred to as 'unlearning by not doing'. The two-person games that may arise from 'unlearning by not doing' form a special class of stochastic games (Shapley [1953a]) with finite state and action spaces. We show existence of solutions to several classes of zero-sum, as well as general-sum games with vanishing actions, by giving explicit Nash equilibria and optimal strategies for the limiting average reward criterion.

In the second model of Chapter 4, we investigate aspects of changing payoffs by means of differential games. Two players receive streams of payoffs, a fixed amount of which is to be invested in two activities by each player. Each player has one activity of which the payoff only depends on his investment into that activity. The other activity depends not only on his investment into this activity, but also on the investment of the other player in this activity. Assuming that both players wish to maximize their infinite stream of payoffs, Nash equilibria and optimal open-loop strategies are derived.

The chapter called 'Egalitarianism, potentials, and values[4]', deals with a topic in cooperative game theory. One of the questions which we try to answer is: How should possible gains, or costs, made by a group of cooperating players, be allocated? A 'value' for a cooperative game with transferable utility can be regarded as a division-rule of the gains or costs made by a group of cooperating agents. A straightforward manner of dividing the gains or costs made by a group of cooperating agents is to divide them equally among all members of the group. The corresponding value is called the egalitarian value as obviously all agents are treated the same. A prominent example of a value solution is the Shapley value (Shapley [1953b]), which has applications in the social sciences beyond economics and game theory. For instance, the Shapley value has been applied to quantify the power of agents in committees or parliaments.

Our contribution is motivated by a wish to devise a value incorporating a notion of egalitarianism. We do this by assuming that null-players, i.e., players who do not contribute anything to any coalition, i.e., a subset of the player set, in the game, receive a fraction α of the per-capita 'income' in the situation where all players cooperate. This may come about by voluntary contributions to some fund, or by some form of taxation of the total income, which are then to be redistributed evenly among all agents. Thus, the parameter α reflects the prevailing level of egalitarianism or solidarity. We consider values that satisfy this condition, called α-egalitarianism, and several other 'reasonable' properties. This leads to axiomatic characterizations of a class of values, the 'α-egalitarian Shapley values'. We proceed by giving several other axiomatic characterizations of each α-egalitarian Shapley value. As it turns out, any α-egalitarian Shapley value is a(n α-dependent) linear combination of the Shapley value and the egalitarian value.

We take the idea of 'taxing and redistributing' further to derive more general classes of values depending on a tuple of parameters. For this purpose, we make use of so-called 'potentials'. A potential in this context is a map which attributes to every game a unique real number (see e.g. Hart & Mas-Colell [1989]). Given a tuple of parameters, we define a potential and connect to this potential a unique value as follows. Each player in the game receives his marginal contribution to the potential of the game, under such a value. We examine relations between restrictions on the tuple of parameters and several properties of the value which is determined by this tuple of parameters.

[4]Part of Chapter 5 is based on Joosten, Peters, and Thuijsman [1994b].

References

Arrow, K.J., & F. Hahn, 1971, "General Competitive Analysis", Holden-Day, San Francisco.

Hart, S., & A. Mas-Colell, 1989, Potential, value and consistency, *Econometrica* **57**, 615-635.

Hofbauer, J., & K. Sigmund, 1988, "The Theory of Evolution and Dynamical Systems", Cambridge University Press, Cambridge.

Joosten, R.A.M.G., 1995, An evolutionary model with myopic learning, MERIT RM 2/94-019, Maastricht University.

Joosten, R.A.M.G., 1996a, Deterministic evolutionary dynamics: A unifying approach, *Journal of Evolutionary Economics* **6**, 313-324.

Joosten, R.A.M.G., 1996b, "Dynamics, Equilibria, and Values", Dissertation no. 96-37, Faculty of Economics and Business Administration, Maastricht University.

Joosten, R., H. Peters & F. Thuijsman, 1994a, Games with changing payoffs, in "The Economics of Growth and Technical Change", G. Silverberg & L. Soete (eds.), Edward Elgar, Aldershot, pp. 244-257.

Joosten, R., H. Peters & F. Thuijsman, 1994b, Socially acceptable values for transferable utility games, MERIT RM 2/94-021, Maastricht University.

Joosten, R., H. Peters & F. Thuijsman, 1995, Unlearning by not-doing: Repeated games with vanishing actions, *Games and Economic Behavior* **9**, 1-7.

Joosten, R.A.M.G., & A.J.J. Talman, 1993, A simplicial variable dimension restart algorithm to find economic equilibria on the unit simplex using $n(n+1)$ rays, Report M 93-01, Department of Mathematics, Maastricht University.

Joosten, R.A.M.G., & A.J.J. Talman, 1995, A globally convergent price adjustment process for exchange economies, to appear in *Journal of Mathematical Economics.*

Maynard Smith, J., & G.A. Price, 1973, The logic of animal conflict, *Nature* **246**, 15-18.

Samuelson, P.A., 1947, "Foundations of Economic Analysis", Harvard University Press, Cambridge.

Shapley, L.S., 1953a, Stochastic games, *Proceedings of the National Academy of Sciences U.S.A.* **39**, 1095-1100.

Shapley, L.S., 1953b, A value for n-person games, in H.W. Kuhn, A.W. Tucker (eds.), "Contributions to the Theory of Games II", Princeton University Press, Princeton, pp. 307-317.

Walras, L., 1874, "Élements d'Économie Politique Pure", Corbaz, Lausanne.

On consistency of reward allocation rules in sequencing situations

Jeroen Suijs Herbert Hamers Stef Tijs
Center and Department of Econometrics
Tilburg University
E-mail: h.j.m.hamers@kub.nl

Abstract

In this paper we consider the equal gain splitting rule and the split core. Both are solution concepts for sequencing situations and were introduced by *Curiel, Pederzoli* and *Tijs (1989)* and *Hamers, Suijs, Tijs* and *Borm (1994)* respectively. Our goal is a characterization of these solution concepts using consistency properties. However, to do this we need a more subtle look at the allocations assigned by both solution concepts. In the current definitions they assign aggregated allocations, i.e. only the total reward is assigned to each agent. To use consistency in sequencing situations, aggregated solution concepts do not provide sufficient information. What we need is a further specification of this total reward of an agent. Therefore we introduce so called non-aggregated solution concepts. A non-aggregated solution concept assigns a vector to each agent, in some way representing the specification of his total reward. Consequently, a non-aggregated solution concept assigns to each sequencing situation a matrix instead of a vector. In this paper we introduce the non-aggregated counterparts of the equal gain splitting rule and the split core and characterize them using consistency.

1 Introduction

Consistency properties arise in both cooperative and non-cooperative game theory. For surveys we refer to *Thomson (1990)* and *Driessen (1991)* for the first and *Peleg* and *Tijs (1996), Peleg, Potters* and *Tijs (1996)* and *Norde, Potters, Reijnierse* and *Vermeulen (1996)* for the latter. Roughly speaking, a solution concept is called consistent if renegotiation of the subsolution by subcoalitions on the basis of the same solution concept applied to an intuitively appealing reduced situation, will lead to the same suboutcome.

A property closely related to consistency is converse consistency. This property appeared in *Peleg (1985)* in characterizing the core. The main idea behind converse consistency is the following. Given a situation on an agent set N and a solution concept, if it is the case that the prescribed outcomes for the reduced situations on all appropriate subsets of agents fit in the sense that each player receives the same payoff in each reduced situation in which he is involved, the corresponding payoff (if feasible) should be prescribed by the solution concept for the original non-reduced situation.

For the class of combinatorial optimization situations consistency and/or converse consistency has already appeared in assignment situations *(Owen (1992))*, flow situations *(Reijnierse, Maschler, Potters* and *Tijs (1994))* and minimum cost spanning tree situations

(Feltkamp, Tijs and *Muto (1994))*. In this paper we deal with consistency properties of solutions for one machine sequencing situations.

In one-machine sequencing situations a finite number of agents are lined up in front of a single machine, with each agent having exactly one job that has to be processed on this machine. Further each agent incurs costs for every time unit he is in the system. One problem arising from these situations is how to determine the processing order of the jobs which minimizes total costs. This problem was solved by *Smith (1956)* in case all cost functions are linear.

For the class of one machine sequencing situations *Curiel, Pederzoli* and *Tijs (1989)* defined combinatorial optimization games called sequencing games. Moreover, they introduced the Equal Gain Splitting (EGS) rule, which assigns to each sequencing situation a vector that is in the core of the corresponding sequencing game. They characterized this rule using an efficiency, dummy, switch and equivalence property.

The split core, introduced by *Hamers, Suijs, Tijs* and *Borm (1994)*, is a generalization of the EGS rule and assigns to each sequencing situation a non-empty subset of the core of the corresponding sequencing game. They provide a characterization of the split core using efficiency, the dummy property and a kind of monotonicity.

In this paper we give characterizations of the EGS rule and the split core using certain consistency properties. However, to achieve these characterizations, we have to take a more subtle look at the allocations assigned by both solution concepts. Usually these allocations are aggregated. But here we will consider non-aggregated allocations corresponding to both solution concepts. In the current definitions they assign aggregated allocations, i.e. only the total reward is assigned to each agent. To use consistency in sequencing situations, aggregated solution concepts do not provide sufficient information. What we need is a further specification of this total reward of an agent. Therefore we introduce so called non-aggregated solution concepts. A non-aggregated solution concept assigns a vector to each agent, in some way representing the specification of his total reward. Consequently, a non-aggregated solution concept assigns to each sequencing situation a matrix instead of a vector. In this paper we introduce the non-aggregated counterparts of the equal gain splitting rule and the split core and characterize them using consistency.

The paper is organized as follows. One machine sequencing situations are formally described in section 2. We also recall the definitions of the aggregated EGS rule and the aggregated split core and introduce their non-aggregated counterparts. In section 3 efficiency, symmetry and consistency are used to characterize the non-aggregated Equal Gain Splitting rule and efficiency, consistency and converse consistency are used to characterize the non-aggregated split core.

2 Sequencing and solution concepts

In a one machine sequencing situation a finite number of agents, each having one job, are lined up in front of a single machine, waiting for their jobs to be processed. We denote by $N \subseteq \mathbf{N}$ the finite set of agents and n the number of agents. Further, we describe the queue formed by the agents with a bijection $\sigma : N \to \{1, 2, ..., n\}$, where $\sigma(i)$ denotes the position of player i in the queue. Particularly we denote by σ_0 the initial order of the agents and with Π_N the set of all such bijections σ. Without loss of generality we may assume that $\sigma_0(i) = i$ for all $i \in N$. The processing time p_i is the time the machine needs to process the job of agent i. Finally we assume that agent i has an affine cost function $c_i : \mathbf{R}_+ \to \mathbf{R}_+$ defined by $c_i(t) = \alpha_i t + \beta_i$ with $\alpha_i > 0$ and $\beta_i \in \mathbf{R}_+$. So $c_i(t)$ are the costs for agent i when he spends

t time units in the system.

A sequencing situation as above is denoted by (N, p, α, σ_0), where $N \subseteq \mathbf{N}$, $p = (p_i)_{i \in N} \in \mathbf{R}^n_+$, $\alpha = (\alpha_i)_{i \in N} \in \mathbf{R}^n_+$ and $\sigma_0 : N \to \{1, 2, ..., n\}$. The vector $\beta = (\beta_i)_{i \in N} \in \mathbf{R}^n_+$ representing fixed costs is omitted in the description of a sequencing situation since these costs are independent of the positions of the agents in the queue. In the remainder we denote with SEQ the set of all sequencing situations with player set any finite subset of the natural numbers. For ease of notation an element of SEQ is denoted with $\Gamma(N)$, where N is the set of agents.

Given the processing order of the jobs $\sigma : N \to \{1, 2, ..., n\}$ the completion time of job i equals $C(\sigma, i) = \sum_{j:\sigma(j) \leq \sigma(i)} p_j$ and the costs incurred by player i equal $c_i(C(\sigma, i)) = \alpha_i C(\sigma, i) + \beta_i$. By rearranging the agents the total costs can be reduced. *Smith (1956)* showed that the total costs are minimal if the agents are placed in decreasing order with respect to α_i / p_i. In the remainder of this paper we call such a cost minimizing order an optimal order.

The Equal Gain Splitting (EGS) rule of a sequencing situation $\Gamma(N)$ is for all $i \in N$ defined by

$$EGS_i(\Gamma(N)) = \frac{1}{2} \sum_{j:\sigma_0(i) \leq \sigma_0(j)} g_{ij}(\Gamma(N)) + \frac{1}{2} \sum_{k:\sigma_0(k) \leq \sigma_0(i)} g_{ki}(\Gamma(N))$$

where $g_{ij}(\Gamma(N)) = \max(0, p_i \alpha_j - p_j \alpha_i)$ represents the gain agents i and j can obtain if agent i is directly in front of agent j. An optimal order can be obtained from the initial order by consecutive switches of neighbours i and j with $g_{ij}(\Gamma(N)) > 0$. The EGS rule then divides the gain obtained with a neighbour switch equally among both agents involved in the neighbour switch. Note that the EGS rule only assigns the final payoff to each agent. So the allocation corresponding with the EGS rule is aggregated. *Curiel et al. (1989)* showed that for every sequencing situation (N, p, α, σ_0) the EGS rule results in a core allocation of the corresponding sequencing game.

Based on a generalization of the EGS rule *Hamers et al. (1994)* introduced the split core of a sequencing game. The split core consists of all gain splitting allocations. One obtains a gain splitting allocation by dividing the gain obtained with a neighbour switch not equally but arbitrarily among the agents involved in the neighbour switch. Formally, a gain splitting allocation of $\Gamma(N)$ is defined for all $i \in N$ and all $\lambda \in \Lambda$ by

$$GS^{\lambda}_i(\Gamma(N)) = \sum_{j:\sigma_0(i) \leq \sigma_0(j)} \lambda_{ij} g_{ij}(\Gamma(N)) + \sum_{k:\sigma_0(k) \leq \sigma_0(i)} (1 - \lambda_{ki}) g_{ki}(\Gamma(N)) \tag{1}$$

with $\Lambda = \{\{\lambda_{ij}\}_{i,j \in N} | 0 \leq \lambda_{ij} \leq 1\}$. Then the split core of a sequencing situation $\Gamma(N)$ is equal to

$$SPC(\Gamma(N)) = \{GS^{\lambda}(\Gamma(N)) | \lambda \in \Lambda\}.$$

Hamers et al. (1994) showed that the split core is a subset of the core. Moreover, if $\lambda_{ij} = 1/2$ for all $i, j \in N$ we have $GS^{\lambda}(\Gamma(N)) = EGS(\Gamma(N))$. Finally, note that the split core is a set of aggregated allocations.

Example 2.1 Let $N = \{1, 2, 3\}$, $p = (1, 1, 1)$, $\alpha = (1, 2, 4)$ and $\sigma_0(i) = i$ for all $i \in N$. It follows that $g_{12}(\Gamma(N)) = 1$, $g_{13}(\Gamma(N)) = 3$ and $g_{23}(\Gamma(N)) = 2$. Then $GS^{\lambda}_1(\Gamma(N)) = \lambda_{12} + 3\lambda_{13}$, $GS^{\lambda}_2(\Gamma(N)) = (1 - \lambda_{12}) + 2\lambda_{23}$ and $GS^{\lambda}_3(\Gamma(N)) = 2(1 - \lambda_{23}) + 3(1 - \lambda_{13})$ with $0 \leq \lambda_{ij} \leq 1$ for all $i, j \in N$. In particular $EGS(\Gamma(N)) = (2, 3/2, 5/2)$.

We will now define solution concepts on sequencing situations in a slightly different manner. Instead of assigning an aggregated allocation of the total cost savings, we assign to each sequencing situation a non-aggregated allocation. In this context, non-aggregated means that a specification of the total reward an agent obtains is assigned to that agent. More formally, a non-aggregated solution ϕ is a map assigning to each sequencing situation $\Gamma(N) \in SEQ$ a matrix $W \in \mathbf{R}_+^{N \times N}$, where an element w_{ij} of W represents the non-negative gain assigned to agent i for cooperating with agent j. The aggregated allocation corresponding with a solution W can be found by multiplying W with the vector $e = (1, 1, ..., 1)^\top \in \mathbf{R}^N$. Now we can define the non-aggregated counterparts of the Equal Gain Splitting rule and the split core.

The non-aggregated Equal Gain splitting solution $\mathcal{EGS}$ assigns to each sequencing situation $\Gamma(N) \in SEQ$ a solution $\mathcal{EGS}(\Gamma(N)) \in \mathbf{R}_+^{N \times N}$ such that

$$\mathcal{EGS}(\Gamma(N))_{ij} = \begin{cases} \frac{1}{2} g_{ij}(\Gamma(N)), & \text{if } \sigma_0(i) \leq \sigma_0(j) \\ \frac{1}{2} g_{ji}(\Gamma(N)), & \text{if } \sigma_0(i) \geq \sigma_0(j) \end{cases}$$

for all $i, j \in N$. Note that the allocation $\mathcal{EGS}(\Gamma(N)) \cdot e$ is equal to the equal gain splitting allocation $EGS(\Gamma(N))$.

Example 2.2 Take the sequencing situation of example 2.1. The optimal order for this situation is 3,2,1. The gain splitting matrix $\mathcal{EGS}(\Gamma(N))$ and the corresponding allocation equal respectively

$$\mathcal{EGS}(\Gamma(N)) = \begin{bmatrix} 0 & 1/2 & 3/2 \\ 1/2 & 0 & 1 \\ 3/2 & 1 & 0 \end{bmatrix} \qquad \mathcal{EGS}(\Gamma(N)) \cdot e = \begin{bmatrix} 2 \\ 3/2 \\ 5/2 \end{bmatrix}$$

The non-aggregated split core $\mathcal{SPC}$ assigns to each sequencing situation $\Gamma(N) \in SEQ$ a non-empty subset $\mathcal{SPC}(\Gamma(N)) \subseteq \mathbf{R}_+^{N \times N}$ such that for each gain splitting matrix $\mathcal{GS}(\Gamma(N)) \in \mathcal{SPC}(\Gamma(N))$

$$\mathcal{GS}(\Gamma(N))_{ij} + \mathcal{GS}(\Gamma(N))_{ji} = \begin{cases} g_{ij}(\Gamma(N)), & \text{if } \sigma_0(i) \leq \sigma_0(j) \\ g_{ji}(\Gamma(N)), & \text{if } \sigma_0(i) \geq \sigma_0(j) \end{cases}$$

for all $i, j \in N$. An allocation corresponding with an element $\mathcal{GS}(\Gamma(N))) \in \mathcal{SPC}(\Gamma(N))$ equals $\mathcal{GS}(\Gamma(N)) \cdot e$ and is an element of the split core $SPC(\Gamma(N))$. This is easily checked by taking

$$\lambda_{ij} = \begin{cases} \mathcal{GS}(\Gamma(N)))_{ij} / g_{ij}(\Gamma(N)), & \text{if } \sigma_0(i) < \sigma_0(j) \text{ and } g_{ij}(\Gamma(N)) > 0 \\ 0, & \text{otherwise} \end{cases}$$

for all $i, j \in N$ and substituting in expression (1). We conclude this section with another example.

Example 2.3 Take the sequencing situation of example 2.1. The optimal order for this situation is 3,2,1. Then the split core $\mathcal{SPC}(\Gamma(N))$ equals

$$\mathcal{SPC}(\Gamma(N)) = \left\{ \begin{bmatrix} 0 & \lambda_{12} & \lambda_{13} \\ 1 - \lambda_{12} & 0 & \lambda_{23} \\ 3 - \lambda_{13} & 2 - \lambda_{23} & 0 \end{bmatrix} \mid 0 \leq \lambda_{ij} \leq 1,\ i, j \in \{1, 2, 3\} \right\}$$

Note that the set of allocations $\{W \cdot e | W \in \mathcal{SPC}(\Gamma(N))\}$ coincides with the split core $SPC(\Gamma(N))$.

3 Axiomatizations of the $\mathcal{SPC}$ and $\mathcal{EGS}$ solutions

In this section we characterize both the non-aggregated split core $\mathcal{SPC}$ and the non-aggregated $\mathcal{EGS}$ rule. For these axiomatizations we need the notions of connected coalitions and reduced sequencing situations. A coalition S is connected if for all $i,j \in S$ and all $k \in N$ with $\sigma_0(i) < \sigma_0(k) < \sigma_0(j)$ it holds that $k \in S$. The set of all non-empty connected coalitions with respect to the initial processing order σ_0 is denoted with $con(\sigma_0)$.

A sequencing situation reduced to a connected coalition S is the sequencing situation remaining when the agents outside coalition S are left out of consideration. The situation which remains is described by $\Gamma(N|_S) = (S, p^S, \alpha^S, \sigma_0^S)$ with $p^S = (p_i)_{i\in S}, \alpha^S = (\alpha_i)_{i\in S}$ and $\sigma_0^S \in \Pi_S$, where the latter is such that for all $i,j \in S$ it holds that $\sigma_0^S(i) < \sigma_0^S(j)$ whenever $\sigma_0(i) < \sigma_0(j)$. We will clarify this with the following example.

Example 3.4 Take $N = \{1,2,3,4,5\}, p = (1,2,2,1,3), \alpha = (1,1,3,2,7)$ and $\sigma_0(i) = i$ for all $i \in N$. Note that the total cost savings are maximal when the jobs are processed in the order 5,4,3,1,2. The coalition $S = \{2,3,4\}$ is a connected coalition. This situation reduced to S is the situation with $S = \{2,3,4\}, p^S = (2,2,1), \alpha^S = (1,3,2)$ and $\sigma_0^S(2) = 1, \sigma_0^S(3) = 2, \sigma_0^S(4) = 3$.

Let ψ be a non-aggregated solution concept that assigns to each $\Gamma(N) \in SEQ$ a matrix $\psi(\Gamma(N)) \in \mathbf{R}_+^{N\times N}$ and let $\hat{\sigma}$ denote an optimal order for $\Gamma(N)$. For the characterization of the non-aggregated equal gain splitting solution $\mathcal{EGS}$ we introduce the following three properties.

(i) **Efficiency** : ψ is efficient if for all $\Gamma(N) \in SEQ$ it holds that

$$\sum_{i,j\in N} \psi(\Gamma(N))_{ij} = \sum_{i\in N} c_i(C(\sigma_0, i)) - \sum_{i\in N} c_i(C(\hat{\sigma}, i)).$$

(ii) **Symmetry** : ψ is called symmetric if for all $\Gamma(N) \in SEQ$ the matrix $\psi(\Gamma(N)) \in \mathbf{R}_+^{N\times N}$ is symmetric.

(iii) **Consistency** : Let $\Gamma(N) \in SEQ$. Then ψ is called consistent if for all $\Gamma(N) \in SEQ$ and all $S \in con(\sigma_0)$ different from N it holds that $\psi(\Gamma(N))|_S = \psi(\Gamma(N|_S))$, where $\psi(\Gamma(N))|_S$ is the matrix with all columns and rows of members outside S deleted.

Efficiency means that exactly the maximal total cost savings is allocated over the agents. Symmetry tells us that the gain two agents can obtain by cooperating is divided equally among both of them. Consistency of a solution concept means that subcoalitions obtain the same outcome if they renegotiate the (sub)solution on the basis of the same solution concept to an intuitively appealing reduced situation To explain consistency more specific for sequencing situations we use the following example, based on the situation described in example 3.4.

In this situation we have $N = \{1,2,3,4,5\}$, $p = (1,2,2,1,3)$, $\alpha = (1,1,3,2,7)$ and $\sigma_0(i) = i$ for all $i \in N$. Next, consider the coalition $S = \{2,3,4\}$. The members of S form a connected coalition. Hence, the agents in coalition S can rearrange their processing order without the cooperation of agents outside S. This problem can be considered as a reduced sequencing situation $(S, p^S, \alpha^S, \sigma_0^S)$ with agents $S = \{2,3,4\}$, $p^S = (2,2,1)$, $\alpha^S = (1,3,2)$

and $\sigma_0^S(i) = i - 1$ for all $i \in S$. Note, however, that the agents outside S have not left the queue. But since all cost functions are affine, the processing times of the agents in front of coalition S do not influence the cooperation of coalition S. Hence, we may consider the initial order $\sigma_0^S(i) = i - 1$, (for all $i \in S$) in the above reduced situation instead of the order $\sigma_0^S = i$, (for all $i \in S$), which describes the real positions of the members of S in the initial processing order σ_0. The allocation assigned by the non-aggregated $\mathcal{EGS}$ rule then equals for this reduced situation

$$\mathcal{EGS}(\Gamma(N|_S)) = \frac{1}{2} \cdot \begin{bmatrix} 0 & 4 & 3 \\ 4 & 0 & 1 \\ 3 & 1 & 0 \end{bmatrix} \tag{2}$$

We will now show that for coalition $S = \{2, 3, 4\}$ and the non-aggregated $\mathcal{EGS}$ solution consistency is indeed satisfied in this example. For the situation with agent set N the non-aggregated $\mathcal{EGS}$ allocation equals

$$\mathcal{EGS}(\Gamma(N)) = \frac{1}{2} \cdot \begin{bmatrix} 0 & 0 & 1 & 1 & 4 \\ 0 & 0 & 4 & 3 & 11 \\ 1 & 4 & 0 & 1 & 5 \\ 1 & 3 & 1 & 0 & 1 \\ 4 & 11 & 5 & 1 & 0 \end{bmatrix}$$

The reduced matrix $\mathcal{EGS}(\Gamma(N))|_S$ can then be found by deleting the columns and rows of agents outside S of the matrix $\mathcal{EGS}(\Gamma(N))$, that is deleting columns 1 and 5 and rows 1 and 5. The resulting matrix equals $\mathcal{EGS}(\Gamma(S))$. Hence, the allocation of the gain obtained by coalition S is not influenced by the agents 1 and 5.

Why only reductions to connected coalitions are considered is a result of the model introduced in *Curiel et al. (1989)*. In this paper the authors introduce cooperative games which correspond with the sequencing situations described in section 2. In these games two members of a coalition S can only cooperate if the agents standing between them in the processing order are also members of this coalition, that is, coalition S is connected. As a consequence, connected coalitions are the only coalitions which have to be considered.

We will now state our characterization of the non-aggregated $\mathcal{EGS}$ solution.

Theorem 3.5 The $\mathcal{EGS}$ solution is the unique solution satisfying efficiency, symmetry and consistency.

PROOF: First we will show that $\mathcal{EGS}$ satisfies these properties. Therefore let $\Gamma(N) \in SEQ$ be a sequencing situation and denote with $\hat{\sigma}$ an optimal order for $\Gamma(N)$. Symmetry follows from the definition of $\mathcal{EGS}$. Efficiency follows from

$$\sum_{i,j \in N} \mathcal{EGS}(\Gamma(N))_{ij} = \sum_{i,j:\sigma_0(i)<\sigma_0(j)} g_{ij}(\Gamma(N)) = \sum_{i \in N} c_i(C(\sigma_0, i)) - \sum_{i \in N} c_i(C(\hat{\sigma}, i)).$$

Finally, for consistency it is again sufficient to show that for all connected coalitions $S \in con(\sigma_0)$ we have $g_{ij}(\Gamma(N|_S)) = g_{ij}(\Gamma(N))$ for all $i, j \in S$. This follows from the fact that $\sigma_0^S(i) < \sigma_0^S(j)$ if and only if $\sigma_0(i) < \sigma_0(j)$ for all $i, j \in S$ and all $S \in con(\sigma_0)$.

The reverse will be proved with induction on the number of agents. Let ψ be a non-empty solution concept satisfying symmetry, efficiency and consistency. If $|N| = 1$ efficiency yields $\psi(\Gamma(N)) = \mathcal{EGS}(\Gamma(N)) = [0]$ for all $\Gamma(N) \in SEQ$. Now assume that $\psi = \mathcal{EGS}$ for

all $|N| < m$. Take $|N| = m$ and choose $\Gamma(N) \in SEQ$. Reducing $\Gamma(N)$ to $S = N\backslash\{\sigma_0^{-1}(1)\}$ and $S = N\backslash\{\sigma_0^{-1}(n)\}$ respectively, applying consistency and using the induction hypothesis yields

$$\psi(\Gamma(N))_{ij} = \psi(\Gamma(N))_{ji} = \begin{cases} \frac{1}{2}g_{ij}(\Gamma(N)), & \text{if } \sigma_0(i) \leq \sigma_0(j) \\ \frac{1}{2}g_{ji}(\Gamma(N)), & \text{if } \sigma_0(i) \geq \sigma_0(j) \end{cases}$$

for all pairs $(i,j) \neq (1,n)$ and $(i,j) \neq (n,1)$. Efficiency and symmetry then gives

$$\psi(\Gamma(N))_{1n} = \psi(\Gamma(N))_{n1} = \begin{cases} \frac{1}{2}g_{1n}(\Gamma(N)), & \text{if } \sigma_0(1) \leq \sigma_0(n) \\ \frac{1}{2}g_{n1}(\Gamma(N)), & \text{if } \sigma_0(1) \geq \sigma_0(n) \end{cases}$$

Hence, $\psi(\Gamma(N)) = \mathcal{EGS}(\Gamma(N))$ for all $\Gamma(N) \in SEQ$. □

Before we turn to the characterization of the non-aggregated split core, we show that the properties in theorem 3.5 are logically independent. First consider the solution assigning to each sequencing situation the null matrix. It is obvious that this solution is not efficient but satisfies symmetry and consistency. As we will show later, a non-aggregated Gain Splitting solution with fixed $\{\lambda_{ij}\}_{i,j\in N}$ satisfies efficiency and consistency but not necessarily symmetry. Finally, the solution concept assigning to each sequencing situation $\Gamma(N) \in SEQ$ the matrix $W(\Gamma(N))$ with

$$W(\Gamma(N))_{ij} = \begin{cases} \frac{1}{n}\sum_{k,l\in N} g_{kl}(\Gamma(N)) & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases} \tag{3}$$

satisfies efficiency and symmetry but violates consistency.

For the characterization of the non-aggregated split core, let ψ be a non-aggregated solution concept that assigns to each $\Gamma(N) \in SEQ$ a non-empty subset of $\mathbf{R}_+^{N\times N}$ and let $\hat{\sigma}$ denote an optimal order for $\Gamma(N)$. Consider the following three properties for ψ.

(i) **Efficiency** : ψ is efficient if for all $\Gamma(N) \in SEQ$ and all $W \in \psi(\Gamma(N))$ it holds that

$$\sum_{i,j\in N} W_{ij} = \sum_{i\in N} c_i(C(\sigma_0, i)) - \sum_{i\in N} c_i(C(\hat{\sigma}, i)).$$

(ii) **Consistency** : Let $\Gamma(N) \in SEQ$. Then ψ is called consistent if for all $\Gamma(N) \in SEQ$, all $S \in con(\sigma_0)$ and all $W \in \psi(\Gamma(N))$ it holds that $W|_S \in \psi(\Gamma(N|_S))$, where $W|_S$ is the matrix W with all columns and rows of agents not in S deleted.

(iii) **Converse consistency** : ψ is conversely consistent if for all $W \in \mathbf{R}_+^{N\times N}$ and all $\Gamma(N) \in SEQ$ with $\sum_{i,j\in N} W_{ij} = \sum_{i\in N} c_i(C(\sigma_0, i)) - \sum_{i\in N} c_i(C(\hat{\sigma}, i))$ the following statement is true. If $W|_S \in \psi(\Gamma(N|_S))$ for all connected coalitions $S \in con(\sigma_0)$ different from N then $W \in \psi(\Gamma(N))$.

The efficiency property states that exactly the maximal total cost savings are allocated over the agents. For the multifunction case, consistency can also be seen as a stability condition. To see this, consider again the situation described in example 3.4. Next, reduce this sequencing situation to the connected coalition $S = \{2,3,4\}$. The non-aggregated split core equals for this reduced situation

$$\mathcal{SPC}(\Gamma(N|_S)) = \left\{ \begin{bmatrix} 0 & 4\lambda_{23} & 3\lambda_{24} \\ 4(1-\lambda_{23}) & 0 & \lambda_{34} \\ 3(1-\lambda_{24}) & 1-\lambda_{34} & 0 \end{bmatrix} | \lambda_{23}, \lambda_{24}, \lambda_{34} \in [0,1] \right\}$$

Although the allocations differ for the several choices of $\lambda_{23}, \lambda_{24}, \lambda_{34}$, the total which is allocated to coalition S is constant and equal to 8. So a possible allocation $\mathcal{GS}(\Gamma(N)) \in \mathcal{SPC}(\Gamma(N))$ will only be accepted by coalition S if the total cost savings assigned to the agents of S for cooperating with members of S is not less than 8. The consistency property guarantees that coalition S gets exactly 8. Hence, coalition S will accept an allocation satisfying consistency. For the split core this property is satisfied for coalition S in this example. This can easily be checked by computing the total cost savings assigned by the non-aggregated split core to agents in S for cooperating with other agents in S. The split core equals

$$\mathcal{SPC}(\Gamma(N)) = \left\{ \begin{bmatrix} 0 & 0 & \lambda_{13} & \lambda_{14} & 4\lambda_{15} \\ 0 & 0 & 4\lambda_{23} & 3\lambda_{24} & 11\lambda_{25} \\ \bar{\lambda}_{13} & 4\bar{\lambda}_{23} & 0 & \lambda_{34} & 5\lambda 35 \\ \bar{\lambda}_{14} & 3\bar{\lambda}_{24} & \bar{\lambda}_{34} & 0 & \lambda_{45} \\ 4\bar{\lambda}_{15} & 11\bar{\lambda}_{25} & 5\bar{\lambda}_{35} & \bar{\lambda}_{45} & 0 \end{bmatrix} \middle| \begin{array}{ll} \lambda_{ij} \in [0,1], & i,j \in N \\ \bar{\lambda}_{ij} = 1 - \lambda_{ij}, & i,j \in N \end{array} \right\}$$

and the total cost savings for coalition S equals $\sum_{i,j \in S} \mathcal{GS}(\Gamma(N))_{ij} = 8$ for all $\mathcal{GS}(\Gamma(N)) \in \mathcal{SPC}(\Gamma(N))$.

So a consistent solution concept assigns to each connected coalition exactly the gain this coalition can obtain in its reduced situation. Thus, consistency guarantees a form of stability which differs from the stability guaranteed by the core of a cooperative game. Because the core consists of allocations for which each coalition, connected or not, gets at least the gain this coalition can obtain without the cooperation of agents outside this coalition.

Finally, converse consistency means that when each allowed reduced matrix of a feasible matrix (that is, the maximal cost savings are allocated over the agents) is an element of the solution of the corresponding reduced situation, then this gain splitting matrix must also be an element of the solution of the non-reduced situation. Note that for sequencing situations only reductions to connected coalitions are allowed.

With the three aforementioned properties we can characterize the non-aggregated split core.

Theorem 3.6 The non-aggregated split core $\mathcal{SPC}$ is the unique non-empty solution satisfying efficiency, consistency and converse consistency.

PROOF: We will first show that $\mathcal{SPC}$ satisfies all three properties. Therefore, let $\Gamma(N) \in SEQ$ and let $\hat{\sigma}$ be an optimal order for $\Gamma(N)$. Take $W \in \mathcal{SPC}(\Gamma(N))$. Efficiency follows from

$$\sum_{i,j \in N} W_{ij} = \sum_{i,j:\sigma_0(i)<\sigma_0(j)} g_{ij}(\Gamma(N)) = \sum_{i \in N} c_i(C(\sigma_0, i)) - \sum_{i \in N} c_i(C(\hat{\sigma}, i)).$$

Next, consider consistency. From the definition of the non-aggregated split core $\mathcal{SPC}$, it is sufficient to show that for all connected coalitions $S \in con(\sigma_0)$ we have $g_{ij}(\Gamma(N|_S)) = g_{ij}(\Gamma(N))$ for all $i,j \in S$. But this follows from $\sigma_0^S(i) < \sigma_0^S(j)$ if and only if $\sigma_0(i) < \sigma_0(j)$ for all $i,j \in S$ and all $S \in con(\sigma_0)$.

For converse consistency, take $\Gamma(N) \in SEQ$ and a solution $W \in \mathbf{R}_+^{N \times N}$ such that $\sum_{i,j \in N} W_{ij} = \sum_{i \in N} c_i(C(\sigma_0, i)) - \sum_{i \in N} c_i(C(\hat{\sigma}, i))$. Reducing the situation to $S = N \backslash \{\sigma_0^{-1}(1)\}$ and $S = N \backslash \{\sigma_0^{-1}(n)\}$ respectively and using $W|_S \in \mathcal{SPC}(\Gamma(N|_S))$ and $g_{ij}(\Gamma(N|_S)) = g_{ij}(\Gamma(N))$ for all $i,j \in S$ and all $S \in con(\sigma_0)$ gives

$$W_{ij} + W_{ji} = \begin{cases} g_{ij}(\Gamma(N)), & \text{if } \sigma_0(i) \leq \sigma_0(j) \\ g_{ji}(\Gamma(N)), & \text{if } \sigma_0(i) \geq \sigma_0(j) \end{cases}$$

for all pairs $(i,j) \neq (1,n)$ and $(i,j) \neq (n,1)$. Efficiency then implies that

$$W_{1n} + W_{n1} = \begin{cases} g_{1n}(\Gamma(N)), & \text{if } \sigma_0(1) \leq \sigma_0(n) \\ g_{n1}(\Gamma(N)), & \text{if } \sigma_0(1) \geq \sigma_0(n) \end{cases}$$

Hence, $W \in \mathcal{SPC}(\Gamma(N))$.

So we are left to prove that if a non-empty solution satisfies these three axioms this solution concept must be the split core $\mathcal{SPC}$. Therefore take a non-empty solution concept ψ satisfying efficiency, consistency and converse consistency. We prove by induction to the number of agents that $\psi = \mathcal{SPC}$. Take $|N| = 1$ and let $\Gamma(N) \in SEQ$. Efficiency implies that $\psi(\Gamma(N)) = \mathcal{SPC}(\Gamma(N)) = [0]$. So for $|N| = 1$ we have $\psi = \mathcal{SPC}$.

Now suppose that $\psi = \mathcal{SPC}$ for $|N| < m$. Take $|N| = m$ and let $\Gamma(N) \in SEQ$. Let $W \in \psi(\Gamma(N))$, then consistency of ψ implies that $W|_S \in \psi(\Gamma(N|_S))$ for all connected coalitions $S \in con(\sigma_0)$ with $S \neq N$. Using the induction hypothesis yields $W|_S \in \mathcal{SPC}(\Gamma(N|_S))$. Applying the converse consistency of $\mathcal{SPC}$ gives $W \in \mathcal{SPC}(\Gamma(N))$. Hence, $\psi(\Gamma(N)) \subseteq \mathcal{SPC}(\Gamma(N))$ for all $\Gamma(N) \in SEQ$. Interchanging the roles of ψ and $\mathcal{SPC}$ yields $\mathcal{SPC}(\Gamma(N)) \subseteq \psi(\Gamma(N))$ for all $\Gamma(N) \in SEQ$, so $\psi = \mathcal{SPC}$, which proves the result. □

To conclude this paper we will show that these properties are logically independent. As showed before, the set-valued solution $\{\mathcal{EGS}\}$ satisfies all properties but converse consistency. The solution assigning to each sequencing situation the null matrix satisfies both consistency properties but not efficiency. And finally, the solution concept defined in (3) satisfies efficiency and converse consistency and violates consistency.

References

I. Curiel, G. Pederzoli, S. Tijs (1989), Sequencing games, *European Journal of Operational Research*, **40**, 344-351.

T. Driessen (1991), A survey of consistency properties in cooperative game theory, *SIAM Review*, **33**, 43-59.

V. Feltkamp, S. Tijs, S. Muto (1994), Bird's tree allocation revisited, *CentER discussion paper, Tilburg University*, **9435**.

H. Hamers, J. Suijs, S. Tijs, P. Borm (1994), The split core for sequencing games, *Games and Economic Behavior*, **15**, 165-176.

H. Norde, J. Potters, H. Reijnierse, D. Vermeulen (1996), Equilibrium selection and consistency, *Games and Economic Behavior*, **12**, 219-225.

G. Owen (1992), The assignment game: the reduced game, *Ann. Econom. Statist.*, **25**, 71-79.

B. Peleg (1985), An axiomatization of the core of cooperative games without side payments, *Journal of Mathematical Economics*, **14**, 203-214.

B. Peleg (1986), On the reduced game property and its converse, *International Journal of Game Theory*, **15**, 187-200.

B. Peleg, J. Potters, S. Tijs (1996), Minimality of consistent solutions for strategic games, in particular for potential games, *Economic Theory*, **7**, 81-93.

B. Peleg, S. Tijs (1996), The consistency principle for games in strategic form, *International Journal of Game Theory*, **25**, 13-34.

H. Reijnierse, M. Maschler, J. Potters, S. Tijs (1996, Simple flow games, *Games and Economic Behavior*, **16**, 238-260.

W. Smith (1956), Various optimizers for single- stage production, *Naval Research Logistics Quarterly*, **3**, 59-66.

W. Thomson (1990), The consistency principle, *Game Theory Applications* (T. Ichiishi, A. Neyman, Y. Tauman eds.). San Diego, California: Academic Press., 187-215.

A Survey on Optimality and Equilibria in Stochastic Games

Frank Thuijsman
Department of Mathematics
Maastricht University
frank@math.unimaas.nl

Abstract

In this paper we discuss the main existence results on optimality and equilibria in two-person stochastic games with finite state and action spaces. Several examples are provided to clarify the issues.

1 The Stochastic Game Model

In this introductory section we give the necessary definitions and notations for the two-person case of the stochastic game model and we briefly present some basic results. In section 2 we discuss the main existence results for zero-sum stochastic games, while in section 3 we focus on general-sum stochastic games. In each section we discuss several examples to illustrate the most important phenomena.

It all started with the fundamental paper by Von Neumann [1928] in which he proves the so called minimax theorem which says that for each finite matrix of reals $A = [a_{ij}]_{i=1,j=1}^{m\ n}$ there exist probability vectors $\bar{x} = (\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_m)$ and $\bar{y} = (\bar{y}_1, \bar{y}_2, \ldots, \bar{y}_n)$ such that for all x and y it holds that[1] $xA\bar{y} \leq \bar{x}A\bar{y} \leq \bar{x}Ay$. In other words: $\max_x \min_y xAy = \min_y \max_x xAy$. This theorem can be interpreted to say that each matrix game has a value. A matrix game A is played as follows. Simultaneously, and independent from each other, player 1 chooses a row i and player 2 chooses a column j of A. Then player 2 has to pay the amount a_{ij} to player 1. Each player is allowed to randomize over his available actions and we assume that player 1 wants to maximize his expected payoff, while player 2 wants to minimize the expected payoff to player 1. The minimax theorem tells us that, for each matrix A there is a unique amount val(A), which player 1 can guarantee as his minimal expected payoff, while at the same time player 2 can guarantee that the expected payoff to player 1 will be at most this amount.

Later Nash [1951] considered the n-person extension of matrix games, in the sense that all n players, simultaneously and independently choose actions that determine a payoff for each and every one of them. Nash [1951] showed that in such games there always exists at least one (Nash-)equilibrium: a tuple of strategies such that each player is playing a best reply against the joint strategy of his opponents. For the two-player case this boils down to a "bimatrix game" where players 1 and 2 receive a_{ij} and b_{ij} respectively in case their choices

[1] Note that we do not distinguish row vectors from column vectors. In the matrix products this should be clear from the context.

determine entry (i,j). The result of Nash says that there exist $\bar{x}$ and $\bar{y}$ such that for all x and y it holds that $\bar{x}A\bar{y} \geq xA\bar{y}$ and $\bar{x}B\bar{y} \geq \bar{x}By$, where $A = [a_{ij}]$ and $B = [b_{ij}]$ are finite matrices of the same size.

Shapley [1953] introduced dynamics into game theory by considering the situation that at discrete stages in IN the players play one of finitely many matrix games, where the choices of the players determine a payoff to player 1 (by player 2) as well as a stochastic transition to go to a next matrix game. He called these games "stochastic games", which brings us to the topic of this paper. Formally, a two-person stochastic game with finite state and action spaces can be represented by a finite set of matrices $A^1, A^2, \ldots, A^z$ corresponding to the set of states $S = \{1, 2, \ldots, z\}$. For $s \in S$ matrix A^s has size $m_s \times n_s \in IN \times IN$ and entry (i,j) of A^s contains:

a) a payoff $r^k(s,i,j) \in IR$ for each player $k \in \{1,2\}$

b) a transition probability vector $p(s,i,j) = (p(1|s,i,j), p(2|s,i,j), \ldots, p(z|s,i,j))$ where $p(t|s,i,j)$ is the probability of a transition from s to t whenever entry (i,j) of A^s is selected.

Play can start in any state of S and evolves by players independently choosing actions i_n and j_n of A^{s_n}, where s_n denotes the state visited at stage n. In case $r^1(s,i,j) + r^2(s,i,j) = 0$, the game is called zero-sum, otherwise it is called general-sum. In zero-sum games players have strictly opposite interests, since they are paying each other.

A strategy for a player is a rule that tells him for any history $h_n = (s_1, i_1, j_1, s_2, i_2, j_2, \ldots, s_{n-1}, i_{n-1}, j_{n-1}, s_n)$ up to stage n, what mixed action to use in state s_n at stage $n \in IN$. Such behavior strategies will be denoted by π for player 1 and by σ for player 2.

For initial state s and any pair of strategies (π, σ) the limiting average reward and the β-discounted reward, $\beta \in (0,1)$, to player $k \in \{1,2\}$ are respectively given by

$$\gamma^k(s,\pi,\sigma) = E_{s\pi\sigma}\left(\liminf_{T\to\infty} \tfrac{1}{T} \textstyle\sum_{n=1}^{T} r^k(S_n, I_n, J_n)\right) \tag{1}$$

$$\gamma_\beta^k(s,\pi,\sigma) = E_{s\pi\sigma}\left((1-\beta) \textstyle\sum_{n=1}^{\infty} \beta^{n-1} r^k(S_n, I_n, J_n)\right) \tag{2}$$

where S_n, I_n, J_n are random variables for the state and actions at stage n. Let $\gamma^k(\pi,\sigma)$ and $\gamma_\beta^k(\pi,\sigma)$ denote vectors of rewards with coordinates corresponding to the initial states.
A stationary strategy for a player consists of a mixed action for each state, to be used whenever that state is being visited, regardless of the history. Stationary strategies for player 1 are denoted by $x = (x_1, x_2, \ldots, x_z)$, where x_s is the mixed action to be used in state s. For player 2's stationary strategies we write y. A pair (x,y) of stationary strategies determines a Markov-chain (with transition matrix) $P(x,y)$ on S, where entry (s,t) of $P(x,y)$ is $p(t|s,x_s,y_s) = \sum_{i=1}^{m_s} \sum_{j=1}^{n_s} x_s(i)p(t|s,i,j)y_s(j)$. If we use the notation $r^k(x,y) = (r^k(1,x_1,y_1), r^k(2,x_2,y_2), \ldots, r^k(z,x_z,y_z))$ with $r^k(s,x_s,y_s) = \sum_{i=1}^{m_s} \sum_{j=1}^{n_s} x_s(i)r^k(s,i,j)y_s(j)$ then

$$\gamma_\beta^k(x,y) = (1-\beta)(I - \beta P(x,y))^{-1} r^k(x,y) \tag{3}$$

where I is the identity matrix, and

$$\gamma^k(x,y) = Q(x,y)r^k(x,y) \tag{4}$$

with

$$Q(x,y) = \lim_{T\to\infty} \tfrac{1}{T} \textstyle\sum_{n=1}^{T} (P(x,y))^n \tag{5}$$

It is well-known (cf. Blackwell [1962]) that

$$Q(x,y)P(x,y) = Q(x,y) \tag{6}$$

$$Q(x,y) = \lim_{\beta\uparrow 1} (1-\beta)(I-\beta P(x,y))^{-1} \tag{7}$$

and hence (3), (4) and (7) give

$$\gamma^k(x,y) = \lim_{\beta\uparrow 1} \gamma^k_\beta(x,y) \tag{8}$$

Notice that (5) and (6) imply that row s of $Q(x,y)$ is the unique stationary distribution for the Markov chain $P(x,y)$ starting in state s. A stationary strategy x is called pure if $\#\{i : x_s(i) > 0\} = 1$ for all s. Pure stationary strategies shall be denoted by f and g for players 1 and 2 respectively. It is well-known (cf. Hordijk et al. [1983]) that, when playing against a fixed stationary strategy, a player always has a pure stationary best reply:

$$\forall\beta\ \forall y\ \exists f^1, f^1_\beta\ \forall\pi \left[\gamma^1(f^1,y) \geq \gamma^1(\pi,y) \text{ and } \gamma^1_\beta(f^1_\beta,y) \geq \gamma^1_\beta(\pi,y)\right] \tag{9}$$

Obviously, for player 2's best replies an analogon of (9) holds.

Finally, we wish to mention one more type of strategies, namely Markov strategies. These are strategies that, at any stage of play, prescribe actions that only depend on the current state and stage. Thus, the past actions of the opponent are not being taken into account. Strategies for which these choices do depend on those past actions shall be called history dependent.

2 Zero-sum Stochastic Games

In zero-sum stochastic games it is customary to consider only the payoffs to player 1, which player 1 wishes to maximize and player 2 wants to minimize. Since in the sequel we also consider the zero-sum situation where player 2 is the maximizer and player 1 the minimizer, we shall incorporate the player number in the definitions of value and optimal strategies. Thus, for $k = 1,2$, the k-zerosum game is the stochastic game determined by player k's payoffs, where player k is maximizing his reward while the other player is minimizing player k's reward.

In his ancestral paper on stochastic games Shapley [1953] shows

$$\forall\beta\ \exists v^1_\beta\ \exists x^1_\beta, y^1_\beta\ \forall\pi,\sigma \left[\gamma^1_\beta(x^1_\beta,\sigma) \geq v^1_\beta \geq \gamma^1_\beta(\pi,y^1_\beta)\right] \tag{10}$$

The vector v^1_β is called the β-discounted 1-value and the strategies x^1_β, y^1_β are called stationary β-discounted optimal strategies in the 1-zerosum game. Shapley's proof is based on the observation that v^1_β is the unique solution of the following system of equations:

$$\alpha_s = \text{ val}[(1-\beta)r^1(s,i,j) + \beta\textstyle\sum_t p(t|s,i,j)\alpha_t]_{i=1,j=1}^{m_s\ \ n_s} \quad ,s \in S \tag{11}$$

where val denotes the matrix game value operator.

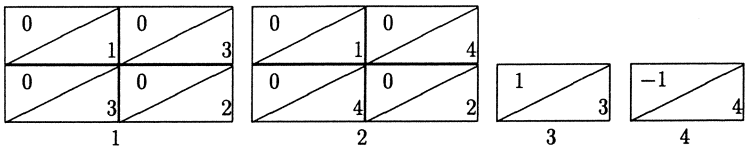
Everett [1957] and Gillette [1957] were the first to consider undiscounted rewards. Everett [1957] examined recursive games, which can be defined as stochastic games where the only non-zero payoffs can be obtained in absorbing states, i.e. states that have the property that once play gets there, it remains there forever. Although optimal strategies need not exist for such games, Everett [1957] shows that for each recursive game the limiting average value v^1 exists, and can be achieved by using stationary ε-optimal strategies $x^1_\varepsilon, y^1_\varepsilon$. Precisely:

$$\exists v^1\ \forall\varepsilon > 0\ \exists x^1_\varepsilon, y^1_\varepsilon\ \forall\pi,\sigma \left[\gamma^1(x^1_\varepsilon,\sigma) + \varepsilon 1_z \geq v^1 \geq \gamma^1(\pi,y^1_\varepsilon) - \varepsilon 1_z\right] \tag{12}$$

Here 1_z denotes the vector $(1,1,\ldots,1)$ in IR^z.

Example 2.1

Consider the following recursive game.



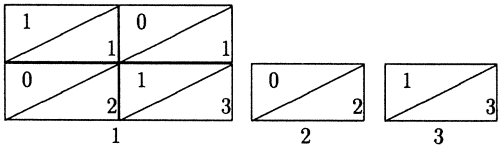
To explain this notation: Player 1 chooses rows; player 2 chooses columns; for each entry the above diagonal number is the payoff to player 1 and the below diagonal number is the state at which play is to proceed; in case of a stochastic transition we write the transition probability vector at this place.

States 3 and 4 are absorbing and obviously states 1 and 2 are the only interesting initial states. For this game the limiting average value is $v^1 = (1,-1,1,-1)$. For player 1 a stationary limiting average ε-optimal strategy is given by $((1-\varepsilon,\varepsilon),(1,0))$ for states 1 and 2 respectively (clearly, in states 3 and 4 he can only choose the one available action). As can be verified using (11), the β-discounted value is $v^1_\beta = (\frac{1-\sqrt{1-\beta^2}}{\beta}, \frac{-1+\sqrt{1-\beta^2}}{\beta}, 1, -1)$ and for player 1 the unique stationary β-discounted optimal strategies are given by playing Top, his first action, with probability $\frac{1-\beta^2-\sqrt{1-\beta^2}}{\beta-\beta^2-\beta\sqrt{1-\beta^2}}$ in state 1 as well as in state 2.

An elementary proof for Everett's [1957] result is given by Thuijsman & Vrieze [1992], where for the recursive game situation a stationary limiting average ε-optimal strategy is constructed from an arbitrary sequence of stationary β_n-discounted optimal strategies, with $\lim_{n\to\infty} \beta_n = 1$.

Example 2.2

This famous game is the so called big match introduced by Gillette [1957].



For this game the unique stationary β-discounted optimal strategies are given by $x^1_\beta = (\frac{1}{2-\beta}, \frac{1-\beta}{2-\beta})$ and $y^1_\beta = (\frac{1}{2}, \frac{1}{2})$ for players 1 and 2 respectively, and $v^1_\beta = \frac{1}{2}$ for initial state 1. However, it was not clear for a long time, whether or not the limiting average value would exist. The problem was that against any Markov strategy for player 1 and for any $\varepsilon > 0$ player 2 has a Markov strategy such that player 1's limiting average reward is less than ε. On the other hand, player 2 can guarantee that he has to pay a limiting average reward of at most $\frac{1}{2}$, but he cannot guarantee anything less than $\frac{1}{2}$. Hence there is an apparent gap between the amounts the players can guarantee using only Markov strategies. The matter was settled by Blackwell & Ferguson [1968], who formulated, for arbitrary $\varepsilon > 0$, a history dependent strategy for player 1 which guarantees a limiting average reward of at least $\frac{1}{2} - \varepsilon$

against any strategy of player 2. This history dependent limiting average ε-optimal stratey is of the following type. At stage n suppose that play is still in state 1 where player 2 has chosen left $l(n)$ times, while he has chosen right $r(n)$ times. Then, player 1 should play Bottom (his second action) with probability $\varepsilon^2(1-\varepsilon)^{k(n)}$, where $k(n) = \max\{0, l(n) - r(n)\}$.

Later, this result on the big match was generalized by Kohlberg [1974], who showed that every repeated game with absorbing states has a limiting average value. A repeated game with absorbing states is a stochastic game in which, just like in the big match, all states but one are absorbing.

Finally, by an ingeneous proof Mertens & Neyman [1981] showed that for every stochastic game the limiting average value exist. Their proof exploits the remarkable observation by Bewley & Kohlberg [1976] that the β-discounted value as well as the stationary β-discounted optimal strategies can be expanded as Puiseux series in powers of $1-\beta$. For example, for the above big match we have that $x^1_\beta = (1,0)+(-1,1)(1-\beta)+(1,-1)(1-\beta)^2+(-1,1)(1-\beta)^3+\ldots$

Apart from these general results, specially structured stochastic games have been examined. We already discussed recursive games and repeated games with absorbing states, but we should also mention the following classes: irreducible/unichain stochastic games (cf. Rogers [1969], Sobel [1971], Federgruen [1978]), i.e. stochastic games for which for any pair of stationary strategies the related Markov chain is irreducible/unichain; single controller stochastic games (cf. Parthasarathy & Raghavan [1981]), i.e. games in which the transitions only depend on the actions of one and the same player for all states; switching control stochastic games (cf. Filar [1981], Vrieze et al. [1983]), i.e. games with transitions for each state depending on the action of only one player; perfect information stochastic games (cf. Liggett & Lippman [1969]), where in each state one of the players has only one action available; stochastic games with additive rewards and additive transitions ARAT (cf. Raghavan et al. [1985]), i.e. there are r^k_1, r^k_2, p_1, p_2 such that $r^k(s,i,j) = r^k_1(s,i) + r^k_2(s,j)$ and $p(s,i,j) = p_1(s,i) + p_2(s,j)$ for all s,i,j; stochastic games with separable rewards and state independent transitions (cf. Parthasarathy et al. [1984]), i.e. there are r^k_a, r^k_b, p_a such that $r^k(s,i,j) = r^k_a(s) + r^k_b(i,j)$ and $p(s,i,j) = p_a(i,j)$ for all s,i,j. All these classes admit stationary limiting average optimal strategies. Later, in Thuijsman & Vrieze [1991, 1992] and in Thuijsman [1992] new (and far more simple) proofs were provided for the existence of stationary solutions in several of these classes. Characterizations, in terms of game properties, for the existence of stationary limiting average optimal strategies are provided in Vrieze & Thuijsman [1987], Filar et al. [1991] and Thuijsman [1992].

Before closing this section on optimality we mention the result by Tijs & Vrieze [1986] (also see Vrieze [1987]) who show that for every stochastic game there is for each player a non-empty set of initial states for which a stationary limiting average optimal strategy exists. This proof relies on the Puiseux series work by Bewley & Kohlberg [1976]. A new and direct proof for the same result is given in Thuijsman & Vrieze [1991], Thuijsman [1992]. A detailed study of the possibilities for limiting average optimality by means of stationary strategies can be found in Thuijsman & Vrieze [1993], while in Flesch et al. [1996b] it is shown that the existence of a limiting average optimal strategy implies the existence of stationary limiting average ε-optimal strategies.

3 General-sum Stochastic Games

One of the first persons to examine non-zerosum stochastic games was Fink [1964], who showed the existence of stationary β-discounted equilibria for stochastic games:

$$\forall\beta\ \exists x,y\ \forall\pi,\sigma\ \left[\gamma^1_\beta(\pi,y) \leq \gamma^1_\beta(x,y) \text{ and } \gamma^2_\beta(x,\sigma) \leq \gamma^2_\beta(x,y)\right] \tag{13}$$

Since, by its definition, for the zero-sum situation an equilibrium can only consist of a pair of optimal strategies, the big match (cf. example 2.2) immediately shows that limiting average equilibria do not always exist. Where we introduced ε-optimal strategies for the zero-sum case, we now have to introduce ε-equilibria for the general-sum case. A pair of strategies $(\pi_\varepsilon, \sigma_\varepsilon)$ is called a limiting average ε-equilibrium ($\varepsilon > 0$) if neither player 1 nor player 2 can gain more than ε by a unilateral deviation. To put it precisely

$$\forall \pi, \sigma \; [\gamma^1(\pi, \sigma_\varepsilon) \leq \gamma^1(\pi_\varepsilon, \sigma_\varepsilon) + \varepsilon 1_z \text{ and } \gamma^2(\pi_\varepsilon, \sigma) \leq \gamma(\pi_\varepsilon, \sigma_\varepsilon) + \varepsilon 1_z] \tag{14}$$

The existence of limiting average ε-equilibria for arbitrary general-sum stochastic games has not yet been established. Neither do we know of any counterexample to their existence. The most general results on the existence of equilibria are the following. First it was observed that in every stochastic game there is a non-empty set of initial states for which ε-equilibria exist (cf. Thuijsman & Vrieze [1991], Thuijsman [1992] or Vieille [1993]). Our proof of this result was based on ergodicity properties of a converging sequence of stationary β_n-discounted equilibra, with $\lim_{n\to\infty} \beta_n = 1$. However, the equilibrium strategies are of a behavioral type: at all stages players must take into account the history of past moves of their opponent. Nevertheless, a side-result of this approach was a simple and straightforward proof for the existence of stationary limiting average equilibria for irreducible/unichain stochastic games (which was earlier derived by Rogers [1969], Sobel [1971], Federgruen [1978]).

Concerning the (simultaneous) existence of limiting average ε-equilibria for all initial states, sufficient conditions have been formulated in Thuijsman [1992], which are based on properties of a converging sequence of stationary β_n-discounted equilibra, with $\lim_{n\to\infty} \beta_n = 1$, while in Thuijsman & Vrieze [1997] quite general sufficient conditions have been formulated in terms of stationary strategies, and of observability and punishability of deviations. We call this the threat approach, since the players are constantly checking after each other, and any "wrong" move of the opponent will immediately trigger a punishment. Thus the threats are the stabilizing force in the limiting average equilibria. Using this threat approach existence of ε-equilibria has been shown for repeated games with absorbing states (cf. Vrieze & Thuijsman [1989], where a prototype threat approach is being used), as well as for stochastic games with state independent transitions (cf. Thuijsman [1992]), as well as for stochastic games with three states (cf. Vieille [1993]), as well as for stochastic games with switching control (cf. Thuijsman & Raghavan [1997]), and existence of pure 0-equilibria has been shown for stochastic games with additive rewards and additive transitions (ARAT, cf. Thuijsman & Raghavan [1997]), which includes the perfect information games.

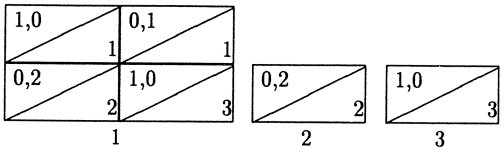
We remark that previous to our threat approach for none of these classes, the existence of limiting average ε-equilibria was known, even though the zero-sum solutions had been derived a long time ago. Also note that even for perfect information stochastic games stationary limiting average equilibria generallly do not exist, although for the zero-sum case pure stationary limiting average optimal strategies are available (cf. Liggett & Lippman [1969]). Example 3.2 below will illustrate this point.

For recursive repeated games with absorbing states (cf. Flesch et al. [1996a]) and for ARAT repeated games with absorbing states (cf. Evangelista et al. [1997]) the existence of stationary limiting average ε-equilibria has been shown (without threats).

We conclude this paper with three very special examples. In example 3.1 we examine a repeated game with absorbing states for which there is a gap between the β-discounted equilibrium rewards and the limiting average equilibrium rewards. In example 3.2 we discuss a perfect information stochastic game which does not have stationary limiting average ε-equilibria, but where the only equilibria known to us, are of the threat type. In example 3.3 we discuss a three person recursive repeated game with absorbing states for which the only

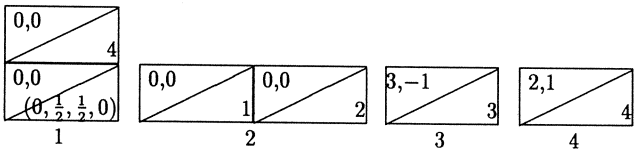
limiting average equilibria consist of cyclic Markov strategies. This is very remarkable since, in that game, neither history dependent nor stationary limiting average ε-equilibria do exist.

Example 3.1



This is an example of a repeated game with absorbing states, where play remains in the initial state 1 as long as player 1 chooses Top, but play reaches an absorbing state as soon as player 1 ever chooses Bottom. Sorin [1986] examined this example in great detail. The 1-zero-sum and 2-zero-sum limiting average values (for initial state 1) are given by $(v^1, v^2) = (\frac{1}{2}, \frac{2}{3})$. Clearly then, there can be no stationary limiting average ε-equilibrium, because against any stationary strategy of player 1, player 2 can get at least 1, and by doing so player 1 would get $0 < v^1$, which he can always achieve by playing limiting average ε-optimal in the 1-zero-sum game. However, for each pair in $\text{Conv}\{(\frac{1}{2}, 1), (\frac{2}{3}, \frac{2}{3})\}$, where Conv stands for convex hull, Sorin [1986] gives history dependent limiting average ε-equilibria that yield this pair as an equilibrium reward. Besides, he shows that any limiting average ε-equilibrium corresponds to a reward in $\text{Conv}\{(\frac{1}{2}, 1), (\frac{2}{3}, \frac{2}{3})\}$, while all β-discounted equilibria yield $(\frac{1}{2}, \frac{2}{3})$. Although this observation suggests that the limiting average general-sum case can not be approached from the β-discounted general-sum case, by studying this example Vrieze & Thuijsman [1989] discovered a general principle to construct, starting from any arbitrary sequence of stationary β_n-discounted equilibria with $\lim_{n\to\infty} \beta_n = 1$, a limiting average ε-equilibrium.

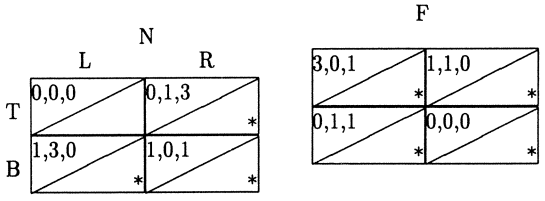
Example 3.2



This game is a recursive perfect information game for which there is no stationary limiting average ε-equilibrium. One can prove this as follows. Suppose player 2 puts positive weight on Left in state 2, then player 1's only stationary limiting average ε-best replies are those that put weight at most $\frac{\varepsilon}{2-\varepsilon}$ on Top in state 1; against any of these strategies, player 2's only stationary limiting average ε-best replies are those that put weight 0 on Left in state 2. So there is no stationary limiting average ε-equilibrium where player 2 puts positive weight on Left in state 2. But there is neither a stationary limiting average ε-equilibrium where player 2 puts weight 0 on Left in state 2, since then player 1 should put at most 2ε weight on Bottom in state 1, which would in turn contradict player 2's putting weight 0 on Left. Following the construction of Thuijsman & Raghavan [1997], where existence of limiting average 0-equilibria is shown for arbitrary n-person games with perfect informtion, we can find an equilibrium by the following procedure. Take a pure stationary limiting average optimal strategy f^1 for player 1 (this exists by Liggett & Lippman [1969]); let g^1 be pure stationary limiting average optimal for player 2 in the 1-zero-sum game; let g^2 be a pure

stationary limiting average best reply for player 2 against f^1 in the 2-zero-sum game (which exists by (9)). Now define g^* for player 2 by: play g^2 unless at some stage player 1 has ever deviated from playing f^1, then play g^1. Here, $f^1 = (1,0) = g^2$ and $g^1 = (0,1)$. Now it can be verified that (f^1, g^*) is a limiting average equilibrium.

Example 3.3



This is a three-person recursive repeated game with absorbing states, where an asterisk in any particular entry denotes a transition to an absorbing state with the same payoff as in this particular entry. There is only one entry for which play will remain in the non-trivial initial state. One should picture the game as a $2 \times 2 \times 2$ cube, where the layers belonging to the actions of player 3 (Near and Far) are represented separately. As before, player 1 chooses Top or Bottom and player 2 chooses Left or Right. The entry (T, L, N) is the only non-absorbing entry for the initial state. Hence, as long as play is in the initial state the only possible history is the one where entry (T, L, N) was played at all previous stages. This rules out the use of any non-trivial history dependent strategy for this game. Therefore, the players only have Markov strategies at their disposal. In Flesch et al. [1997] it is shown that, although (cyclic) Markov limiting average 0-equilibria exist for this game, there are no stationary limiting average ε-equilibria. Moreover, the set of all limiting average equilibria is being characterized completely. An example of a Markov equilibrium for this game is (π, σ, τ), where π is defined by: at stages 1, 4, 7, 10, ... play T with probability $\frac{1}{2}$ and at all other stages play T with probability 1. Similarly, σ is defined by: at stages 2, 5, 8, 11, ... play L with probability $\frac{1}{2}$ and at all other stages play L with probability 1. Likewise, τ is defined by: at stages 3, 6, 9, 12, ... play N with probability $\frac{1}{2}$ and at all other stages play N with probability 1. The limiting average reward corresponding to this equilibrium is $(1,2,1)$.

References

Bewley T, Kohlberg E [1976] The asymptotic theory of stochastic games. *Math Oper Res* **1**: 197–208.

Blackwell D [1962] Discrete dynamic programming. *Ann Math Statist* **33**: 719–726.

Blackwell D, Ferguson TS [1968] The big match. *Ann Math Statist* **39**: 159–163.

Evangelista FS, Raghavan TES, Vrieze OJ [1997]: ARAT repeated games with absorbing states. In: Volume in Honor of D Blackwell (forthcoming).

Everett H [1957] Recursive games. In: M. Dresher, A.W. Tucker & P. Wolfe (eds), *Contributions to the Theory of Games* **III**, *Annals of Mathematical Studies* **39**, Princeton University Press, Princeton, 47–78.

Federgruen A [1978] On N-person stochastic games with denumerable state space. *Adv Appl Prob* **10**: 452–471.

Filar JA [1981] Ordered field property for stochastic games when the player who controls transitions changes from state to state. *J Opt Theory Appl* **34**: 503–515.

Filar JA, Schultz TA, Thuijsman F, Vrieze OJ [1991] Non-linear programming and stationary equilibria in stochastic games. *Math Progr* **50**: 227–237.

Fink AM [1964] Equilibrium in a stochastic n-person game. *J Sci Hiroshima Univ*, Series A-I **28**, 89–93.

Flesch J, Thuijsman F, Vrieze OJ [1996a] Recursive repeated games with absorbing states. *Math Oper Res* **21**: 1016–1022.

Flesch J, Thuijsman F, Vrieze OJ [1996b] Simplifying optimal strategies in stochastic games. Report M96-11, Department of Mathematics, Maastricht University.

Flesch J, Thuijsman F, Vrieze OJ [1997] Cyclic Markov equilibria in a cubic game. *Int J Game Theory* (forthcoming).

Gillette D [1957] Stochastic games with zero stop probabilities. In: Dresher M, Tucker AW, Wolfe P (eds), *Contributions to the Theory of Games* **III**, *Annals of Mathematical Studies* **39**, Princeton University Press, Princeton, pp 179–187.

Hordijk A, Vrieze OJ, Wanrooij GL [1983] Semi-Markov strategies in stochastic games. *Int J Game Theory* **12**: 81–89.

Kohlberg E [1974] Repeated games with absorbing states. *Annals of Statistics* **2**: 724–738.

Liggett TM, Lippman SA [1969] Stochastic games with perfect information and time average payoff. *SIAM Review* **11**: 604–607.

Mertens JF, Neyman A [1981] Stochastic games. *Int J Game Theory* **10**: 53–66.

Nash J [1951] Non-cooperative games. *Annals of Mathematics* **54**: 286–295.

Parthasarathy T, Raghavan TES [1981] An orderfield property for stochastic games when one player controls transition probabilities. *J Opt Theory Appl* **33**: 375–392.

Parthasarathy T, Tijs SH, Vrieze OJ [1984] Stochastic games with state independent transitions and separable rewards. In: Hammer G, Pallaschke D (eds), *Selected Topics in Operations Research and Mathematical Economics*, Springer Verlag, Berlin, pp 262–271.

Raghavan TES, Tijs SH, Vrieze OJ [1985] On stochastic games with additive reward and transition structure. *J Opt Theory Appl* **47**: 451–464.

Rogers PD [1969] Non-zerosum stochastic games. PhD thesis, report ORC 69-8, Operations Research Center, University of California, Berkeley.

Shapley LS [1953] Stochastic games *Proc Nat Acad Sci USA* **39**: 1095–1100.

Sobel MJ [1971] Noncooperative stochastic games. *Ann Math Statist* **42**: 1930–1935.

Sorin S [1986] Asymptotic properties of a non-zerosum stochastic game. *Int J Game Theory* **15**: 101–107.

Thuijsman F [1992] *Optimality and Equilibria in Stochastic Games*. CWI-tract **82**, Center for Mathematics and Computer Science, Amsterdam.

Thuijsman F, Raghavan TES [1997] Perfect information stochastic games and related classes. *Int J Game Theory* (forthcoming).

Thuijsman F, Vrieze OJ [1991] Easy initial states in stochastic games. In: Raghavan TES et al (eds), *Stochastic Games and Related Topics*, Kluwer Academic Publishers, Dordrecht, pp 85–100.

Thuijsman F, Vrieze OJ [1992] Note on recursive games. In: Dutta B et al (eds), *Game Theory and Economic Applications, Lecture Notes in Economics and Mathematical Systems* **389**, Springer, Berlin, pp 133–145.

Thuijsman F, Vrieze OJ [1993] Stationary ε-optimal strategies in stochastic games. *OR Spektrum* 15, 9–15.

Thuijsman F, Vrieze OJ [1997] The power of threats in stochastic games. In: Bardi et al (eds), *Stochastic Games and Numerical Methods for Dynamic Games*, Birkhauser, Boston (forthcoming).

Tijs SH, Vrieze OJ [1986] On the existence of easy initial states for undiscounted stochastic games. *Math Oper Res* **11**: 506–513.

Vieille N [1993] Solvable states in stochastic games. *Int J Game Theory* **21**: 395–404.

Von Neumann J [1928] Zur Theorie der Gesellschafsspiele. *Mathematische Annalen* **100**: 295–320.

Vrieze OJ [1987] *Stochastic Games with Finite State and Action Spaces*. CWI-tract **33**, Center for Mathematics and Computer Science, Amsterdam.

Vrieze OJ, Thuijsman F [1987] Stochastic games and optimal stationary strategies, a survey. In: Domschke W et al (eds), *Methods of Operations Research* **57**:513–529.

Vrieze OJ, Thuijsman F [1989] On equilibria in repeated games with absorbing states. *Int J Game Theory* **18**: 293–310.

Vrieze OJ, Tijs SH, Raghavan TES, Filar JA [1983] A finite algorithm for the switching control stochastic game. *OR Spektrum* **5**: 15–24.

STABLE SETS IN NON-COOPERATIVE GAME THEORY

Dries Vermeulen
Department of Economics
Maastricht University
E-mail: a.vermeulen@ke.unimaas.nl

1. Introduction

Generally speaking, game theory deals with the analysis of situations in which a number of people, each of them having his own interest in the outcome of the situation, must decide what to do. In the area of non-cooperative game theory these people (the players) are supposed to be unable or unwilling to commit themselves to taking specific actions in the given situation (the game). This non-cooperative character of the players forces us to search for solutions of the game that are self-enforcing.

In their paper of 1986 Kohlberg and Mertens argued that a convincing self-enforcing solution must necessarily satisfy a number of conditions. A solution that is designed to meet all conditions is called a stability concept. They also showed that such a concept is inevitably set-valued. Since then a number of stability concepts have been proposed. Although there is nowadays a wide variety of stability concepts, the two that are most commonly known are the ones defined by Mertens (1989) and Hillas (1990).

In this paper the general framework is introduced first. Then some of the conditions proposed by Kohlberg and Mertens are reviewed. Finally some of the results concerning the stability concepts of Mertens and (especially) Hillas are given.

2. The name of the game

Each of the players has a finite number of choices in the particular type of non-cooperative game considered in this paper. Further, the players are supposed to make their choices simultaneously. (To put it a little differently, no player has any information concerning the choices of the other players when he has to make his own choice.) After these choices are made, each player receives a *payoff*. The obvious goal of any player of the game is to obtain a payoff that is as high as possible. Such a game is usually called a one-shot game, or normal form game.

Example 1. If there are two players involved in the game, we can represent a normal form game by what is called a bimatrix. For example,

$$\begin{bmatrix} (1,2) & (0,0) \\ (0,0) & (0,0) \end{bmatrix}$$

is a 2×2-bimatrix game. Player I (the row player) has to choose between the first and the second row, player II (the column player) between the first and the second column. The

choices of the players determine an entry of the bimatrix. If for instance player I chooses his first row and player II chooses the first column, we end up in the entry $(1,2)$. The first number is the payoff to player I, so he gets one guilder. Similarly, player II gets two guilders. ◁

In order to give a general definition, let n be a positive natural number. The *player set* $\{1,\ldots,n\}$ of an *n-person normal form game* Γ (from now on simply called a game) is denoted by N. Each player $j \in N$ is assumed to have a finite set A_j of (pure) strategies and a payoff function $u_j\colon A := \prod_i A_i \to \mathbb{R}$. Thus, using the shorthand notation $u = (u_i)_{i\in N}$, the game can be written as $\Gamma = \langle A, u\rangle$.

Given a game $\Gamma = \langle A, u\rangle$, a player, say $j \in N$, can decide to randomize between the choices in A_j that are available to him. The game that results when randomization is explicitly allowed is called the *mixed extension* of Γ. In this paper we will simply identify the game Γ with its mixed extension.

In the mixed extension each player $i \in N$ is assumed to choose a *mixed strategy* $x_i := (x_{ia})_{a\in A_i}$ from the set $\Delta(A_i)$ of probability distributions on A_i. So, the coordinates of x_i are not negative and their sum equals one. Given the strategy *profile* $x := (x_i)_{i\in N}$ of choices of the players, player j calculates his *expected* payoff as follows. A pure strategy profile $a = (a_i)_{i\in N} \in A$ is played with probability $\prod_i x_{ia_i}$. So, player j expects to get a payoff equal to $u_j(x) := \sum_{a\in A} \prod_i x_{ia_i} u_j(a)$ when x is played. Clearly, the payoff function of this mixed extension is a function from the set $\Delta_A := \prod_i \Delta(A_i)$ of (mixed) strategy profiles to $\mathbb{R}$.

Example 2. A mixed strategy of player I in the bimatrix game of example 1 can be written as $(p, 1-p)$ for $0 \le p \le 1$. Playing such a strategy $(p, 1-p)$ means that the first row is chosen with probability p and the second one with probability $1-p$. Playing $(1,0)$ obviously means that the first row is chosen with certainty. ◁

The central issue in this paper is the search for a 'reasonable' solution for each game. This means that we need to answer two questions. First of all, what do we mean by a solution? Secondly, what do we consider to be a 'reasonable' solution? The first question answered in the following definition. We specifically allow *sets* of strategy profiles in the solution, because the conditions for 'reasonable' solutions in the next section necessarily yield set-valued solutions as we will see in example 6.

Definition 1. A *solution* σ is a rule that assigns to each game Γ a collection $\sigma(\Gamma)$ of (usually closed and non-empty) subsets of the space of strategy profiles of the game Γ. The elements of $\sigma(\Gamma)$ are called the *solution sets* of the game Γ.

Example 3. The map τ that assigns the collection $\{\{x\} \mid x \text{ is a strategy profile of } \Gamma\}$ to a game Γ is a solution by definition. However, this is not a very 'reasonable' one. If we look for instance to the game of example 1, it is clear that the strategy pair $((0,1),(1,0))$ is not a good deal for player I. If he agrees to play this profile his payoff will be zero, while he will get one if he decides to play $(1,0)$ instead (provided that player II will stick to the agreement). ◁

Thus, we need to specify what 'reasonable' means for a solution. And, although this question does not seem to pose too much problems, it turns out to be a very difficult one indeed. In the next section we will give some examples of possible answers to this question.

3. Requirements for solutions

Usually in non-cooperative game theory 'reasonable' is interpreted as what is called 'self-enforcing'. A set of strategy profiles of a given game is called self-enforcing if no player of the game *wishes* to deviate to a strategy profile outside the set once it has been *agreed* not to play any of the profiles outside this set. Such a condition is obviously necessary for a solution set of a non-cooperative game, since the players are not *obliged* to stick to any agreement as we supposed in the model. So, the motivation to do so anyway must lie within the agreement itself. The first natural formalization of self-enforcingness is the equilibrium concept introduced by Nash in 1952.

Definition 2. For a game $\Gamma = \langle A, u \rangle$, let $(x \mid y_i)$ denote the strategy profile where player i uses the strategy $y_i \in \Delta(A_i)$ and his opponents use the strategies in $x \in \Delta_A$. For player i and a strategy profile x

$$\beta_i(x) := \{y_i \in \Delta(A_i) \mid u_i(x \mid y_i) \geq u_i(x \mid z_i) \quad \text{for all } z_i \in \Delta(A_i)\}$$

is the set of *best replies* of player i against x. A strategy profile in the set

$$\beta(x) := \prod_i \beta_i(x)$$

is also called a *best reply* against x. A (Nash) *equilibrium* is a strategy profile x that is a best reply against itself. In other words, x must be an element of $\beta(x)$. The associated solution assigns to each game Γ the collection $\{\{x\} \mid x \text{ is an equilibrium of } \Gamma\}$.

At first sight, equilibria seem to catch the spirit of self-enforcingness quite nicely. After all, each player plays 'as good as he can' (i.e., a best reply) given the strategy choices of his opponents in a Nash equilibrium. However, some Nash equilibria are not as convincing as we would expect a 'reasonable' solution to be. This will be explained in the next

Example 4. First note that the strategy pair $((0,1),(0,1))$ is a Nash equilibrium of the bimatrix game in example 1. This is evident, once we realize that *any* strategy of player I (player II) is a best reply against the strategy $(0,1)$ of player II (player I).
However, suppose that player II plays a given strategy $(q, 1-q)$. Then the expected payoff of player I is q when he plays his first row, while he will get zero when he plays his second row. So, playing $(1,0)$ will always give him *at least* the payoff he gets when he plays $(0,1)$, the above-mentioned equilibrium strategy. For this reason $(0,1)$ is called a weakly *dominated* strategy. ◁

For this reason attempts have been made in the past to find methods to eliminate 'bad' equilibria like the one in the above example. The idea was to refine the collection of equilibria of a game to a smaller set of acceptable equilibria by imposing extra conditions. Probably the two most notorious examples of such *refinements* of the Nash equilibrium solution are *perfect* equilibria defined by Selten (1975) and *proper* equilibria defined by Myerson (1978).

Traditionally, a definition of a refinement was given, after which one tried to prove that it did not suffer from flaws like the one described above. Unfortunately all refinements introduced so far have their own specific shortcomings.

In their seminal paper of 1986, Kohlberg and Mertens broke with this tradition. They first composed a list of (initially seven) properties they thought to be essential for any reasonable solution. After that they tried to find a solution that satisfies all of their

requirements. Although it lies outside the scope of this contribution to describe all seven requirements in detail, we will describe some of them.

EXISTENCE This basic requirement states that every game should have at least one solution set.

ADMISSIBILITY The admissibility of a strategy profile can be seen as a natural strengthening of the notion of undominatedness. The requirement states that, given a game, *any* strategy profile in *any* solution set of the game should be admissible, and in particular not use (weakly) dominated strategies.

Example 4 shows that Nash equilibria are not always admissible since, as is already said, weakly dominated strategies are not used in admissible profiles. On the other hand, it is a well-known fact that perfect equilibria are admissible (see the Appendix for a precise definition of perfect equilibria and admissibility). We will nevertheless show that perfect equilibria do not satisfy the next requirement.

DELETION OF A BAD STRATEGY The philosophy behind this requirement is as follows. Suppose that a player of a game has a 'bad' pure strategy, for instance a dominated one. Then it is first of all reasonable to suppose that this strategy is not used in any solution set of that game, like we did in the previous requirement. One can however take the argument even further. Since every opponent of this player *knows* that the pure strategy under consideration is a bad one, the opponents also know that the player will never use his bad strategy. Thus one might argue that any solution set of the game should also be a solution set of the game that results when the bad strategy is eliminated from the set of pure strategies of the player in question.

Example 5. We will show that perfect equilibria do not survive the deletion of a weakly dominated strategy. Consider the 2×3-bimatrix game

$$\Gamma = \begin{bmatrix} (1,2) & (1,1) & (0,1) \\ (1,1) & (0,0) & (1,0) \end{bmatrix}.$$

The strategy pair $((0,1),(1,0,0))$ is perfect since the sequence $((\frac{1}{k}, 1-\frac{1}{k}),(1-\frac{3}{k}, \frac{1}{k}, \frac{2}{k}))_{k\in\mathbb{N}}$ is $\frac{2}{k}$-perfect and converging to $((0,1),(1,0,0))$. Now note that the third column is dominated by the first column for player II. The game that results from the deletion of the third column is

$$\Gamma' = \begin{bmatrix} (1,2) & (1,1) \\ (1,1) & (0,0) \end{bmatrix}.$$

Furthermore, $((0,1),(1,0))$ is the strategy pair in this new game that corresponds to the original strategy pair $((0,1),(1,0,0))$. However, playing the second row has now become a weakly dominated strategy for player I. Hence, the strategy pair cannot be perfect, since a perfect equilibrium is admissible, and does therefore not use a weakly dominated strategy. ◁

INVARIANCE In general, invariance means that 'similar' games should have the 'same' solution sets. In this case, the similarity of two games refers to the deletion of pure strategies of one (large) game that are duplicates of other (possibly mixed) strategies. The iteration of this deletion process yields a second (smaller) game. The formal connection between these two games is given in

Definition 3. Let $\Gamma = \langle A, u\rangle$ and $\Gamma' = \langle B, v\rangle$ be two games. A map $f = (f_i)_{i\in N}$ from Δ_B to Δ_A is called a *reduction map* from Γ' to Γ if for every player i,

(1) $f_i: \Delta(B_i) \to \Delta(A_i)$ is linear and onto and
(2) $v_i = u_i \circ f$.

The two games Γ and Γ' are similar in the sense that for every player j and every strategy $z_j \in \Delta(B_j)$ of this player in the game Γ', the strategy $f_j(z_j)$ gives him the same payoff in Γ as z_j does in Γ', no matter what the other players do (using the functions f_i to transform *their* strategies from one game to the other). The two games do however have different strategy spaces. So, now the problem is, what do we mean by the 'same' solution sets? This question can be answered as follows.

Definition 4. A solution σ is called *invariant* if for any pair of games Γ and Γ', and any reduction map f from Γ' to Γ we have

$$\sigma(\Gamma) = \{f(S) \mid S \in \sigma(\Gamma')\}$$

and

$$f^{-1}(T) = \bigcup\{S \in \sigma(\Gamma') \mid f(S) = T\} \qquad \text{for all } T \in \sigma(\Gamma).$$

These equalities state that every solution set of the larger game Γ' projects (via f) onto a solution set of the smaller game Γ and, moreover, that every strategy profile of the larger game that projects into a solution set T of the smaller game is an element of a solution set S of the larger game that projects onto T.

4. Stable sets

The main problem of Kohlberg and Mertens was that in 1986 no solution was known that satisfied all of their requirements. Thus, the central question in their paper was:

DOES THERE EXIST A SOLUTION THAT SATISFIES ALL REQUIREMENTS?

A solution that is designed to satisfy all requirements is usually called a *stability concept.* The solution sets of such a concept are called *stable sets.*

Although Kohlberg and Mertens did not manage to find a stability concept that satisfies all requirements, at least two observations can be made concerning the way they searched for one. First of all they showed that their requirements inevitably led to a set-valued solution.

Example 6. Consider the game Γ introduced in example 5. Let S be a solution set of this game according to some solution σ satisfying all requirements. We will show that S contains at least two strategy profiles. To this end, note that, by existence and admissibility,

$$\{((1,0),(1,0))\}$$

is the unique solution set of Γ', since playing the first row (column) is the only undominated strategy of player I (player II). Thus, since σ satisfies the deletion of a bad strategy by assumption, the set S must contain the strategy pair $((1,0),(1,0,0))$. However, the same line of reasoning applied to the deletion of the (dominated) second column shows that S must also contain $((0,1),(1,0,0))$. For this particular reason solutions are allowed to be set-valued in this paper. ◁

Secondly, they introduced a specific method to generate stability concepts. Roughly speaking, they constructed for every game Γ a neighborhood of 'perturbed games' and then said that a set of strategy profiles of Γ is stable if for any possible perturbation there is at least one strategy profile in the set that survives the perturbation.

Formally, let $\Gamma = \langle A, u \rangle$ be a game. Let $\mathcal{P}$ be a set of 'perturbed games', equiped with some metric d. It is assumed that each perturbed game $\Gamma' \in \mathcal{P}$ has a non-empty set $E(\Gamma')$ of equilibria and that Γ is an element of $\mathcal{P}$.

Definition 5. A closed set $S \subset \Delta_A$ is called a *$\mathcal{P}$-set* of the game Γ if for any neighborhood V of S there exists a number $\delta > 0$ such that $E(\Gamma') \cap V$ is not empty if $d(\Gamma, \Gamma') < \delta$.

Remark. The sets satisfying this condition are not themselves called stable since, for example, Δ_A also satisfies the condition. The sets that will actually be called stable also need to be sufficiently small. The selection of sufficiently small $\mathcal{P}$-sets can be done in various ways.

Obviously, the resulting stability concept depends on the choices of the specific set $\mathcal{P}$ and the metric d. Using a variation on this theme Mertens (1989) was able to construct a stability concept that did indeed satisfy all requirements.

Example 7. In 1990 Hillas also introduced a stability concept. He first identified a game $\Gamma = \langle A, u \rangle$ with its *best reply correspondence* $\beta: \Delta_A \twoheadrightarrow \Delta_A$ defined by

$$\beta(x) := \prod_i \beta_i(x).$$

The set of perturbed games was chosen to be the collection $\mathcal{C}$ of all compact- and convex-valued upper hemicontinuous (uhc) correspondences $\varphi: \Delta_A \twoheadrightarrow \Delta_A$.
It is not difficult to show that β is such a correspondence. So, the game Γ is indeed an element of $\mathcal{C}$ given the above identification. Further, for a correspondence $\varphi \in \mathcal{C}$, the set of fixed points $\mathrm{fix}(\varphi) := \{x \in \Delta_A \mid x \in \varphi(x)\}$ of φ serves as the set of equilibria of the 'perturbed game' φ. This is in agreement with the identification of Γ with β, since $\mathrm{fix}(\beta)$ equals the set of equilibria of Γ.
The metric on $\mathcal{C}$ is based on the Hausdorff distance d_H with, for $X, Y \subset \Delta_A$,

$$d_H(X,Y) := \inf\{\varepsilon > 0 \mid X \subset B_\varepsilon(Y) \text{ and } Y \subset B_\varepsilon(X)\}.$$

Then the pointwise Hausdorff metric d on $\mathcal{C}$ is defined by

$$d(\varphi, \psi) := \sup\{d_H(\varphi(x), \psi(x)) \mid x \in \Delta_A\} \qquad (\varphi, \psi \in \mathcal{C}).$$

Now a closed set $S \subset \Delta_A$ is called a *$\mathcal{C}$-set* of the game Γ if for any neighborhood V of S there exists a number $\delta > 0$ such that $\mathrm{fix}(\varphi) \cap V$ is not empty if $d(\beta, \varphi) < \delta$. Finally, a $\mathcal{C}$-set that is *minimal* (w.r.t. set inclusion) within the collection of $\mathcal{C}$-sets is called stable in the sense of Hillas. ◁

Results. Hillas (1990) showed that the solution that assigns to each game Γ its collection $\tau(\Gamma)$ of stable sets in the sense of Hillas satisfies EXISTENCE, ADMISSIBILITY and, moreover BACKWARD INDUCTION and CONNECTEDNESS. Later on Hillas and I were able to prove that it also satisfies a strong variant of DELETION OF A BAD STRATEGY.

In my thesis a certain type of stable sets is introduced whose definition stays closer to known game theoretic notions while it satisfies the same requirements. Looking back, this is not so surprising, since Hillas, Jansen, Potters and I were able to prove that both definitions yield the same stability concept. Using this equivalence we also showed each stable set the sense of Mertens contains a stable set in the sense of Hillas.

I also managed to construct a counterexample for the INVARIANCE of this stability concept. The example moreover shows that this is not due to Hillas' particular choice $\mathcal{C}$ of perturbed

games. It is merely caused by the choice of *minimal* $\mathcal{C}$-sets as final solution sets. The following selection method however does yield an invariant solution.

Let Γ be a game and let S be a $\mathcal{C}$-set of this game. Let us call S *extendable* if, for all games Γ' and all reduction maps f from Γ' to Γ, the set $f^{-1}(S)$ is a $\mathcal{C}$-set of Γ'.
Now a non-empty and closed set S of strategy profiles of Γ is called stable if it is extendable, connected and consists entirely of perfect equilibria. It can be shown that this solution satisfies all requirements of Kohlberg and Mertens.

5. Appendix

Definition of admissibility. Let $\Gamma = \langle A, u \rangle$ be a game. A strategy profile z in Δ_A is called *completely mixed* if all coordinates z_{ia} are positive. For player i, a strategy y_i is an *admissible* best reply against a strategy profile x if there is a sequence $(x^k)_{k\in\mathbb{N}}$ of completely mixed strategy profiles converging to x such that y_i is a best reply against x^k for all k.
Now we say that a solution σ satisfies admissibility if, for every game Γ, every solution set S in $\sigma(\Gamma)$ and every strategy profile x in S, the strategy x_i is admissible for every player i.

Definition of perfect equilibria. Let $\eta > 0$ and let $x \in \Delta_A$ be a completely mixed strategy profile. Then x is called *η-perfect* if for all players $i \in N$ and for all $a \in A_i$ we have that $x_{ia} \leq \eta$ whenever a is not a best reply against x.
A strategy profile $x \in \Delta_A$ is called *perfect* if there exist a sequence $(\eta_k)_{k\in\mathbb{N}}$ of positive real numbers converging to zero and a sequence $(x^k)_{k\in\mathbb{N}}$ of strategy profiles in Δ converging to x, such that x^k is η_k-perfect for all k.

References

Hillas, J. (1990) *On the definition of the strategic stability of equilibria*, Ectra **58**, 1365 – 1391.

Kohlberg, E., and J.F. Mertens (1986) *On strategic stability of equilibria*, Ectra **54**, 1003 – 1037.

Mertens, J.F. (1989) *Stable equilibria – a reformulation. Part I: Definitions and basic properties*, MOR **14**, 575 – 625.

Vermeulen, Dries (1996) *Stability in non-cooperative game theory*, Dissertation, Department of Mathematics, University of Nijmegen.

Vermeulen A.J., Potters J.A.M. and M.J.M. Jansen (1997) *On stable sets of equilibria*, Bangalore Conference Volume, Kluwer Academic Publishers, The Netherlands (forthcoming).

Part IV

Applications

A Review of Applications of Operations Research in the Netherlands

Rommert Dekker
Econometric Institute
Erasmus University Rotterdam
E-mail: rdekker@few.eur.nl

1 Introduction

In this paper we present a short overview of applications of operations research (OR) in the Netherlands over the years 1985-1997. We concentrate on those areas where scientific research was the basis for the application. We do not want to report on each individual application, instead we focus on some main areas. The selection of applications may be somewhat subjective, since the author can not overview every company in the Netherlands. The idea is to give an overview about the developments going on. In that sense this paper builds on earlier reviews of the use of OR in the Netherlands.

Several publications document the history of the use of OR in the Netherlands. The first ten years (up to 1968) were described in a pocket-book edited by Lombaers et al. [4]. In 1981 a compilation of articles was published giving an overview of the use of OR in industry, transportation, banks and hospitals (ref. [3]). Slightly later, in 1983, a large compilation of successes and failures of OR was published [2] to celebrate the first 25 years of the section Operations Research of the Dutch Society for Statistics and OR (VVS). The most recent publication describing a number of applications of OR in detail is the book by Fortuin et al. [1]. We incorporate it here since many of the applications are Dutch.

To understand the role of OR in practice we first discuss the state of OR groups in companies and knowledge centres. This limitation is somewhat unfortunate since OR is also applied outside these groups. Yet the identification of an application as belonging to operations research is often related to the fact that its originates from an OR group or that it is purchased from an OR oriented company. In the following section we discuss the information-technological developments which stimulated the use of OR in the last decade. Subsequently we discuss a number of areas to which OR has had a substantial contribution. We conclude with a general discussion on applications of OR and the opportunities which lie ahead.

2 Operations research groups in the Netherlands

Operations research groups existed both in companies and knowledge institutes. We will briefly discuss each of them.

2.1 Companies

Roughly speaking, operations research groups existed only in large companies or in small companies specialized in OR.

Separate OR groups could be found in large manufacturing companies, like Shell, Hoogovens and Philips, in banks, like ABN-AMRO, ING and RABO, in the post and telecommunication company KPN, in the main airline KLM and in the container transhipment company ECT.

Shell had both a research group in their Amsterdam laboratory and a consulting group in their Central Office in The Hague. Whereas over the years the latter got a more computer science profile, the first was reduced in size. Philips privitized their OR, logistics and statistics department (CQM). Also at the Dutch steelworks Hoogovens the OR group was threatened to be dissolved. These developments can be understood from the general business climate, in which companies were concentrating on their core business and in which staff groups were under pressure. In contrast with the preceding decade in which many blue collar workers lost their jobs, it was now the turn of the white collar workers. Concluding we can say that the OR groups in most manufacturing companies were under severe pressure.

OR in the banks survived, although some banks merged. For example Amro merged with ABN and the NMB bank was taken over by the ING group. The RABO group still has some OR people in their subsidiary Rabofacet.

At KLM and at the other Dutch airlines operations research was increasingly used in many areas, like flight planning, crew rostering, revenue management and aircraft maintenance. Not all OR was done in-house, large parts were contracted out to specialized companies. At container transhipment company ECT the OR group grew in size, mainly because of their move to automated container terminals which required quite sophisticated quantitative planning and control methods. The transportation sector thus showed to be a healthy area for OR.

OR did not flourish in the large consulting or software companies, like CMG, BSO-Origin, Cap Gemini or KPMG. Although these hired quite some OR graduates, none of these had a separate OR group or was developing special OR products. Most of these companies were either consulting, body shopping or concentrating on the much larger market of information systems. The few OR people in these companies were often acting under the flag of logistic consultants. One small logistic consultancy group with a substantial OR content is the former Nedlloyd subsidiary Logion, which has recently been taken over by NEI. Another recent exception worth mentioning is the Baan Info Systems company. Once her enterprise wide information system Triton became very popular, the company realized that providing more intelligence in her planning and scheduling modules is attractive. An OR group is presently being formed consisting of some recent PhD's in production planning and inventory control.

The few specialized OR companies flourished: we name Ortec Consultants, the Centre for Quantitative Methods (CQM), Paragon Decision Technology, Beyers and Partners, Point Logic Systems and AKB. The first is the largest specialized OR company in the Netherlands, doing applications in many areas, like planning and scheduling in the oil business (SHELL), vehicle routing, crew rostering and finance. It grew in these 10 years from 20 to 70 people. The CQM group was formerly a Philips department. Although it had a difficult time during its privitization it managed to grow afterwords. Many PhDs in OR found a job in that company. Some other companies worth mentioning are Paragon Decision technology, which developed a Windows-based algebraic modelling system for linear programming (the AIMMS package) and AKB which has a long reputation in OR and presently markets a

vehicle routing package. Finally, the two smallest companies are the Dutch subsidiary of Beyers and Partners, who is specialised in production planning, cutting systems, and markets an in-house developed LP package (OMP) and Point Logic Systems who specializes in marketing optimization.

The conclusion of this overview is that there is certainly a market for OR. Apart from some exceptions this market seems to be best served by specialized companies with OR as their core business. These companies are able to grow substantially and are even able to compete on the world market with their products.

2.2 Professional organization

The Dutch Society for Statistics and Operations Research (VVS) is the professional organisation for operations researchers. The section solely devoted to this area (NGB, formerly called SOR) had about 500 members, about half of which were academics. After some increase since 1983, membership stabilized and decreased slightly thereafter. The NGB has always been active in promoting applications of OR by stimulating reviews like the one from Lombaers et al. (1968) and the one by Tilanus et al. (1983). Several applications are documented in her journals Statistica Neerlandica and Kwantitatieve Methoden. Issue 48 of the latter gives an overview of OR in logistics.

2.3 Universities and knowledge centres

Operations research curricula existed at almost all universities and hardly at any other higher educational institutes (the so-called HBO's). Only at one HBO institution a business oriented mathematics curriculum was given (bedrijfswiskunde). The OR curricula were either in the mathematics department or the economics faculty. In the latter case OR was incorporated in the econometrics study and configured under the names 'besliskunde' or 'bedrijfseconometrie'. In every university there was at least one chair in OR and in some two (all technical universities). OR was also done in the management science departments and in one mechanical engineering department (TUT). Student numbers were heavily dropping in the mid nineties, even when the economy boomed and many positions were available. The national network of operations research (LNMB) offered a quite successul educational program. Most research at the universities was concentrated in the so- called "onderzoekscholen". Unfortunately, OR was scattered among many of them: both in mathematical schools, like the Stieltjes institute and the MRI, as well in discrete mathematics (EIDMA), in economic oriented schools, like CENTER and the Tinbergen Institute and finally in two logistic oriented schools, TRAIL and BETA. Primary funding for research from the government showed a slow but steady decrease over the decade. Although the national science foundation (NWO) saw its budget increased, the overall funding from the government decreased. As a result more and more funding was obtained from companies. Pure research had to be very good to remain unaffected by these financial pressures.

Only one knowledge centre did have an OR group. This was the TNO institute for defense research (FEL-TNO) in the Hague. Quite some people were active in various areas, including search and detection strategies, radar scheduling, military logistics (distribution optimization), spare parts control, as well as civil applications like maintenance optimization.

3 Information and computer environment

Developments in information technology and computer hardware in the last decade were very favorable for operations research. The speed of the most popular personal computers, introduced in the eighties, rose from 10 Mhz in 1987 to about 166 MHz, with a halving of the price. The RAM memory for such a computer increased from 1 to 16 Mb. Workstations saw similar increases in speed and performance. These two computer types were especially useful for OR and much larger problems could be tackled. Algorithmic research also led to faster algorithms. In case of the simplex method for example, speed increases of a factor 100 were obtained over the decade.

The whole society also became computer minded and software development became a more mature technology. The advantage for OR was that finally a structured environment arose, with accurate information which is a must for optimization. It does not make sense to put much effort in optimization if the underlying information is not reliable. Software development also became more professional. Several tools were developed for prototyping, software testing and libraries of reusable programs were marketed. Both the PC and the workstation platform allowed for graphical user interfaces, which was an essential element of decision support systems. Finally, software platforms became standardized, e.g. the Windows environment for the PC, which allowed software vendors to regain their investments over many users.

Other information technological means yielded a wealth of accurate data. Bar code readers were helpful in warehouses and retail shops. Electronic payments could well be registered and serve as input data for optimization. Communication technology also became computerized, allowing precise information of telephone traffic. Moreover, these computerized environments often required automatic decision making procedures, e.g. to route telephone calls or to schedule and route automatic guided vehicles. Each one required quite sophisticated operations research methods.

4 Applications areas

A traditional application area of Operations Research is production and logistics. Within this area one may distinguish between production planning and scheduling, inventory management, distribution optimization and vehicle routing. A related but more general problem area is timetabling and (crew) rostering. Apart from the manufacturing industry, Operations Research is increasingly important for the service industry, viz. telecommunication, transportation and finance. Closely related is the medical sector in which OR has also played a role. We conclude with a discussion on maintenance optimization and reliability.

4.1 Production planning and scheduling

Production planning and scheduling has been a popular area in operations research. The impact on practice, however, is limited, because the necessary information-technological infrastructure and problem structuring is not yet estabilished in many companies. Shell has been among the first companies to apply linear programming in its refinery planning and scheduling, starting already in the sixties. Since that time quite some research has been done. In practice the refinery scheduling was integrated in the overall refinery information system, to ensure more up to date information. Several advances were made on nonlinear problems, like multi-period blending.

In many other companies planning and scheduling was less structured and the extent to which OR was used varied from company to company. The popular Manufacturing Resource Planning (MRP) systems which were introduced in many companies did not use any OR models to support planning or scheduling, despite all the research carried out in this area. The couple of applications done by e.g. Beyers and Partners, Ortec Consultants and AKB were often developed for a specific company and separate from the MRP packages. Noteworthy is the success of Beyers and Partners in a specific subarea, viz. cutting problems for the cardboard, plastic, steel and paper industry. Research on production concepts in relation with industry was done at the Technical Universities of Eindhoven (Bertrand, Wijngaard and van Donselaar) and Twente (Zijm). From the latter a shopfloor scheduling decision support system emerged, from which much is to be expected.

The ultimate goal of the research in this area is to incorporate the OR methods in the standard planning and scheduling packages around and in particular in the many MRP systems. The move by Baan Info Systems to start an OR group gives rise to high expectations in this respect.

4.2 Distribution: location and network optimization

A main strategic problem with distribution optimization is the design and analysis of a distribution network. The purpose of such a network is to bring the finished goods from production plants to customers. The main questions to be answered are: (i) how many layers to use, e.g. european, national or regional distribution centres, (ii) where to locate these distribution centres and (iii) which customers to supply from which distribution centre.

These problems are tackled by the location/allocation models within Operations Research. Apart from some small applications, we like to mention the Strategic Location Allocation Model (SLAM) which was used by the beer company Heineken in her international distribution network and which was developed together with the Erasmus Universities management science department (Van Nunen). This model was successfully applied in several countries.

4.3 Distribution: vehicle routing

Vehicle routing has been a successful area for operations research. Already in the early eigthies programs were developed with which trucks could be routed more efficiently. Several companies in the Netherlands (among which Ortec and AKB) developed such programmes. Also at the universities and at the Centre for Mathematics and Computer Science (CWI) research was done in this area (Lenstra, Savelsbergh). Soon the area became more established, extending the original routing problem (which was already computationally hard to solve) with time windows and other constraints. The success of the packages also depended on a good user interface and the upcoming of more graphical oriented computer systems (PC's and workstations) was welcomed. There was some variety between companies: routing gasoline trucks appeared to be much more difficult (because of the need for special loading algorithms) than trucks delivering packages. Also the average number of drops per trip appeared to be a distinguishing variable. Nowadays there is a trend to extend the routing systems with a dynamic allocation of the trips (real-time scheduling) and to take more detailed information on traffic (like traffic congestion) into account. A drawback to be mentioned was that each implementation required quite dedicated software and that the savings per implementation were interesting, but not overwhelming.

4.4 Inventory management

The famous Economic Order Quantity (EOQ) formula for the optimum order size is one of the first known applications of Operations Research. It was already developed in the first decade of this century. Its application in practice only came off the ground in the seventies when computerized inventory management systems were introduced. Before that time people had problems with the square root and a story even goes that one administration office, which had to implement the EOQ formula, found it that difficult that they changed the square root by a division through 2, since that gave the same outcomes in case the number to be dealt with is 4. Today, the EOQ formula is presumably the most applied and misapplied formula of OR and it can be found in almost every inventory management system. Other simple results of OR, like the expression for the safety stock in case of normally distributed leadtime demands, are also widely applied. The more sophisticated optimization behind many inventory models, however, has hardly found application. This may be due to the complexity of the optimization, to the fact that it is not that clear when which optimization model should be used or finally to a lack of theory on how all kind of practical issues should be incorporated. Cases done by the Technical University of Eindhoven (De Kok), however, show that substantial savings may be obtained in this area. A recent decision support system developed at the CERN's warehouse is promising. Finally, we would like to mention that several OR techniques are used in warehouse management systems for e.g. order picking.

4.5 Timetabling, crew rostering and scheduling

Another area in which OR managed to penetrate was timetabling and personnel rostering. The best known example is the timetable of the Dutch Railways. In the early eighties several attempts had been made to generate this timetable by a computer, but only in the nineties initial successes were booked. The first success in the OR/computer science area was a program, called Reisplanner, with which for couple of stations the fastest route could be determined. This program was successfully marketed to the general public. Pioneering in in the Netherlands was the CWI (Schrijver) who tackled the problem of determining which trains should travel when. Later on, the Erasmus University (Zwaneveld, Kroon), together with Railned tackled the problem of the train routing at stations (which route to tack to enter/leave a station, which platform to take) in a deterministic way. Stochastic delays were tackled by University of Amsterdam (Van Dijk). This work is not yet finished, it is however, that promising that decision support systems are being built. Next to this example major work was done on crew rostering and manpower scheduling, especially with the airlines. A last interesting example is the roster of the soccer competition which was made using OR methods (Schreuder and Telgen).

4.6 Telecommunication and queueing

With the privitization of the Dutch telecom market several new developments were initiated. As a result their was much interest in telecom network design, both in a deterministic as well as in stochastic way. Queueing research peaked in the eighties, but most research was more methodologically rather than application oriented. Nevertheless, applications were done in many projects, ranging from manpower planning in call centres to queueing disciplines in postal offices (Van Dijk). The techniques needed and the projects appeared to be much more mature compared to some time ago.

4.7 Finance

Also in the finance area operations research had its impact. Its role however, was often hidden. For example, one can use quadratic programming to determine the optimal investment portfolio. A major development in this decade was the so-called asset-liability management, in which assets of pension funds and insurance companies were matched with their liabilities as to determine a good investment portfolio. In the mid nineties most pension funds were applying such studies which made use of econometric analysis of the economy, simulation, scenarios and heuristic optimization. In another branch of finance like option pricing, much applied probability theory was used. Major developments came from within those within the finance community with a strong mathematical/econometric background. The OR community only lately realized the importance of this area in which their are also major challanges for optimization. Finally, a pure OR project worth mentioning in banks is cash management. Stimulated by a wealth of data from automated teller machines, projects were carried out to determine optimal cash inventories and to reduce money transportation costs and risks.

4.8 Medical decision making

This appeared to be a fruitful area for application of OR. Health budgets were under high pressure because of many reasons. In several areas decisions had to be made on the use of new and expensive technology (technology assessment) and on introducing wide scale preventive programs (screening). This all called for a well structured and well-founded decision making. Here practical statistics went hand-in-hand with quantitative models to support decision making. Several groups flourished at the universities, especially at the Erasmus University (Habbema). Areas investigated concerned screening of cervical cancer, breath cancer, prostate cancer as well as clinical decision making (on e.g. replacement of heart valves) and infectuous desease control. Outcomes of the research directly influenced public decision making. The AIDS epidemic proved to be a very interesting area for quantitative researchers.

4.9 Maintenance and reliability

Even in maintenance and reliability there was a significant contribution of OR. Shell (Schornagel, Van der Heijden and Groenendijk) did for many years research on methods to predict system availability from reliability data on individual units, to be used in their evaluation of potential designs of production systems. Several new stochastic methods were invented and some PhD's were obtained in this area. OR was also used to optimize maintenance during operations and several decision support systems found their way in this technique-oriented discipline, both for mechanical equipment as well as for civil structures like highways (Dekker, Vanneste, Van Rijn and Van Harten).

5 General discussion

OR was founded in World War II with a problem oriented perspective. Its theoretical foundations were laid in the fifties. The most important model, the linear programming model and the simplex algorithm were invented in 1947-49 and further developed in the 50ies. First OR courses at universities were given in the sixties. At the same time large companies started OR groups. The whole field entailed big promises. Some people envisaged that all

decision making in a factory would one day be done using a big company model.

A disappointment came in the seventies, although at that time OR reached a peak at the universities, where at each university one or two professors in OR were appointed. OR could not solve all problems. It took much time to do the studies, data was often lacking, reality was more complex than the models could handle and it was difficult to convince the management of the outcomes of the studies. Finally, there were little tools to get the results from OR implemented in practice. Articles appeared with titles like: the future of OR is passed (Ackoff [5]). At the same time alternative approaches came forward, like expert or knowledge based systems. From hindsight one may conclude that society was not yet ripe for a widescale application of OR, nor was its theory enough developed to tackle the often complex practical problems. The whole technological infrastructure which we have now, had to be created.

In the mid eighties there came again an upward trend. The technological push assisted OR in many ways as is argued in section 3. The high hopes from the sixties, however, remained lost, but OR managed to have an impact on society and optimism was regained (Rinnooy Kan [6]).

Although real life problems are often large and complex with a lot of uncertainty, we should strive to formulate general problem definitions like the LP problem, the G/G/1 queueing problem and the location problem. These can then be studied in isolation and after some time major results can be obtained.

We should take time in this respect. Once we make some progress both in algorithmic aspects and in problem specific modelling and analysis we can also expect for many more applications, as the problems driven by technology and automation will ask for them.

References

[1] Fortuin, L. Van Wassenhove L.G. and Van Beek, P., OR at Work, Taylor and Francis, London, 1996.

[2] Tilanus, C.B., O.B. de Gans en J.K. Lenstra, Quantitative methods in management: Aula paperback 69, Het Spectrum, Utrecht, 1983.

[3] Hacquebard, A.E.N., Lombaers, H.J.M., Tilanus, C.B. and Chamalaun, M. (eds.), (1981), The practice of operations research, Intermediair publishers, Amsterdam (in Dutch).

[4] Lombaers, H.J.M., Meinardi, J.J. and Ravestijn, D. (eds)., (1968), Operational research in the Netherlands, Marka pocket 104, Het Spectrum, Utrecht, The Netherlands (in Dutch).

[5] Ackoff, R.L. The future of Operational Research is past. *Journ. Oper. Res. Soc.* **30** (2), 93-104, 1979.

[6] Rinnooy Kan, A.H.G. The future of Operations Research is bright. *Eur. Journ. Oper. Res.* **38**, 282-285, 1989.

Integration of Operational Research and Environmental Management*

Jacqueline M. Bloemhof-Ruwaard
Faculty of Business Administration,
Erasmus University Rotterdam,
E-mail: J.Bloemhof@fac.fbk.eur.nl

1 Introduction

Environmental issues have come to play an important role in the decision making processes of agriculture, trade and industry. The obligation to reduce waste and to control emissions affects decisions on e.g. production planning, logistics, and inventory control. On the other hand, the international nature and the complexity of many environmental problems make it virtually impossible to base policy decisions on intuition and simple methods.

These developments have stimulated us to investigate the possibilities of incorporating environmental issues in 'traditional' Operational Research (OR) applications, and the possibilities of using OR models and techniques in Environmental Management (EM) problems. Two research questions structure this exploration of the integration of Operational Research and Environmental Management:

1. How can Operational Research models and methods be adapted to include environmental issues?
2. To which degree can Operational Research models and methods be used to solve environmental problems?

Section 2 describes the framework we developed to show how environmental issues and economic decision making interact in two ways. Section 3 gives two examples of including environmental issues in OR applications, whereas Section 4 gives two examples of using OR models to solve environmental problems. Section 5 concludes with a summary of the main results.

2 Framework

We describe the development of a framework to show how environmental issues and economic processes can interact in two ways:

*This is a summary of a PhD thesis with the same title

1. *Impact of environmental issues on the supply chain:* Environmental issues play a role in the routine activities of firms. Decisions on e.g. production planning, logistics, and inventory control will change due to legal requirements or consumer pressures to reduce waste and emissions. Therefore, there is a need to adapt OR tools such as production planning algorithms, location models and routing heuristics in order to deal adequately with a new situation requiring 'green supply chain modelling'.

2. *Impact of economic activities on the environmental chain:* The amount of waste and the level of emissions caused by the supply chain can result in a number of serious environmental pollution problems, such as global warming and acid rain. Frequently, these environmental problems are international and complex. The interaction between OR and EM can result in a clear formulation of the headlines of these problems and in new insights in the impacts of alternative policy measures.

Figure 1 shows the degrees of the integration of OR and EM with respect to supply chain modelling and environmental chain modelling.

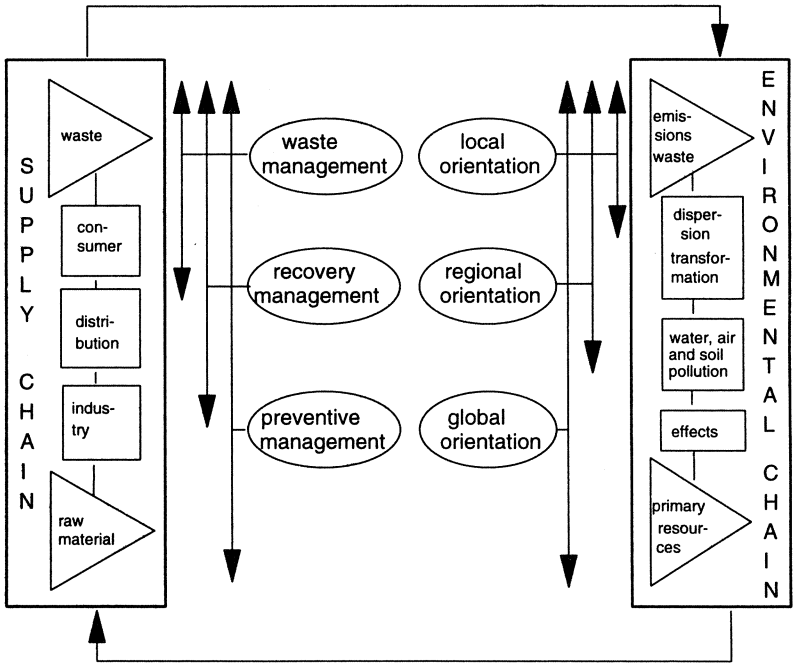


Figure 1: Framework

The *supply chain* comprises the extraction of raw materials, production, distribution, use of goods and waste collection. In general, changes within the supply chain are necessary to reduce the amount of waste and emissions, and the use of non-renewable resources. We structure the discussion around a hypothetical shift from an end-of-pipe approach (waste management), via recovery management (e.g. recycling) towards a source-oriented approach

of pollution prevention:

1. *waste management*
   In the first degree of integration, environmental issues (waste and emissions) influence only the final processes in the supply chain, i.e. distribution, product use and waste disposal. Economic activities that have to be developed or adapted are e.g. the distribution of hazardous waste and the selection of appropriate locations for disposal sites and incinerators. Emissions to air and water are in most cases reduced through end-of-pipe techniques.

2. *recovery management*
   On the intermediate level of integration, production is also subject to change because of environmental issues. Recovery management aims at postponing the generation of waste and lengthening the life time of products by means of recycling and reuse. This requires manufacturers to feel responsible for their products after consumer use and to consider ways to increase the use of recycled materials. It requires changes in both process and product design.

3. *preventive management*
   Prevention is the key-issue in the final stage of integration. The objective of pollution prevention is to avoid the use of materials whose acquisition and transformation are environmentally damaging. Environmental burdens associated with a product or process have to be quantified. Evaluation studies of this assessment are in general complex and expensive.

In the *environmental chain*, emissions and waste are transported and transformed and result in water, air, and soil pollution with damaging effects to the environment. We structure the discussion around a hypothetical shift from single pollutant/single effect abatement policies, via a regional orientation towards a global, integral approach that can cope with a.o. multipollutant transboundary problems:

1. *local orientation*
   At this level, local problems (e.g. soil pollution, smog) can be solved by local solutions (end-of-pipe technologies). Measures to abate emissions reduce environmental effects at the end of pipe without influencing economic activities (flue gas desulfurization, low NO_x burners).

2. *regional orientation*
   Emissions and waste disposal cause effects on ecosystems and human health. These environmental effects depend largely on the geographic distribution and movement of damaging components. Therefore, environmental policy evolves towards an approach that can cope with the effects of emissions on all components of the environment (soil, water and air).

3. *global orientation*
   General indicators of the effects on ecosystems and human beings are developed, rather than focusing on effects of a single pollutant on a single species. An example is the concept of critical loads in acidification policy, i.e. the highest deposition of an acidifying compound that will not lead to long-term harmful effects. The intention of policy makers at this level is to use an integrated approach for global problems.

3 Integration of Environmental Management in Operational Research

Incorporating environmental issues in OR models can vary from just changing one parameter to complete new problem formulations. Two examples are given to show why incorporating environmental issues (and knowledge from Environmental Management) enriches the working area of Operational Research. The first example deals with the distribution and disposal of waste as an illustrative example of *waste management.* The second example deals with the environmental impacts of raw materials in product blending, as an illustrative example of *preventive management.*

Example 1 describes an extension of the classical location problem (Brandeau and Chiu, 1989). The location problem deals with the design of an 'optimal' distribution structure of plants, based on a trade-off between fixed costs of plants and variable costs of the transportation of products between plants and clients. Environmental issues play a role if waste, generated during production, has to be disposed of at waste disposal units. This situation occurs for example in agriculture (transportation of manure), and in nuclear power stations (transportation of nuclear waste). The question is then how to design an efficient distribution structure which simultaneously takes into account the location of plants and waste disposal units (WDUs) (see Figure 2).

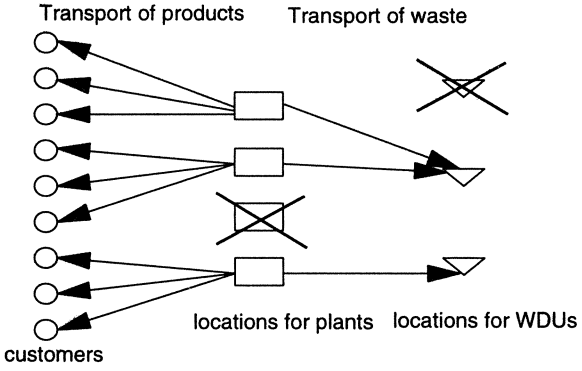


Figure 2: Product and waste flows

This problem has a very close relation to the traditional location models. Therefore, classical solution approaches such as linear programming round-off heuristics, lagrangean relaxations and greedy-like heuristics perform rather well. However, results indicate that problem-specific solution approaches require less computation time or give better bounds.

Example 2 describes an extension of the classical blending problem (Winston, 1991). The blending problem concerns the composition of a raw materials blend such that the costs of the raw materials are minimal and the blend satisfies all quality conditions of the product. Apart from having a good-quality product, it is also important to sell an environmentally friendly product. The environmental quality of e.g. margarine depends on the environmental quality of the raw materials, i.e. the growing of the crops, the production of crude oil, refining

and processing of the oils and transportation (see Figure 3). Life cycle analysis is used as a method to identify the environmental burden of these processes (SETAC, 1993). To weigh the environmental quality of the raw materials against the costs of raw materials, multiple criteria analysis is used (Zahedi, 1986). The environmental index is a one-dimensional parameter, that can be used to find the optimal blend of an environmentally friendly product using single-objective or multi-objective linear programming.

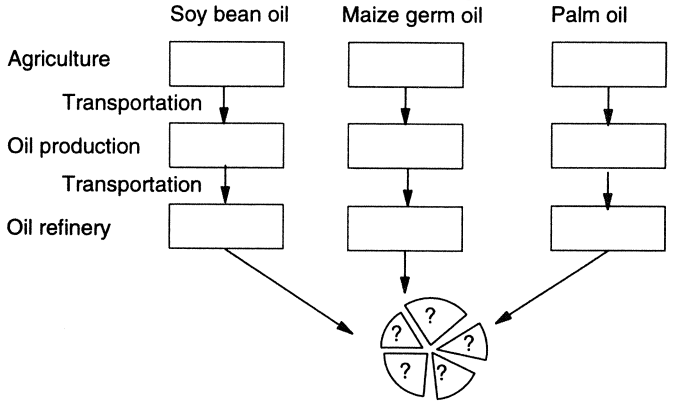


Figure 3: Relevant processes for the environmental quality of margarine

The outcomes of the study show that when sufficient reliable data are available, the method is capable of ranking different alternatives on their environmental performance. The obtained environmental index is rather robust for the opinions of decision makers. Summarizing, this methodology study is a first step towards using environmental information in product development.

4 Integration of Operational Research in Environmental Management

We present two examples that show why the knowledge of OR will be useful in environmental problem solving. Example 3 deals with the Dutch manure problem which is a local problem. Example 4 deals with environmental policy making for the recycling of European paper as an example of a regional approach to environmental problems.

Example 3 describes the mineral excess problem in the Netherlands. After World War II the agricultural sector in the Netherlands experienced an explosive growth. Especially intensive livestock has increased enormously (e.g. the number of pigs increased from almost 2 million in 1950 to 15 million in 1993). This growth, together with specialisation and excess minerals in fodder, led to excess minerals in soil, water and air, leading to eutrophication and acidification. A *local approach* towards this problem consists of fodder adjustments, emission-poor stables, covered storage, and application measures. Beforehand, it is hard to tell which combination of measures is the most effective and efficient, due to the complexity

of the problem (see Figure 4).

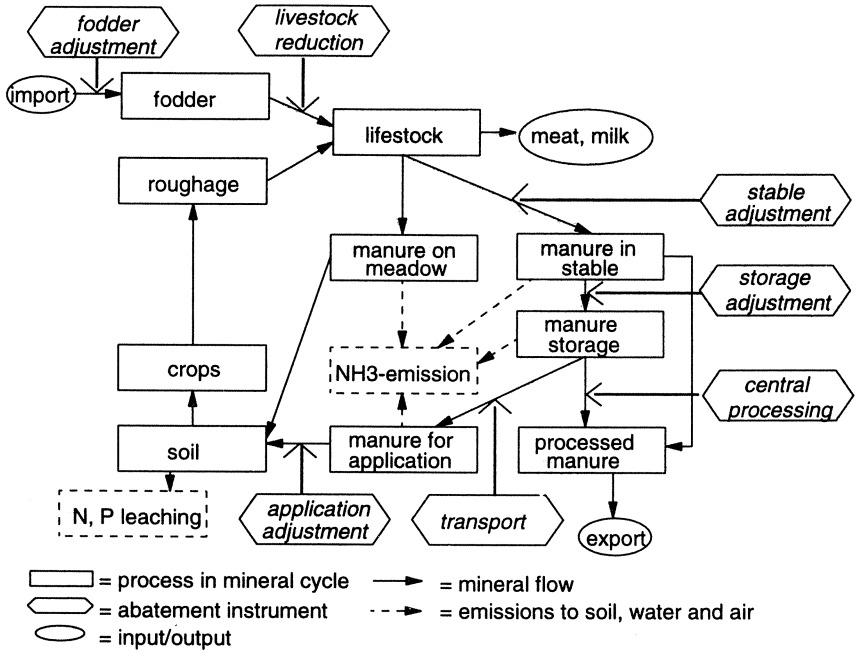


Figure 4: Abatement measures in the mineral cycle

Operational Research can play a role both in structuring the problem and determine strategies for various environmental standards. In this example, linear programming is used to find suitable measures for the different parts of the Netherlands.

In *Example 4* we discuss the issue of paper recycling in Europe. Recycling of waste paper will reduce the environmental impacts of the pulp and paper industry. Is maximal paper recycling the best policy (from an environmental point of view and from an economic point of view) or should other possibilities such as waste paper incineration for energy recovery be considered? (see Figure 5). To explore this question, we use a combination of life cycle analysis and optimization. A linear network flow model helps to find answers to questions such as: what is the optimal collection rate of waste paper, and what is the optimal share of recycled pulp in paper furnish such that total environmental impact of the paper and pulp industry in Europe is minimized. This is an example of the use of OR models in a *regional orientation.*

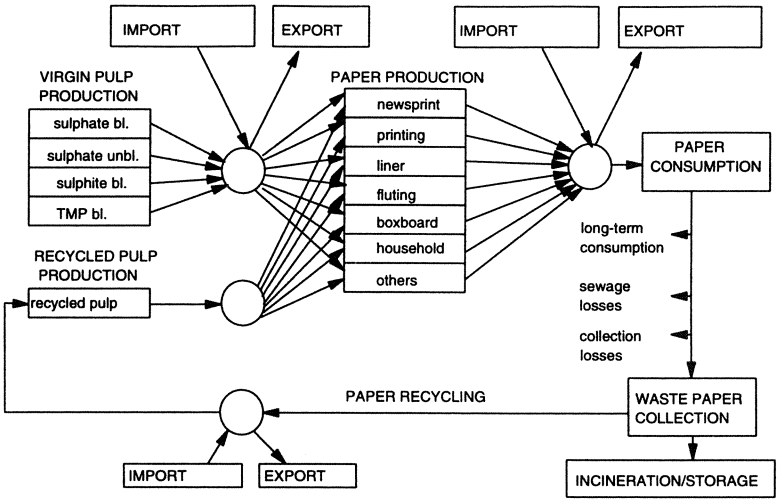


Figure 5: Network flows in the paper recycling problem

This example illustrates that dealing with regional problems requires an integrated modelling approach using optimization models, economic models, and physical models. Operational Research can be effective for the policy debate on the reduction of environmental pollution. A multiobjective approach accomodates dealing with different opinions.

5 Conclusions

The growing number of examples of dealing with environmental issues within the field of Operational Research can be classified using a framework consisting of two approaches: the supply chain approach and the environmental chain approach. In both approaches there is a shift from local, corrective, end-of-pipe policies to global, preventive policies.

In general, we conclude that the first phase in the shift from corrective policy towards preventive policy generates a rather straightforward use of all kinds of Operational Research applications. The intermediate phase in this shift describes more complicated and adapted models (and methods). The final stage in the shift will probably be reached in the near future. Here, OR has to integrate with related sciences to be able to use tools like life cycle analysis, economic input-output modelling and system analysis.

References

[1] Bloemhof-Ruwaard, J.M. (1996), *Integration of Operational Research and Environmental Management*, Doctoral Thesis, Agricultural University of Wageningen.

[2] Brandeau, M.L., and S.S. Chiu (1989), "An overview of representative problems in location research", *Management Science* **35**, 645-674.

[3] SETAC (1993), *Guidelines for life-cycle assessment: a 'Code of Practice'*, Brussels.

[4] Winston, W.L. (1991), *Operations Research; Applications and Algorithms*, 2nd edition, PWS-Kent, USA.

[5] Zahedi, F. (1986), "The Analytic Hierarchy Process – A survey of the method and its applications", *Interfaces* **16**, 96-108.

Robust investments

Douwe-Frits Broens
Department of Marketing and Marketing Research
Wageningen Agricultural University
E-mail: Douwe-Frits.Broens@alg.menm.wau.nl

Abstract

This paper is a summary of my PhD–thesis *Investment evaluation with respect to commercial uncertainty*, prepared at the University of Groningen, Department of Econometrics, and defended on January 26, 1996 (Broens, 1995).

Investment planning

In the *investment planning* of a company, the following actions are comprehended:

a. the search for the desirability of extensions in the technical equipment of the company, like capacity expansion;

b. in case changes are desired, the generation of useful alternatives as to the future layout of the technical equipment such that commercial objectives will be better served;

c. the evaluation of the alternatives and the selection of the best option.

Changes in the technical equipment are supposed to be indicated if it is expected that in future the production planning will not be able to fulfill all requirements following from the then existing commercial contracts.

Robustness and flexibility

While preparing for an uncertain future, a planner can choose two strategic directions: robustness and flexibility, conform the two survival strategies under uncertainty presented by the classical fable about the oak and the reed (La Fontaine, 1985). In analogy, we distinguish two investment planning archetypes. The *flexible planning strategy* implies late and cautious commitments, making no firm plans until all necessary information is available and changing its direction as often as it seems right. In a *robust planning strategy*, the planner commits him– or herself completely to an investment plan, making sure that the plan covers enough future scenarios to be acceptable, so that the plan can be executed as was decided from the start. The difference between the two planning archetypes resembles the distinction, made in stochastic programming theory, between *here–and–now* and *wait–and–see* decision problems. In the case of a robust planning strategy, there is no difference between the primal commitments and the final plan describing the desired end–state. In

addition, the *robustness of an investment plan* is defined as the degree to which it covers the future commercial scenarios which are considered as realistic.

In our view, flexibility is primarily a property of the planning process, whereas a robust planning is robust merely because its plans are robust. Of course, real planning processes make the best of both worlds. A purely robust planning might lead to unacceptable over-capacity. A flexible planning promises to deliver tailor–made solutions but it may lead to an unacceptable degree of 'planning nervousness' as a result of constantly changing insights. And what is more, long investment lead times and political involvements often force the planner to early commitments. Though in some cases flexibility of the planning method might lessen the need for robustness of the plan, in most situations a certain robustness of the plan is well–grounded. In a sense, robustness and flexibility are complementary: one should have the planning as flexible as possible, but, once commitments are made, one should make them as robust as necessary.

The abovementioned definitions of flexibility and robustness are not indisputable. The 'robustness indicator' developed by Rosenhead (1989), according to the above definitions, expresses an aspect of planning flexibility. On the other hand, for instance the 'flexibility index' developed by Swaney and Grossmann (1985), describes the robustness of a given design.

Worst case approaches go a long way in the right direction, except for the 'minimax regret' approach, in which to our opinion the 'ex post optimal profit' plays too heavy a role. All worst case or minimax approaches pursue 'satisfactory results' under *all* possible futures. In our opinion this is a rather limited tool in long term planning. First of all, too much attention is given to the so called 'worst case' scenario, and all other information about future uncertainties is spoilt. Now it may be true that the planner has at his disposal a kernel of scenarios which he has to tackle at all cost. For these scenarios a worst case approach is appropriate. But mostly there is a large grey field of futures which are not very probable. That is, a planner would like to tackle these scenarios as well, but not at all cost. Cost and robustness criteria have to be balanced.

Investment objectives

Robustness can be defined as the insensitivity of conclusions to deviations from assumptions. Commercial robustness of investment plans can be defined as the ability to meet many different (relevant) commercial futures. There are no reported examples of the explicit use of commercial robustness measures in investment planning procedures. Based on some polls among large English companies in the eighties, refering to their investment planning, Pike (1982) reports the use of financial variables like:

- net present value,
- internal rate of return,
- payback period

or 'risk appraisal techniques' like:

- shortening the payback period,
- raising the required rate of return,
- sensitivity analysis,

- 'probability analysis'

Although in the first part of the eighties, trends went towards more 'sophisticated' techniques like stochastic risk calculations, later trends showed a tendency towards scenario–wise analysis. Many respondents found risk techniques too much demanding in the sense of information and time, and too little rewarding in the sense of reliable results.

In management science theory, some examples can be found of robustness measures applied to planning problems. In fact, in stochastic programming each decision model aims at robustness. In these models it is avoided to give uncertain parameters one specific value, so that the optimal solutions do not depend on the choice of the specific parameter value. Some authors identify the traditional two–stage recourse approach in general with the name robust optimization. In a stricter sense, robust stochastic programming models are characterized by the property that only partial information on the probability distribution or the risk assessment is needed. Sengupta (1991) names mean–variance analysis, well–known from portfolio analysis, as a way to generate robust decisions. A nice example of this interpretation can be found in Paraskevopoulos (1991). In the same sense, Sengupta argues that the use of stochastic dominance leads to robustness: the best choice according to this criterion is optimal for a large class of utility functions. In a more direct sense, the minimax approach of stochastic programming, see Zackova (1966) or Klein Haneveld (1986), can be called robust, since it bases its definition of optimality upon the worst case in the large class of all distributions compatible with the partial information on the distribution. A worst–case approach for decision problems under uncertainty based on a predefined set of parameter values rather than distribution functions, leads to well–known problems like maximin utility, minimax loss or minimax regret optimizations, see Sengupta (1991).

Kouvelis *et al.* (1992) study robust optimization of production facility layout planning. They use a variation of the minimax (relative) regret criterion. They relate the robustness of a facility layout decision to the resulting material handling cost, which is a direct function of the future facility layout and the uncertain future commercial circumstances. The regret under a certain plan and commercial future, is defined as the material handling cost minus the minimal material handling cost that could have been realized, would the commercial scenario have been known beforehand. Kouvelis *et al.* call a layout alternative robust if, under all possible scenarios, the regret is less than a prespecified percentage of the 'complete information' optimum.

Rosenhead (1989) directly addresses the investment problem under uncertainty. He particularly distinguishes between initial commitments and the final technical equipment configuration to arrive at. The final configuration, called the 'investment plan', depends both on the initial commitments and on later commitments. Rosenhead defines the robustness of any initial commitment to be the number of 'acceptable' investment plans with which it is compatible, expressed as the ratio of the total number of acceptable plans. Occasionally this robustness is split out for several future scenarios. Rosenhead, contrary to our clear distinction between robustness and flexibility, interpretes robustness as 'a particular perspective on flexibility'.

Many of robust optimisation approaches, like the mean–variance optimization, try to diminish the sensitivity of the optimum of a certain primary goal, like future profit, for variations in the uncertain parameters. In case of high uncertainty however there is no need for a robust optimum, but for optimal robustness. That is, we are interested in the use of robustness measures as investment criteria. Worst case approaches go a long way in the right direction, but they pursue satisfactory results under *all* possible futures. This is a rather limited tool in long term planning. A heavy weight is put on the worst case scenario, all

other information about future uncertainties is spoilt. It may be that a planner has at his disposal a kernel of scenarios to be tackled all cost, justifying a worst case approach. But mostly there is a grey area of future scenarios which the planner likes to meet, but not at all cost. Costs and robustness criteria have to be balanced.

Robustness and risk

Closely related to the concept of robustness is the concept of **risk**. It measures the expected loss due to undesirable outcomes. In terms of the investment planning problem, risk is the expected value of the loss due to the occurence of an infeasible commercial scenario. In general, the larger the robustness, the smaller the risk. Contrary to robustness, risk addresses the possibility of an undesirable future. It depends on the investment plan, the uncertain commercial future, and also on the loss connected to undesirable outcomes. Since risk depends also on the loss function, it is essentially different from robustness. The two types of measures are complementary: both serve to describe the value of a given investment plan, represented by its commercial scope, with respect to the uncertain commercial future.

The commercial scope

Let the future short term planning problem, which is to be facilitated by the investments under study, be given by linear restrictions on the production decision variables y, which are a function of the uncertain commercial future, described by the vector s, and of the investment plan, described by the vector x:

$$\text{Find a } y \in \mathbb{R}^n \text{ such that } Ky \leq Ls + Mx \tag{1}$$

where: $y \in \mathbb{R}^n$ is the vector of production decision variables,
$s \in \mathbb{R}^{n_s}$ is a vector representing the commercial scenario,
$x \in \mathbb{R}^{n_x}$ is a vector of investment decision variables
The matrices K, L and M have row dimension m, with $n < m$, $n_s \ll m$ and $n_x \ll m$. It is assumed that the vector s of commercial variables is known when the production plan y has to be made. On the other hand, in the investment problem, where the decision on the investment vector x is at stake, the value of s is uncertain.

An investment plan x is robust if it enables a feasible short term planning under 'enough' 'realistic' commercial future scenarios. To formalise this, we define the commercial scope $S(x)$ as the set of scenarios s which, together with investment alternative x, yield a feasible production planning problem.

For a given vector x of investment decisions the set of scenarios defined by

$$S(x) := \{s \in \mathbb{R}^{n_s} \mid \exists\, y \in \mathbb{R}^n \;\; Ky \leq Ls + Mx\} \tag{2}$$

is called the **commercial scope** of the investment x.

Using Farkas's lemma it can be shown (Broens, 1995; Broens and Klein Haneveld, 1996) that for linear production planning restrictions, the set $S(x)$ is a polyhedron described by a finite number of *induced* linear constraints on s:

$$S(x) = \{s \in \mathbb{R}^{n_s} \mid \Pi(Ls + Mx) \geq 0\}$$

for some finite matrix Π with the property $\Pi K = 0$. More general, any closed convex set can be described by a number of such linear constraints, although in general this number is not finite.

Given x and s, the production planning problem is feasible if $s \in S(x)$. Commercial scenarios are called feasible under x if they are in $S(x)$. The commercial scope represents the technical capabilities of the physical system, in terms of commercial variables. Any robustness measure is the result of a confrontation of the uncertainty of the commercial variables s with the commercial scope of the investment decision. Since the scope is multidimensional, in general different robustness measures can be applied beside each other.

Measuring feasibility of a scenario

It is rather unusual to consider feasibility as a property of model parameters rather than of a decision. It may seem unwise to measure feasibility of a problem by varying its parameters rather than its decision variables, but in long term planning this may be the only thing to do: if the future turns out to be infeasible, building new capacity on the spot is impossible, the only alternative is to allow for stockouts.

It is easily tested whether a given scenario $\overline{s}$ is feasible or not under x. To measure its *degree of feasibility*, one could study how far it may change into a specific fixed direction d without losing feasibility. The distance of scenario $\overline{s}$ to the boundary of the scope $S(x)$ in the direction d is referred to as the *directional scope* of x at $\overline{s}$ in the direction of d, denoted as $S_d(x; \overline{s}, d)$:

$$\begin{array}{l} \sup_{\alpha \in \mathbb{R}} \{\alpha \mid \overline{s} + \alpha d \in S(x)\} = \\ \sup_{y \in \mathbb{R}^n,\ \alpha \in \mathbb{R}} \{\alpha \mid Ky - \alpha Ld \leq L\overline{s} + Mx\} \\ \inf_{\pi \in \mathbb{R}^m} \{\pi(L\overline{s} + Mx) \mid \pi K = 0,\ \pi Ld = -1,\ \pi \geq 0\} \end{array} \tag{3}$$

Apart from degenerate cases, the boundary point $s = \overline{s} + \alpha^o d$ lies on exactly one facet of $S(x)$. It is easily proved that the optimal value α^o of (3) defines the boundary point $\overline{s} + \alpha d$ of the scope, whereas the optimal solution π^o of the dual problem gives the nonredundant induced constraint $-\pi^o Ls \leq \pi^o Mx$ defines a facet of the scope.

Such directional searches are applied in practice, when an investment planner, given a plan x and some scenario $\overline{s}$, may ask, how many contracts of a certain type could be attracted in addition to $\overline{s}$. In linear programming theory these line searches are a proved technique to identify nonredundancy among a number of linear inequalities (Kuhn, 1956; Telgen *et al*, 1983; Boender *et al*, 1991).

Global measures

To judge an investment plan on its robustness, one could formulate a number of interesting scenarios, and demand that they are all feasible. One step beyond is made in an interesting study by Swaney and Grossmann (1985). They define two vectors of 'expected positive' and 'negative deviations', Δsk and Δsm respectively. They furthermore define a block formed polyhedron $T(\alpha)$ as

$$T(\alpha) := \{s \mid \overline{s} - \alpha \cdot \Delta sm \leq s \leq s^o - \alpha \cdot \Delta sk\}$$

Now they define a robustness index (they say 'flexibility index') $F(x)$ as the largest value of α such that this block is completely inside $S(x)$, in other words as the optimal value of:

$$\sup_{\alpha \geq 0} \{\alpha \mid T(\alpha) \subseteq S(x)\}$$

Now call the vertices of the blockformed polyhedron $T(1)$

$$s^j = \overline{s} + d^j, \qquad j = 1, \ldots, J = 2^{n_s}$$

Since $S(x)$ is convex, $F(x)$ can be found as the optimal value to:

$$\min_{j=1,\ldots,J} \sup_{\alpha^j \geq 0} \{\alpha^j \mid \overline{s} + \alpha^j d^j \in S(x)\} = \min_{j=1,\ldots,J} S_d(x; \overline{s}, d^j)$$

that is, as the minimum of a set of directional scopes. The index j for which the minimum is attained is a relatively risky direction. Swaney and Grossmann (1985), and Pistikopoulos and Grossmann (1989), use this index as an overall measure of investment robustness, to be balanced with investment cost.

A more natural way to quantify uncertainty with respect to the commercial scenario s of a future year is to assume that it is an unknown realization of a random vector $s(\omega)$, of which one is willing to specify a probability distribution. By computing expected values a global assessment of the adequacy of the scope is achieved, especially regarding the commercial uncertainty which is explicitly incorporated.

As a global measure of the robustness of the investment alternative x serves the *reliability* of x, defined as the probability that the realization of $s(\omega)$ will be feasible:

$$\begin{aligned} &\Pr\{s(\omega) \in S(x)\} = E(f_1(s(\omega), x)) \\ &\text{with } f_1(s,x) = \begin{cases} 1 & \text{if } s \in S(x) \\ 0 & \text{otherwise} \end{cases} \end{aligned} \tag{4}$$

This robustness measure is also known as *customer reliability* or *(expected) service level.*

In practical situations, one may not have a complete list of the induced constraints on s determining $S(x)$, but only a subset of induced constraints. For any such subset $J \subseteq \{1, \ldots, I\}$, the probability that these constraints are met,

$$\Pr\{\pi^i L s(\omega) \geq -\pi^i M x, \ i \in J\}$$

is an upperbound on the reliability. This holds a forteriori for each constraint separately.

Additionally, more detailed information about the scope is given by *expected directional scopes.* The expected distance that separates a feasible $s(\omega)$ from infeasibility in a certain direction gives a clear view of the scope for future marketing policy and of the ability to absorb unexpected developments that are not incorporated in the specified probability distribution. The expected directional scope under x in the direction d is given by:

$$\begin{aligned} &E\{S_d(x; s(\omega), d) \mid s(\omega) \in S(x)\} = Ef_2(s(\omega), x; d)/E(f_1(s(\omega), x)) \\ &\text{with } f_2(s,x;d) := \begin{cases} S_d(x; s, d) & \text{if } s \in S(x) \\ 0 & \text{if } s \notin S(x) \end{cases} \end{aligned} \tag{5}$$

Conclusion

If uncertainty about essential data increases, the uncertainty more and more grows to be a decision criterion itself. Under high uncertainty, decision–makers do not aim at optimal profits, but instead will pursue a guarantee for 'doing well enough' in most or all possible cases. In that case, criteria like robustness, risk and flexibility get predominant. They should be weighed against each other, and the expected investment costs.

In particular the robustness criterion is only little applied. The scope approach forgets about profit for a while, but instead defines 'doing well enough' simply by feasibility of some restrictions. And these restrictions are subsequently projected on *the space of uncertain parameters*. In this space, the robustness discussion can be featured more directly than in the space of traditional planning criteria like profit or net present value, in terms of what really should be discussed: the assumptions concerning the commercial uncertainty.

Commercial data are more than only an expression of expectations with respect to a completely exogenous environment. In most situations, the company itself has some influence on this environment. Marketing plans tend to colour the commercial data, as they are handed over to the investment planners. The interaction between plans and uncertainty is often complex and politically loaded. If the investment planners communicate back their knowledge of the commercial scope, which they developed as a result of the analysis of the commercial data, this gives the marketeers the opportunity to judge the investment alternatives with an eye on their future plans. Furthermore, especially the directional scope gives information about 'weak' and 'strong spots' in the commercial environment. This can give rise to a more differentiated information gathering or to preventive commercial actions. Thus, the commercial scope is a means for the communication between the investment planning and marketing departments. It is especially suited to feedback essential information to the marketing departments.

Experiments (Broens and Klein Haneveld, 1996) showed that particularly the induced constraints are very useful to describe the facets of the scope. In this way a very direct and complete description of all robustness aspects could be achieved. If the number of constraints or production decision variables is large, the commercial scope can still be usefully analysed using induced constraints that follow from a heuristic boundary search: the complexity of the scope does not seem to increase rapidly in these dimensions. However, if the number of uncertain variables increases, it gets increasingly difficult to perform a broad analysis, extensively describing all aspects of the scope. More efforts are needed and more data are generated, which are increasingly less surveyable. For greater numbers of uncertain variables, one should to concentrate on either a smaller number of 'key uncertainties', or a smaller number of global scope measures like risk, reliability and perhaps the directional scope in one or two important directions.

References

[1] C.G.E. Boender, R.J. Caron, J.F. McDonald, A.H.G. Rinnooy Kan, H.E. Romeijn, R.L. Smith, J. Telgen, and A.C.F. Vorst. Shake–and–bake algorithms for generating uniform points on the boundary of bounded polyhedra. *Operations Research*, 39(6):945–954, 1991.

[2] D.F. Broens. *Investment evaluation with respect to commercial uncertainty: the production of natural gas.* Labyrint Publication, 1995.

[3] B. Rustem D. Paraskevopoulos, E. Karakitsos. Robust capacity planning under uncertainty. *Management Science*, 37(7):787–800, 1991.

[4] W.K. Klein Haneveld D.F. Broens. Investment evaluation based on the commercial scope: the production of natural gas. *Annals of Operations Research*, 59:195–226, 1996.

[5] I.E. Grossmann E.N. Pistikopoulos. Optimal retrofit design for improving process flexibility in nonlinear systems - II. Optimal level of flexibility. *Computers and Chemical Engineering*, 13(10):1087–1096, 1989.

[6] A. Schrijver J.K. Lenstra, A.H.G. Rinnooy Kan. *History of Mathematical Programming.* CWI-North Holland, 1991.

[7] M.H. Karwan, V. Lotfi, J. Telgen, and S. Zionts. *Redundancy in mathematical programming: a state-of-the-art survey*, volume 206 of *Lecture Notes in Economics and Mathematical Systems.* Springer–Verlag, Berlin, 1983.

[8] W.K. Klein Haneveld. *Duality in stochastic linear and dynamic programming*, volume 274 of *Lecture notes in economics and mathematical systems.* Springer–Verlag, 1986.

[9] G.J. Gutiérrez P. Kouvelis, A.A. Kurawarwala. Algorithms for robust single and multiple period layout planning for manufacturing systems. *European Journal of Operational Research*, 63:287–303, 1992.

[10] R.H. Pike. A major survey of the investment practices in large companies. *Occasional paper*, 1982.

[11] J. Rosenhead. *Rational analysis for a problematic world: problem structuring methods for complexity, uncertainty and conflict.* John Wiley & Sons, 1989.

[12] J.K. Sengupta. Robust solutions in stochastic linear programming. *J. Opl. Res. Soc.*, 42(10):857–870, 1991.

[13] R.E. Swaney and I.E. Grossmann. An index for operational flexibility in chemical process design. Part I: formulation and theory. *AIChE journal (American Institute of Chemical Engineers)*, 31:621, 1985.

[14] J. Zackova. On minimax solutions of stochastic linear programming problems. *Časopis Pěst. Math.*, 96:423–429, 1966.

Robotic flowshop scheduling is strongly NP-complete

Yves Crama Ecole d'Administration des Affaires
Université de Liège
Belgium
E-mail: Y.Crama@ulg.ac.be

Joris van de Klundert
Department of Quantitative Economics
Faculty of Economics and Business Administration
Maastricht University
E-mail: J.vandeKlundert@ke.unimaas.nl

Abstract

We consider a robotic flowshop model in which a single robot is responsible for the transportation of parts between machines and the amount of time that a part spends on a machine must be comprised in some predefined interval. The objective is to find a feasible schedule with minimal cycle time. Many researchers have proposed nonpolynomial solution methods for a variety of closely related robotic flowshop scheduling problems. This paper provides a proof that a basic version of this problem is strongly NP-Complete.

1 Introduction

One of the offsprings of the Just In Time production philosophy was the introduction of so called One-Worker Multiple-Machine lines, in which several machines are encircling a single operator. Typically, in the highly repetitive manufacturing environments of Just In Time implementors, all products, or parts, enter the line at an input or input/output station, and require processing on every machine in a prescribed order that is identical for all parts. Finally, the parts are delivered at the output (or input/output) station. The Just In Time emphasis on eliminating inventory demanded that in such a production cell, inventory may only be kept at the input/output station(s). As the automation of production advanced, the operator in the center of the cell, who performed materials handling activities, machine setups and quality inspection, has often been replaced by a single robot. The resulting production cell, is often referred as *robotic flowshop* (see Figure 1). Scheduling problems in such cells have become known as robotic flowshop scheduling problems, robotic cell scheduling problems, and crane or hoist scheduling problems. Such a robotic flowshop can be viewed as a small fully automated, and therefore unmanned, manufacturing system, moreover such small production systems are the building blocks of larger automated manfacturing systems. As such, an understanding of the problems arising when planning, scheduling and controling

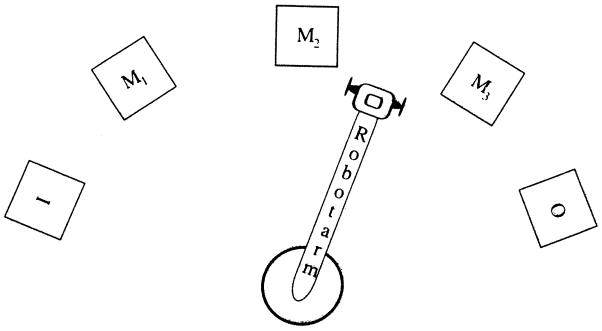


Figure 1: A 3-machine robotic flowshop

the activities in a robotic flowshop are a prerequisite for efficiently operating many large and expensive automated manufacturing systems. The volumes required to earn back the high purchasing costs of such automated manufacturing systems can often only be achieved when producing for a world wide market, such as the automobile industry or consumer electronics. Typically, these markets are characterized by heavy cost based competition, which poses high demands on production efficiency.

The automation of manufacturing in general has led to a variety of problems in which not only the scheduling of different subsystems (e.g. materials handling systems, tool handling systems, flexible machines), but also the interactions of these subsystems with other subsystems are crucial for the overall production efficiency. Scheduling problems in which the coordination between several automated systems is important have therefore received considerable attention lately in the operations research and, more specifically, scheduling literature. The recently widely investigated robotic flowshop scheduling problems (see Asfahl [1985], Crama and Van de Klundert [1994], Hall et al [1994], Sethi et al [1992], Levner et al [1996], Lei and Wang [1994] and the references therein) form a family of such scheduling problems, in which the interaction between the materials handling system and the machines processing the parts determines the production efficiency.

In the next section we describe a basic scheduling problem that has been widely investigated and contains many related problems as a special case. Further we briefly review in section 2, the literature on this problem. In Section 3 we show that the problem is strongly NP-Complete, which provides justification for the nonpolynomial and/or approximate solution methods proposed by several authors.

2 Robotic flowshop Scheduling Models

To formalize and make precise our description of robotic flowshops, we introduce first some notation. We consider robotic flowshops in which there is an input station I or M_0 and

a separate output station O or M_{m+1}. Further, there are m machines $M_1, \ldots, M_m$. All machines $M_1, \ldots, M_m$ can contain only one part at a time, and hence when a machine has finished processing a part, the robot must unload it before the machines starts processing the next one. The robot can only carry one part at a time. Every part becomes available at M_0, and requires processing on every machine $M_i, i = 1, \ldots, m$, in increasing order of the the indices of the machines. Finally, each part must be delivered at the output device M_{m+1}. There are no buffers in the flowshop, which yields that each part that is between the input and output stations is either at some machine or being carried by the robot. The robot performs four types of operations. As already mentioned, it unloads parts from machines. When a part is unloaded, it is carried to the next machine, and subsequently loaded there. Finally, the robot will be repositioned to be ready to unload another machine et cetera.

Since we are interested in establishing a NP-Completeness proof, we consider a problem that is general enough to contain several widely investigated problems as a special case, but at the same time contains the properties that are widely considered to be characteristic for robotic flowshop scheduling problems. Further we are of course interested in identifying the 'easiest' problem variation that is already strongly NP-Complete.

As a first step, we therefore restrict the analysis to the case where all parts are identical, i.e. have the same processing requirements. Notice that this eliminates entirely the part sequencing decisions that constitute a classical flowshop scheduling problem. The resulting complexity is in the robot sequencing, and its interaction with the machines. A second restriction concerns the robot and the sequence of operations that it executes. From the problem description given above, it should be clear that the robot executes subsequences of operations unload machine M_i, carry the part to machine M_{i+1}, load M_{i+1}. After such a subsequence the robot is repositioned to start such a subsequence again et cetera. This leads to the following definition:

Definition 1 The sequence of robot moves

1. Unload M_i,
2. Travel (with the just unloaded part) from M_i to M_{i+1},
3. Load M_{i+1}

is called *(robot) activity* A_i for $i = 0, \ldots, m$.

It is not hard to see that not every sequence of activities constitutes a feasible sequence of operations for the robot to execute. For example, the robot cannot perform A_i immediately after completing A_i, since machine M_i is still empty.

Definition 2 An infinite sequence π of activities $A_0, \ldots, A_m$ is called a *feasible robot move sequence* if,

1. the robot never has to unload any empty machine,
2. the robot never has to load any loaded machine.

Since production efficiency in repetitive manufacturing environments is usually measured in terms of cycle times or (equivalently) throughput rates, we choose as objective to minimize cycle time. This yields that we are interested in the performance of the flowshop in a steady state, long run, situation. For practical purposes, it is not feasible to specify explicitly all the

operations that the robot must perform in the long run. Instead, it is customary to prescribe some 'short' sequence of robot moves that the robot executes repeatedly, and we consequently restrict the analysis to cycle times that can be achieved when repeatedly executing certain classes of short sequences. In fact, all research presented in the open literature restricts the analysis to repetitively executing a finite robot move sequence. Since each machine must be loaded and unloaded when the flowshop is in operation, a short robot move sequence that is to be repeatedly executed must contain each activity at least once. Sequences in which each activity is executed exactly once form the simplest repeatable sequences:

Definition 3 A 1-unit cycle is a sequence of activities $A_0, \ldots, A_m$ in which each activity occurs exactly once and which constitutes a feasible robot move sequence when executed repeatedly.

Thus, each 1-unit cycle is a permutation of the activities $A_0, A_1, \ldots, A_m$. Interestingly, the converse statement is, in general, also true, i.e.

Theorem 1 (Lieberman and Turksen [1981], Sethi et al. [1992]) Every permutation of the activities $A_0, A_1, \ldots, A_m$ is a 1-unit cycle.

Since we are interested in the long run behavior of the flowshop, we assume in the remainder that we are free to specify the initial loaded/unloaded state of the machines. This yields specifically that we do not have to assume that the flowshop is initially empty.

To complete the problem statement, we continue the description of the flowshop. We require that the travel distances for the robot between machines M_i, M_j are given by means of a symmetric *distance matrix* D, whose elements δ_{ij} satisfy the *triangle equality*:

$$\delta_{ij} + \delta_{jk} = \delta_{ik}, \text{ for } 0 \leq i < j < k \leq m+1.$$

This equality models that the robot travels (or rotates) at constant speed along a trajectory. Further, there is a *loading and unloading time* ϵ_i for each machine $M_i, i = 0, \ldots, m+1$.

The processing requirements of the (identical) parts on machine $M_i, i = 1, \ldots, m$ are given by means of processing windows $[L_i, U_i]$: these requirements mean that each part must spend at least L_i time units and at most U_i time units on machine M_i. Such processing requirements naturally arise for instance in a manufacturing situation where the parts have to undergo some chemical treatment that may last neither too short nor too long (see e.g. Philips and Unger [1988], Lei [1993]). Further, the more common situation where a part may reside at a machine arbitrary long after it has been processed can be modelled by setting $U_i = \infty, i = 1, \ldots, m$. (Hall et al [1994], Sethi et al [1992], Crama and Van de Klundert [1994]). Finally, yet another special case arises when the parts are required to be unloaded as soon as they finish processing, see e.g. Levner et al. [1996]. This can be modelled by setting $L_i = U_i, i = 1, \ldots, m$.

A general problem description is now as follows:

Definition 4 *Robotic Flowshop Scheduling Problem* (RFSP):

INPUT : $D, \epsilon_i, [L_i, U_i]$ $(i = 0, \ldots, m+1)$, integer Z.

QUESTION : Is there a 1-unit cycle which when repeatedly executed yields a cycle time of at most Z.

In the remainder of the paper we will show the RFSP to be strongly NP-complete. Notice that proving NP-completeness for the identical parts case implies NP-completeness for the case were parts may have different processing requirements. To our knowledge, this is the first NP-completeness proof that incorporates a realistic distance matrix. An earlier completeness proof of Lei and Wang [1989] assumed for example non-zero travel time between M_i and M_i. That the modelling of the distance matrix is of major importance, can be concluded from results of Crama and Van de Klundert [1994] and Levner et al. [1996] who provide polynomial algorithms for the cases where $U_i = +\infty$, and $L_i = U_i$ resp., that exploit the triangle equality property of the robot travel times.

To formalize the cycle time minization objective, we define

Definition 5 A *schedule* S is defined as a specification of starting times for each load and unload operation. More specifically, we denote by $S(l,i,t)$ $(S(u,i,t))$ the time at which the t-th loading (unloading) of machine M_i starts in schedule S $(i = 0,\ldots,m,\ t \in \mathbb{N})$.

The reader should notice that it not trivial to find a feasible schedule once the order in which the activities are to be executed is known, since the schedule has to respect the lower- and upperbounds of the processing windows. Notice also that this yields that not every 1-unit cycle is feasible for every problem instance. In addition, it is far from trivial to find a schedule with minimum cycle time once the 1-unit cycle is known. For this reason, researchers have commonly restricted the analysis to the special case where the robot executes a cyclic schedule, namely a so called 1-periodic schedule.

Definition 6 A *schedule* S is 1-periodic if there exists a constant C_S such that $S(l,i,t+1) - S(l,i,t) = C_S$ and $S(u,i,t+1) - S(u,i,t) = C_S$ for all $i = 0,\ldots,m+1, t \in \mathbb{N}$.

Obviously the cycle time of a 1-periodic schedule S equals C_S. Notice that the execution of a 1-periodic schedule forces the robot to repeat a 1-unit cycle, to be called $\pi(S)$. Without loss of generality, assume that the 1-unit cycle start with activity A_0. Then, S is feasible if the following relations are satisfied (Lei [1993]):

If A_{i-1} precedes A_i in $\pi(S)$, then

$$S(u,i,t) - S(l,i,t) - \epsilon_i \geq L_i, \tag{1}$$

$$S(u,i,t) - S(l,i,t) - \epsilon_i \leq U_i. \tag{2}$$

On the other hand, if A_i precedes A_{i-1} in $\pi(S)$, then

$$S(u,i,t) + C_S - S(l,i,t) - \epsilon_i \geq L_i, \tag{3}$$

$$S(u,i,t) + C_S - S(l,i,t) - \epsilon_i \leq U_i. \tag{4}$$

The robot must be allowed enough time to perform each activity:

$$S(u,i,t) + \epsilon_i + \delta_{i,i+1} \leq S(l,i+1,t). \tag{5}$$

Furthermore, if A_k is the activity succeeding A_j in $\pi(S)$ then

$$S(l,j+1,t) + \epsilon_{j+1} + \delta_{j+1,k} \leq S(u,k,t), \tag{6}$$

and, if A_j is the last activity in $\pi(S)$, and A_k the first,

$$S(l,j+1,t) + \epsilon_{j+1} + \delta_{j+1,k} \leq S(u,k,t) + C_S. \tag{7}$$

As Lei [1993] observed, the optimal cycle time can be computed in polynomial time once π is known, since minimizing C_S subject to (1) – (7) yields a linear programming problem. Now, for each π', let $C_{\pi'}$ be the minimum long run cycle time attainable by a schedule S satisfying (1) – (7) such that $\pi(S) = \pi'$. Then, RFSP boils down to determining whether there is a 1-unit cycle π for which $C_\pi \leq Z$.

We finish this section by reviewing the literature in which this problem and closely related ones are addressed. The problem was introduced by Philips and Unger [1976]. They formulate the problem as an integer linear program, and solve some instances using standard software. Lieberman and Turksen [1981] formulate several related problems, e.g. problems in which there is more than one robot or problems in which the cell is not restricted to be a flowshop. Song et al. [1993] propose heuristics to find the optimal k-unit feasible robot move sequence for the no-wait version of this problem. In Lei [1993], the problem of minimizing the cycle time for a given permutation of the activities is shown to be solvable in $O(m^2 \log m \log B)$, where B depends linearly on the input parameters. Lei and Wang [1994], Armstrong, Lei and Gu [1994] and Hanen and Munier [1994] discuss branch & bound procedures for RFSI and alike. In Lei, Armstrong and Gu [1993], and in Lei and Wang [1991], heuristic procedures for a similar problem with multiple robots are given. For a more general overview of materials handling related scheduling problems in robotic cells we refer to Crama [1995] and van de Klundert [1996].

3 The NP-Completeness proof

Theorem 2 RFSP is strongly NP-Complete.

Proof. Membership in NP follows from (1) – (7) (see e.g. Lei [1993]). We show its completeness by giving a reduction from the Bin Packing Problem to RFSP.

Bin Packing :

INPUT : Finite set $V = \{v_1, \ldots, v_q\}$ of items, a size $s(v_i) \in \mathbb{Z}^+$ for each $v_i, i = 1, \ldots, q$, positive integer $K \leq q$ and a positive integer B.

QUESTION : Is there a partition of V into disjoint sets $V_1, \ldots, V_K$ such that the sum of the sizes of the items in each V_i is B or less?

Consider an instance of the Bin Packing problem and assume without loss of generality that $s(v_i) \leq B$ for $i = 1, \ldots, q$. We construct an instance of RFSP as follows. There are $m = 2q + 1 + 2K$ machines. The processing windows of the machines $M_1, \ldots, M_K$ and machines $M_{2q+K+2}, \ldots, M_{2q+2K}$ are $[(4q + 2K + 3)B, (4q + 2K + 3)B]$. The processing window of $M_{2q+2K+1}$ is $[(4q + 2K + 2)B, (4q + 2K + 2)B]$. The windows of the machines $M_{K+2i+1}, i = 0, \ldots, q$ are $[0, +\infty]$. Finally the windows of the machines $M_{K+2i}, i = 1, \ldots, q$ are $[s(v_i), s(v_i)]$. The loading and unloading times ϵ_i are all equal to 0. The travel time between two adjacent machines equals B.

The following claim will be useful in the remainder of the proof :

Claim 1 For all $0 \leq i < K$, if A_i precedes A_{i+1} then activities $A_j, j \geq 2q + K + i + 2$ cannot be performed between the execution of A_i and A_{i+1}.

Proof. The processing window of machine M_{i+1} is $[(4q+2K+3)B, (4q+2K+3)B]$, and hence $(4q+2K+3)B$ time units after loading the machine it must be unloaded. To perform an activity with index at least $2q+K+i+2$ the robot must travel to machines with index at least $2q+K+i+3$. Travelling to machine with index at least $2q+K+i+3$ and back *between* the execution of A_i and A_{i+1}, requires at least $2 \times (2q+K+2)B > (4q+2K+3)B$ time, causing the schedule to be infeasible. ∎

Claim 2 There is a 1-periodic schedule with cycle time $(4q+2K+4)KB+(4q+3K+4)B$ if and only if the bin packing instance is a yes instance.

Proof. To prove the claim, we first show that the activities

$$A_0, \ldots, A_K, A_{2q+K+2}, \ldots, A_{2q+2K+1}$$

must be in some specific order in every permutation that denotes a solution with the desired cycle time. Without loss of generality we may assume A_0 to be the first activity in the permutation. We claim activities A_0 to A_K are in order of their index in every feasible solution. Suppose not: let $i \in \{1, \ldots, K-1\}$ be the smallest index for which A_{i+1} precedes A_i (notice that $i \neq 0$). We consider two cases :

1. $A_{2q+2K+1}$ is scheduled before A_i. Let $A_j, j \leq i$ be the activity such that $A_{2q+2K+1}$ takes place between A_{j-1} and A_j. It follows from the Claim 1 that the schedule is infeasible.

2. $A_{2q+2K+1}$ is scheduled after A_i. This implies that between the execution of A_i in some iteration of the schedule and the execution of A_{i+1} in the next execution of the schedule the robot must perform $A_{2q+2K+1}$ and A_0 in that order. It follows again that the total travel time between A_i and A_{i+1} causes the schedule to be infeasible.

Thus activities $A_0, \ldots, A_K$ must indeed be in increasing order of their index. It is also straightforward to check that $A_{2q+K+1}, A_{2q+K+2}, \ldots, A_{2q+2K+1}$ must occur in this order in any feasible schedule (if $A_{2q+K+i+1}$ is performed before A_{2q+K+i}, then the travel time from the machine $M_{2q+K+i+1}$ to M_0 and back exceeds the processing window of $M_{2q+2K+i+1}$).

Let A_0 start at time 0. Considering the processing windows of machines $M_1, \ldots, M_K$ we can derive that the robot cannot start performing activity $A_i, i = 1, \ldots, K$ before time $(4q+2K+3)iB+iB$. More specifically, A_K cannot be started before $(4q+2K+3)KB+KB$. It can also be concluded from Claim 1 that $A_{2q+2K+1}$ cannot take place before A_K since otherwise the schedule would be infeasible. Combining these two observations leads to the conclusion that the total cycle time must be at least $(4q+2K+3)KB + KB + (2q+K+1)B + B + (2q+2K+2)B = (4q+2K+4)KB + (4q+3K+4)B$. Thus we have proved that this quantity (see Claim 2) is a lowerbound on the cycle time.

Claim 1 implies that $A_{2q+K+1+i}$ cannot precede A_i in any solution, for $i = 1, \ldots, K$. We are now going to show that A_{2q+2K} must precede A_K in every schedule having the desired cycle time. First of all, observe that $A_{2q+2K+1}$ cannot precede A_K. Hence if A_{2q+2K} is scheduled after A_K, it is either scheduled between A_K and $A_{2q+2K+1}$ or after $A_{2q+2K+1}$. In the latter case, the schedule was shown above to be infeasible. Thus A_{2q+2K} is scheduled between A_K and $A_{2q+2K+1}$. Now the total cycle time is at least the sum of the following time periods :

1. The interval from A_0 to the start of A_K, execution of A_K and travel time to machine M_{2q+2K} : taking time $(4q+2K+4)KB + B + (2q+2K-K)B$,

2. perform A_{2q+2K}, and wait or do something else until machine $M_{2q+2K+1}$ has finished processing : $B + (4q + 2K + 2)B$,

3. Unload $M_{2q+2K+1}$, bring the part to the output device and travel back to the input device to start the next execution of A_0 : $B + (2q + 2K + 3)B$.

This would result in a total cycle time of at least $(4q + 2K + 4)KB + B + (2q + 2K - K)B + B + (4q + 2K + 2)B + B + (2q + 2K + 3)B > (4q + 2K + 4)KB + (4q + 3K + 4)B$. Now, since Claim 1 implies that A_{2q+2K} cannot precede A_{K-1}, we know that activities A_{K-1}, A_K and A_{2q+2K} are in the order A_{K-1}, A_{2q+2K}, A_K. It is easy to check that $A_{2q+2K-1}$ cannot be scheduled between A_{K-1} and A_K too. Moreover, since $A_{2q+2K-1}$ cannot be scheduled before A_{K-2}, as results from Claim 1, and cannot be scheduled after A_{2q+2K}, it must be scheduled between A_{K-2} and A_{K-1}. An inductive argument then establishes that $A_{2q+2K-i}$ takes place between A_{K-i-1} and A_{K-i}. Hence we conclude that in any schedule that achieves the desired cycle time, the activities $A_0, \ldots, A_K, A_{2q+2K+1}, \ldots, A_{2q+2K+1}$ must be performed in the order

$$A_0, A_{2q+K+1}, A_1, A_{2q+K+2}, \ldots, A_K, A_{2q+2K+1}.$$

The remainder of the proof is now to show how the other activities must be plugged in, so that a schedule with the desired cycle time is obtained if one exists, and that such a schedule exists if and only if the bin packing instance is a yes instance.

We make three observations :

1. A_{K+2i-1} and $A_{K+2i}, i = 1, \ldots, q$ must always occur consecutively in every feasible permutation, since $s(v_i) \leq B$.

2. All the trajectories traveled between the end of A_i and the start of $A_{i+1}, i = 0, \ldots, K - 1$, can be traveled only twice, since otherwise the processing window of M_{i+1} is violated. This implies that we cannot schedule any activities between A_{2q+K+i} and A_i for $i = 1, \ldots, K$.

3. None of the activities A_i with index $K + 1 \leq i \leq K + 2q$ can be scheduled after $A_{2q+2K+1}$ since otherwise the cycle time will be too large.

Together, these three observations imply that we must schedule $K + 1$ sets of pairs of consecutive activities between A_i and $A_{K+2q+i+1}$, for $i = 0, \ldots, K$. The total travel time between loading M_{i+1} in A_i and unloading M_{i+1} in $A_{i+1}, i = 0, \ldots, K - 1$ amounts $(4q + 2K + 2)B$. In view of the processing windows, this leaves us B time to perform pairs of activities A_{K+2i+1}, A_{K+2i+2}. Between any such pair of activities the robot must wait $s(v_{i+1})$ time. Furthermore, we cannot schedule any activities after the execution of A_K, because of the processing window of $M_{2q+2K+1}$. Thus, a 1-periodic schedule with the desired cycle time exists if and only if the numbers $s(v_i)$ can be partitioned into K sets each having weight no more than B. This proves Claim 2. ∎

Instead of focusing directly on schedules, we have in this paper attempted to make clear the distinction between 1-unit cycles and 1-periodic schedules. Notice now that, in the instances created in the reduction, only 1-periodic schedules can lead to cycle times equal to the lowerbound specified in Claim 2, which is a requirement for a yes-instance. We conclude that not only the problem of finding the optimal 1-periodic schedule, but even our more general statement of RFSP is strongly NP-complete. ∎

In conclusion, it is worth mentioning that the minimum cycle time attainable by a 1-unit cycle can be strictly larger than the minimum cycle time attainable by any robot move sequence (Lei [1995]). Very little is known however about the cycle times of non 1-unit cycles, and the reduction in cycle time that can be attained through executing more complex robot move sequences.

Acknowledgements. The first author gratefully acknowledges partial support by the Office of Naval Research (grants N00014-92-J1375 and N00014-92-J4083) and by NATO (grant CRG 931531).

References

C.R. Asfahl, *Robots and Manufacturing Automation*, John Wiley and Sons, New York, 1985.

Y. Crama, Combinatorial models for production scheduling in automated manufacturing systems, *14th European Conference on Operational Research, Semi-plenary Papers* (1995) 237-259. To appear in *European Journal of Operational Research.*

Y. Crama and J. van de Klundert, Cyclic scheduling of identical parts in a robotic cell, 1994, to appear in *Operations Research.*

N.G. Hall, H. Kamoun and C. Sriskandarajah, Scheduling in robotic cells: Classification, two and three machine cells, 1993, to appear in *Operations Research.*

J. van de Klundert, *Scheduling Problems in Automated Manufacturing*, Doctoral Dissertation, Maastricht University, 1996.

L. Lei, Determining the optimal starting times in a cyclic schedule with a given route, *Computers and Operations Research*, **20** (8), (1993) 807-816.

L. Lei, 1995, private communication.

L. Lei and T.J. Wang, A proof : The cyclic hoist scheduling problem is NP-Complete. Working Paper 89-16, Graduate School of Management, Rutgers University, 1989.

L. Lei and T.J. Wang, Determining optimal cyclic hoist schedules in a single-hoist electroplating line, *IIE Transactions* **26** (2) (1994) 25-33.

E. Levner, V.B. Kats, and V.E. Levit, An improved algorithm for a cyclic robotic scheduling problem. *Proceedings of the international workshop on intelligent scheduling of robots and flexible manufacturing systems*, Center for Technological Education Holon, Israel, (1996) 129-141.

R.W. Lieberman and I.B. Turksen, Crane scheduling problems, *AIIE Transactions* **13** (1981) 304-311.

L.W. Philips and P.S. Unger, Mathematical programming solution of a hoist scheduling program, *AIIE Transactions* **8** (2), (1976) pp. 219-225.

S.P. Sethi, C. Sriskandarajah, G. Sorger, J. Blazewicz and W. Kubiak, Sequencing of parts and robot moves in a robotic cell, *The International Journal of Flexible Manufacturing Systems* **4** (1992) 331-358.

Asset Liability Management for Pension Funds[1]

A Multistage Chance Constrained Programming Approach

Cees L. Dert
Department BFS, Free University Amsterdam
ABN-AMRO Asset Management,
P.O.Box 283, NL-1000 EA Amsterdam
E-mail: cdert@econ.vu.nl

Abstract

This thesis presents a scenario based optimisation model to analyze the investment policy and funding policy for pension funds, taking into account the development of the liabilities in conjunction with the economic environment. Such a policy will be referred to as an **asset liability management (ALM) policy**.

The model has been developed to compute dynamic ALM policies that:

- guarantee an acceptably small probability of underfunding,
- guarantee sufficiently stable future contributions,
- minimise the present value of expected future contributions by the plan sponsors.

1. Problem Description

Pension Funds

A pension fund will be considered to be an institute that has been set the task to make benefit payments to people that have ended their active career. The payments to be made to the retirees must be in accordance with the benefit formulae that prescribe the flow of payments to which each participant in the fund is entitled. The word **participant** will be used to refer to all members of the pension fund: active members as well as inactive members.

In general, the pension fund has two sources to fund its liabilities: revenues from its asset portfolio (investment income and appreciation of the value of the portfolio) and contributions to the fund. Contributions are, by definition, made by the **sponsor** of the fund. The sponsor can be the employer, the active participants, or a combination thereof. Thus, at given points in time, the value of the assets of the fund is increased by receiving contributions and by appreciation of the value of invested assets and it is decreased by making benefit payments. It is the responsibility of the pension fund to balance this process

[1] This is a reprint of chapter 1, Introduction and Summary, of the author's Phd thesis Asset Liability Management for Pension Funds (Erasmus University, 1995).

in such a way that the fund meets the solvency standards in force, and that all benefit payments, now and in the future, can be made timely.

Important decisions that determine whether or not the pension fund will manage to fulfil its tasks are the **level of contributions** and the allocation of assets over asset classes in which the fund is willing to invest. This allocation is referred to as the **asset mix**.

These decisions cannot be made freely. The **level of contributions** has to be set in such a way that the sponsor of the fund is able and willing to pay them. This constraint is often reflected by a maximum level of contributions as a percentage of the costs of wages. Moreover, it is customary that annual hikes in contribution, again, as a percentage of the costs of wages, may not exceed a given level.

In principle, the fund is not restricted in its **choice of asset** mix. However, there are widely accepted perceptions of acceptable asset mixes which, in practice, result in upper and lower bounds on the percentage of assets to be invested in each asset category. Moreover, one has to heed constraints that are implied by the size and liquidity of the capital markets of interest, relative to the value of the securities that one would want to trade in a given period of time.

It depends on the ratio of income from contributions and revenues from the investment portfolio which decision, contribution level or asset allocation, is the more important one. In general, the higher the degree to which the pension fund has matured, i.e., the larger the percentage of participants who have ended their active career, the greater the relative impact of the investment decisions.

Although the way in which the **level of future benefit payments** will be determined is given by the benefit formulae, the actual level is uncertain. It is subject to the development of the characteristics of the participants which are determined by future career paths, life and death etc. The major source of uncertainty that affects the level of future benefit payments to be made by many Dutch pension funds, is the future development of price inflation and wage inflation: at retirement, the level of old age pension is usually 70% of the final salary. This pension includes a state pension to a fixed amount. It follows that pension rights of active participants that have been earned over past years of service will be increased by wage inflation. The benefits of inactive participants are often indexed with price inflation.

Once the value of assets proves to be insufficient to make benefits payments that are due, it is in general too late to take any measures to strengthen the financial position of the fund. To avoid this potential problem, the regulating authorities, in The Netherlands the Insurance Chamber have formulated **solvency requirements** for pension funds. They see to it that, at the end of each year, the pension fund has accumulated a level of assets that is sufficient to fund its liabilities.

It seems only natural to require the present value of assets to be at least as high as the present value of liabilities. However, the investment returns as well as the level of future benefit payments are uncertain. As a consequence, it is unclear what the minimal present value of assets is that is sufficient to fund future benefit payments. Neither the present value of assets nor the present value of liabilities can be determined by a universally

accepted method. In this monograph, the **assets** will be valued against their market prices. The valuation of liabilities is the domain of the actuary. Our ALM approach can be used in conjunction with any actuarial method of valuing liabilities. Nevertheless, to appreciate the problem of ALM, it is useful to have some background in actuarial principles. The **present value of liabilities** is usually determined by computing the present value of the expected future benefit payments. Given the characteristics of the current participants in the fund, the expected development of the characteristics (based on mortality tables, invalidity chances etc.) is computed. In conjunction with the benefit formulae, this development serves to compute the expected annual benefit payments for the planning period. Then, the present value of the liabilities can be obtained by discounting this flow of expected benefits. The discount rate that is used to compute the present value of the liabilities is often referred to as the **actuarial rate**. In The Netherlands, the annual actuarial rate that is commonly used to discount liabilities is 4%.

It is tempting to take a clear stand in the ongoing debate on the **appropriate level of the actuarial rate**. This discussion is frequently blurred by the fact that a substantial portion of the liabilities of Dutch pension funds stems from indexation of future benefits with price inflation and/or wage inflation. However, the indexation is usually conditional on the financial position of the pension fund. An actuarial rate equal to 4% can be considerd high if it is used to discount indexed liabilities: one would have to realise an investment return equal to 48% annually over the past 50 years. On the other hand, if the benefit formulae do not contain any indexation promises, then a 4% discount rate seems to be rather low: over the past 50 years, an investor could easily have secured an average return on investments of 6%, without superb investment timing and without having to accept significant price risk or credit risk.

In the sequel, we shall not distinguish between conditional and unconditional liabilities. Liabilities will refer to the sum of **conditional and unconditional liabilities**. Thus, if the benefit formulae contain conditional indexation promises, our ALM approach will aim for a policy that enables one to make indexed benefit payments. As a consequence, one would expect that the minimum funding levels that follow from solutions to our ALM approach will generally exceed the minimum levels that are implied by solvency requirements which have been formulated solely on the basis of unconditional promises.

ALM Policy

A starting point for the analysis is the present state of a pension fund, defined by its actuarial and financial situation (asset value, premium reserve, level of benefit payments etc.), the benefit formulae and/or contribution formulae and the characteristics of the participants.

A good ALM strategy consists of investment decisions and decisions on the level of contributions that result in a desirable **risk/reward** structure with respect to the financial development of a pension plan. It minimises the cost of funding while safeguarding the pension fund's ability to meet its liabilities. The fund should be able to make all benefit payments timely, without becoming underfunded. Given these requirements, the present value of contributions to the fund should be minimised and contributions may be raised only modestly from one year to the next. Unfortunately, even an impeccable implementation of an excellent ALM policy cannot guarantee that all liabilities can be met under

all circumstances. For example, when liabilities are indexed with inflation, exceptional situations may occur, in which inflation rates become so high that it is impossible to meet all liabilities, other than by raising contributions to a fantastic level. Since inflation rates can become very high over extended periods of time, one has to accept that there is a probability that the pension fund cannot meet its funding requirements. This probability is referred to as the **probability of underfunding**. To account for the fact that one cannot expect a pension fund to meet solvency standards under all circumstances, the solvency requirement has to be posed as a chance constraint. I.e., the ALM policy should ensure that the probability of becoming underfunded does not exceed a given level.

Neither asset mixes nor levels of contribution will be fixed for the entire planning period. Instead, decisions will be revisited when warranted by newly emerged circumstances, such as a changes in the funding level and altered perceptions of the future development of the world. However, stability requirements on the ALM policy may imply that one can only deviate so far from decisions that have been made in the past. These observations show that **current decisions and future decisions cannot be made independently**. Therefore, an ALM policy should consist of decisions to be made now and sequences of decisions to be made in the future. Future decisions should be conditional on the situation that has emerged at the time of decision making. Current decisions should anticipate on the ability to adjust decisions later on. Furthermore, to the extent to which they restrict choices in the future, they should reflect a correct trade-off of shorter term effects and longer term effects. Such a policy is referred to as a **dynamic policy**.

Defined Benefit Plans and Defined Contribution Plans

In the above description of the ALM problem, is has been assumed that the benefit formulae are given, whereas the contributions to the fund are to be determined. This is the case with **benefit defined** pension plans. In contrast with this type of pension plan, a **contribution defined** plan is characterised by fixed contribution formulae and uncertain benefit payments. Although the models and illustrations in this thesis assume a defined benefit pension plan, the approach that we present is also suited to determine investment policies for defined contribution plans.

2. Modelling an Uncertain Future by Scenarios

One of the central issues in ALM modelling, is the way in which uncertainty is modelled. Here, uncertainty will be modelled by a large number of scenarios, each of which reflects a plausible development of the environment within which ALM decisions have to be made. More specifically, future environments will be reflected by **states of the world**, which are defined by the level of actuarial reserve, the level of benefit payments, the level of costs of wages and the return on each of the asset classes over the previous period. These states of the world are independent of the decisions to be made with respect to asset mix and contribution policy. They are defined completely by factors that are **exogenous to the decision model**. A path through consecutive states of the world will be referred to as a **scenario**.

After generating a large set of scenarios, it is assumed that this set is a reasonable **representation of the uncertain future**: the model assumption is made that one of these paths will materialize. The uncertainty is still preserved in that the decision maker does

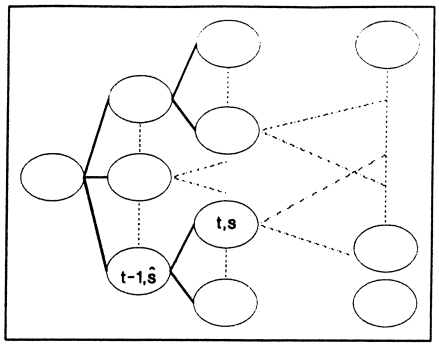


Figure 1 *A Scenario Structure for ALM*

not know yet which scenario describes the true future states of the world.

Scenario Structure

In order to model a multistage decision process with recourse, the states must be structured so that they can reflect the notion of time and the principle of information being revealed as time goes by. The desired information structure and the notion of time are ensured by imposing the tree shape scenario structure as depicted in Figure 1.

At point in time 0, there is only one state of the world: the state that can currently be observed. Given this state of the world there are many states of the world which could emerge by the end of period. Which one of them actually materializes will be known only at time. In general, given state of the world s at time t, there are many states at time $t+1$ which succeed (t, s) with positive probability. This reflects the uncertainty regarding the future environment. At any point in time, the history by which the prevailing state of the world was reached is known: the scenarios are structured so that each node has a unique predecessor.

Statistics of endogenous and exogenous state variables, such as the probability of underfunding and the expected surplus, play an important role in the ALM model. In order to compute these statistics, the scenarios have to be equipped with a **probability structure** on which the statistics can be defined. This structure should specify the probability of each state of the world to occur; unconditional, as well as conditional on the state of the world that has prevailed at the preceding point in time.

Consistency and Variety

The scenarios should be generated in such a way that future states of the world are consistent, i.e., stochastic and deterministic **relationships** between state variables **at each point in time** should be reflected correctly, subsequent states of the world should reflect the **intertemporal relationships** between state variables, and the **variety** of the states of the world should suffice to capture all future circumstances that one would want to reckon with. The scenario generator that is presented in chapter 4 satisfies these require-

ments. The ALM model that we propose, however, can be used in conjunction with any scenario generator that meets the requirements that have been specified above. For example, one could choose to employ a model that is based on economic theory, instead of the time series model that has been included in our scenario generator.

A noteworthy special case of reflecting sufficient uncertainty is the requirement that the scenarios may not allow for **arbitrage opportunities**. I.e., they may not include any states of the world in which it is possible to compose investment portfolios at price zero which have positive probability of a positive pay out, and which never have a negative pay out. In reality these opportunities will not occur to an extent that it is possible to exploit them systematically in an ALM policy. Therefore a realistic model should not allow for arbitrage. Section 4.3.2 has been devoted to this subject. There, it is proven that the continuous probability distribution of states of the world that underlies our scenario generator does not allow for arbitrage opportunities. Moreover, for finite sample sizes, an algorithm is given that eliminates all arbitrage opportunities, if any, by extending a sample of given size by one well chosen state of the world.

3. The Position of our ALM Approach in the Literature

Chapter 2 contains an extensive discussion of publications on ALM for pension funds. Here, we shall restrict ourselves to a short characterisation of the main types of models, after which we shall position our approach relative to the existing methods.

One of the criteria that will be used to classify ALM approaches is whether or not the approach is dynamic. **Dynamic models** can be employed to compute policies that consist of actions to be taken now, and sequences of reactions to future developments. In contrast with dynamic models, **static models** do not make optimal use of the opportunity to react to future circumstances. Static decisions do not anticipate on the ability of making recourse decisions. As a consequence, the employment of static models may lead to:

- current decisions that do not reflect a correct trade-off between short term effects and longer term effects,

- current decisions that are extremely conservative because the ability to reduce risks in the future, when necessary, is neglected. This will cause the costs of funding to turn out unnecessarily high.

Still, most of the models that are currently being used for ALM decision support are static. This is probably caused by the fact that the computational effort to formulate and solve dynamic models for realistic problem sizes is large in comparison with static models. If computationally feasible, however, one should prefer a dynamic model.

Many ALM publications are based on **mean-variance analyses** of the surplus of a pension fund at a given horizon, taking into account stochastic liabilities. The trade-off between risk and reward, in this approach, is usually quantified as the trade-off between the expected level of the surplus at a given horizon and the standard deviation thereof. One of the main drawbacks of standard deviation as a measure of risk is that it does not distinguish between returns higher than expected and returns lower than expected. **Chance constrained programming** offers an alternate to quantifying risk by standard deviation which does not

suffer from this shortcoming. One defines the probability that a certain event will happen as a function of the model's decision variables. The probability of undesirable situations to occur can then be bounded by including constraints on the value of the associated statistics. To facilitate tractability, chance constrained models are usually presented in combination with the assumption that exogenous stochastic parameters, e.g., the growth of liabilities and investment returns, follow a probability distribution that is convenient from a computational point of view.

More recently published models on ALM are **stochastic programming models**. These models can be used to compute dynamic ALM strategies that are based on a set of scenarios which reflect the future circumstances that one wants to take into account. In principle, these scenarios can be based on any stochastic process that is considered to be appropriate to describe the environment for ALM decisions.

We propose a **mixed integer stochastic programming model**. It has the desirable properties of the aforementioned stochastic programming models in the sense that it can be employed to determine dynamic ALM policies that are based on scenarios, which can reflect any set of assumptions that one chooses to make on future circumstances. In contrast with the stochastic programming models that were mentioned earlier, our ALM model includes binary variables that enables one to count the number of times that a certain event happens. This possibility has been used to formulate chance constraints that are based on the probability distribution of states of the world that follows from the scenarios. In the case of ALM, this property is used to model and to restrict the probability of underfunding: at the planning horizon, as well as at intertemporary points in time. The choice has been made to sacrifice the ability to compute optimal solutions to problems of small sizes. Instead, we have opted for developing a heuristic by which good solutions can be computed to problems, the size of which suffices to model realistic problems. The main characteristics of the models that have been discussed in this section are presented in Table 1.

Table 1. A Classification of ALM Approaches

approach	dynamic or static	explicitly consider probability of underfunding	realistic assumptions on probability distributions
mean-variance models	static	no	no
chance constrained programming	static	yes	no
stochastic programming	dynamic	no	yes
our approach	dynamic	yes	yes

To conclude, let us summarize the properties which an ALM model should satisfy: The model should be suitable to determine a **dynamic ALM strategy**, consisting of an invest-

ment strategy and a contribution policy, which account for the development of liabilities. Decisions to be made now should anticipate on the ability to make state dependent decisions in the future. They should be the result of a **trade-off between short term effects and long term effects**. **Risk** must be reflected by the probability of underfunding and the magnitude of deficits when they occur. The model should accommodate the employment of realistic probability distributions of exogenous random variables, and, finally, the model should be feasible from a **computational** point of view.

To our knowledge, the ALM approach that is presented here, is the first one that meets all these requirements. Computational results, obtained on realistic problem instances, which are presented in summary in section 7, corroborate the theoretical notion that this type of model is superior to models that have been presented in the literature which do not meet all of the aforementioned requirements.

4. A Scenario Generator for Asset Liability Management

We have described the technical properties that the scenario structure should have, in order to serve as a framework within which dynamic ALM strategies can be analyzed and optimised. Let us now turn to the question as to what set of scenarios can serve as a reasonable representation of the future.

Different policy makers may consider different factors to be of interest to their ALM decisions. They may choose to base their policy on different assumptions and these assumptions should be reflected by the scenarios. Therefore, our ALM approach has been designed in such a way that it can be used in conjunction with any scenario generator that satisfies the conditions that have been stated in 4.2.

Figure 2 pictures the scenario generator that has been used to obtain the computational results that are reported in chapter 7. A **time series model** is employed **to generate future developments of price inflation, wage inflation and returns on stocks, bonds, cash and real estate** in such a way that means, standard deviations, autocorrelations and cross correlations between state variables are consistent with historical patterns. Given the benefit formulae and all relevant data on the participants (e.g. civil status, age, gender, salary, earned pension rights, medical status, social status), a **Markov model** is employed to determine the future development of each individual that currently participates in the pension fund. For an employee, for example, it is determined whether he remains alive, retires, resigns, gets disabled and/or is promoted to another job category on an annual basis. These transitions are determined by probabilities which depend on characteristics of the individuals such as age, gender and employee-category. Additional promotions and the recruitment of **new employees** are determined in line with the intended personnel policy.

Given the development of wage inflation, the career of each employee in each future state of the world and the current reward system, **the cost of salaries, the level of benefit payments and the actuarial value of the liabilities** can be computed for each state of the world.

All information to describe states of the world is now available: investment returns on all asset classes have been obtained from the time series model, the administrative software

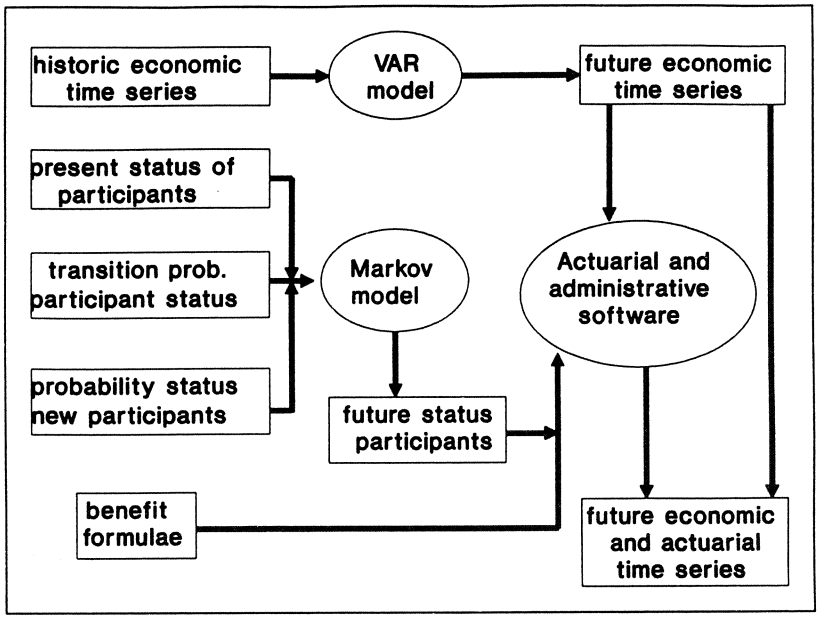


Figure 2 *A Scenario Generator for ALM*

has generated the corresponding cost of wages and, to conclude, the actuarial software has provided the corresponding levels of benefit payments and actuarial reserves. Notice, that the scenario generator has been structured in such a way, that consistency between state variables within a state, as well as consistency between states of the world is preserved.

Once the scenarios have been generated, the following information is available for each state of the world:

- the level of benefit payments,
- the level of the actuarial reserve,
- the level of the costs of wages,
- the return on each of the asset classes over the preceding period of time.

Furthermore, the scenario structure implies that it is also known for each state

- what the preceding state of the world has been,
- which states of the world are possible successors, and what the probability is that they will emerge, given the current state.

All information that is contained in the scenarios is independent of ALM decisions. They

are the subject of the next section.

5. A Dynamic Optimisation Model for Asset Liability Management

Chapter 3 presents an optimisation model that determines an ALM policy that consists of an asset mix and a contribution level for each state of the world. These decisions also determine the level of asset value and, in combination with the exogenously given level of liabilities, the funding level in each state of the world. The decisions in all states of the world are made simultaneously. This allows for a trade-off between longer term effects and shorter term effects, as well as for a trade-off between the outcome of decisions in different future states of the world.

The model has been developed to compute dynamic ALM policies that:

- guarantee an acceptably small probability of underfunding,
- guarantee sufficiently stable future contributions,
- minimise the present value of expected future contributions by the plan sponsors.

Because the probability of underfunding is an important concept in ALM and because it can be modelled in many ways which have substantial implications for its interpretation, we shall discuss it at more length in the following paragraph.

The Probability of Underfunding

The probability of underfunding has been defined on the set of scenarios. For example, suppose that there are 100 states of the world, each of which succeeds a given state of the world with probability 1/100, then the probability of underfunding, when starting from the given state of the world is equal to 1/100 times the number of succeeding states in which underfunding occurs. In general, if a maximum probability of underfunding equal to Ψ^u is considered to be acceptable, then this is reflected by constraints which ensure that for each state of the world, the probability of being succeeded by a state in which underfunding occurs, is less than or equal to Ψ^u. The probability of underfunding has been modelled in such a way that:

1. The model can account for **any probability distribution** that can be reflected by the scenarios. That includes distributions that are specified implicitly, such as the distribution of liabilities which may be given by benefit formulae in the form of computer programmes.

2. Probabilities of underfunding are **endogenous** to the model.

3. Probabilities of underfunding are taken into account **explicitly**, at **intertemporal** points in time, as well as at the **planning horizon**.

Underfunding

What would happen when a situation of underfunding occurs ? It is not clear what would happen in practice. In our model, however, it will be assumed that a remedial payment is

made which is precisely sufficient to **restore the required funding level**. The remedial contributions are included in the costs of funding. Thus, the probability of underfunding, as well as the magnitude of deficits when they occur are taken into account. The structure of the model can accommodate other assumptions with respect to measures to be taken in situations of underfunding as well. Alternative reactions that can be accommodated include remedial contributions to be made during a prespecified number of years until the desired funding level has been restored and, entirely or partially, failing to meet conditional indexation promises.

In summary, the ALM model that will be presented in chapter 3 can be used to compute ALM strategies which specify investment decisions and contribution levels to be set under a wide range of future circumstances. The decisions are made in such a way that the present value of expected contributions to the fund is minimal, subject to raising sufficiently stable annual contributions and the probability of underfunding at the end of each year being acceptably small when starting from the current situation, as well as from all future states of the world that the policy makers of the pension fund choose to take into account.

6. Computational Complexity

The proposed ALM model is a mixed integer linear problem, the size of which increases exponentially with the number decision moments. As a consequence, it is very difficult to solve the model to optimality for realistic problem sizes. Therefore, chapters 5 and 6 have been devoted to the development of a heuristic by which a good, but not necessarily optimal, solution to the ALM model can be obtained.

Chapter 5 presents a special case of the general scenario structure that has been presented earlier. Using this new structure, a heuristic can be used to compute good solutions to the ALM model. The heuristic consists of a backward procedure and a forward procedure. In the **backward procedure**, a sequence of two stage problems is solved; one for each point in time at which state dependent decisions can be made. The solutions to these problems serve to specify desirable situations of the pension fund in each state of the world. However, the two stage problems have not been formulated in such a way that it is always feasible to determine an ALM strategy that results in attaining the desirable situations in all states of the world. Therefore, the backward procedure is followed by a **forward procedure**. The latter consists of solving a one period model for each state world. Given decisions at preceding points in time, it minimises deviations from the desired situations that have been obtained from the backward procedure, subject to the constraints that the ALM policy should satisfy. The computational effort to solve the ALM model by means of the heuristic is proportional to the number decision moments.

The computational effort for each point in time is dependent on the number of states of the world that has to be taken into account. The fewer states of the world the smaller the computational effort to solve the models. Thus, the fewer the better. On the other hand, the number of states of the world should be sufficiently large to represent the underlying continuous probability distribution. In chapter 6, a **variance reduction technique**, importance sampling, will be employed to reduce the number of states of the world that is required to obtain a sufficiently accurate representation of underlying continuous probability distribution of states of the world.

7. Computational Experiments

Chapter 7 presents results of computational experiments with the ALM model. In order to obtain insight in the behaviour of the model on realistic problem instances, it has been applied to the data of a Dutch pension fund with an actuarial reserve in excess of 16 billion Dfl. and approximately 1,020,000 participants of which 240,000 are still in their active career.

One would expect ALM decisions for a wealthy pension fund to be different from those for a thinly funded pension fund. Therefore, three settings have been selected, which differ in the initial funding level and in the amount by which annual contributions may be raised from one year to the next:

Setting 1: a low initial funding level and a low maximum increase of contributions,

Setting 2: a high initial funding level and a high maximum increase of contributions,

Setting 3: the initial funding level to be determined by the ALM model in such a way that costs of funding are minimised subject to satisfying the solvency constraints with moderate maximum increases of contribution.

In all settings, the probability of underfunding was allowed to be at most 5% in each year. The cost figures in Table 2 and Table 3 are presented in mln. Dfl.

Table 2. Summary of computational results from the ALM model

Setting	Initial asset mix				Underfunding		PV Total costs
	Cash	Stocks	Property	Bonds	PV remedial contributions	Average probability	
1	66	21	13	0	699	10.6%	23,002
2	52	44	4	0	19	0.4%	25,356
3	0	100	0	0	42	0.6%	24,682

In order to compare the results that are shown in Table 2 to other approaches, static decision rules have been determined to specify time and state dependent contribution levels, in combination with optimal static asset mixes. These results are presented in Table 3. As can be verified from the tables, the results from the dynamic ALM model are superior in all settings. In setting 1, because it does not violate solvency constraints as much as the static model. In settings 2 and 3 in which both models present feasible

Table 3. Summary of computational results from static decisions

Setting	Initial asset mix				Underfunding		PV Total costs
	Cash	Stocks	Property	Bonds	PV remedial contributions	Average probability	
1	49	18	33	0	24,340	21.8%	22,906
2	35	28	31	6	1276	2.3%	30,063
3	14	52	32	2	827	1.9%	27,099

policies, the present value (PV) of the costs of funding is lower. Moreover, the present values of remedial contributions to be made when the static policy is pursued are 20 to 60 times as high as those that are associated with the dynamic policy from the ALM model. In order to assess the extent to which the results from the ALM model are due to its dynamic character, the results have been compared to results from a model that makes optimal time dependent and state dependent decisions, taking into account a horizon of one year. This comparison indicates that the results from the ALM model are largely determined by its dynamic character.

The computational results which are presented in more detail in chapter 7, provide the following insights with respect to the ALM approach presented in this monograph.

1. Dynamic ALM strategies lead to current decisions that are different from decisions to be made when following a static policy.

2. In comparison to the static models, the employment of the ALM model has resulted in strategies of which the costs of funding are lower, the probabilities of underfunding are substantially smaller and the magnitude of deficits, reflected by the costs of remedial contributions, has been reduced dramatically.

3. The favourable outcome of the comparison of policies determined by the ALM model with policies determined by static decisions, are to a major extent due to:

- the fact that probabilities of underfunding at intertemporal points in time as well as at the planning horizon are endogenous to the model and have been modelled explicitly, and

- the dynamic character of the ALM model which enables the policies to react to situations that have emerged at the time of decision making and to reflect a correct trade-off between their longer term effects and their short term effects.

Minimizing leadtimes by optimizing batch sizes

Peter Tielemans
Moret Ernst & Young Management Consultants
Varrolaan 100, 3584 BW Utrecht
E-mail: ptiel@mc.mey.nl

1 Introduction

In the last decade a new class of models was proposed for aggregate production planning deviating from more traditional models both in the performance criterion used as in the modelling approach taken. Typically, traditional models taking into account capacity restrictions in order to support decision making in production planning are deterministic cost models formulated as a mathematical program. Examples of this modelling approach are the *Economic Lot Sizing Problem* (ELSP), the *Capacitated Lot Sizing Problem* (CLSP), the *Continuous Setup Lot sizing Problem* (CSLP) and the *Discrete Lot sizing and Scheduling Problem* (DLSP). A review of these models is given by Salomon [9]. In the newer approach, in stead of the total average costs, manufacturing lead time is considered as performance measure and concepts of queueing theory are used to introduce stochasticity into the planning models. Karmarkar [4] was one of the first to advocate these stochastic lead time models. Generally, three basic motives can be distinguished that ground this new approach:

- **The performance criterion: costs versus physical measures**
  The lion's share of all models proposed to support tactical production planning are cost models. Time-phased production quantities have to be determined in such way that the total costs are minimized. Apart from the physical decision variables, production quantities and workforce (time), such models need values for parameters as the cost of a setup and the cost of holding one item one time unit in stock. It practice it is difficult to determine appropriate values for these cost factors in an unambiguous way. For instance, the setup cost factor often is used to express the costs of capacity restrictions (non-zero setup times). As Karmarkar [4] states, these costs cannot be considered fixed, because they are dependent on the current state of the manufacturing system, for instance the work load. Because physical measures as the manufacturing lead time or the total amount of work-in-process are a direct consequence of capacity restrictions, they can be considered as alternative measures of manufacturing performance.
- **The modelling approach: Deterministic versus stochastic models**
  Most aggregate production planning models are deterministic models. These capacitated models consider a bottleneck in the production chain to construct an appropriate production plan. A typical property of constrained deterministic models is that the constraint is allowed to be binding, i.e. all time available for production can be used for setups or product transformation. However, at high levels of utilization, uncertainties in the job arrival process and the production process will result in congestion. In practice

[1] This paper is a summary of the thesis 'Lead Time Performance in Manufacturing Systems

the long queueing delays resulting from these congestion phenomena frequently can be observed. In contrast queueing models can account for the queueing delays and the queueing inventory.

- **Competitive advantage: Price versus time**
  Traditionally, the price and quality of a product had the major impact on its competitive advantage. However, in the last decade time became a competitive weapon (Stalk, [10]). For all kinds of make-to-order businesses, the advantage of quick delivery will be obvious. In addition, as the Japanese successes in the late eighties showed, the time-to-market can be of overriding importance with regard to the success of product innovation.

2 Lead time as performance criterion

The recent reversal of managements perception of long manufacturing lead times from a necessity to resolve conflicts between various jobs requiring the same resources, to a daemon that can affect a firms competitive position seriously, can be regarded as a drive to manage lead time more explicitly. Several advantages of considering lead time as a measure of manufacturing performance can be mentioned:

- Manufacturing lead time will be the determinant of a firm's profitability and market share in a time based competition environment.
- Short manufacturing lead times allow for a quick response to changes in customer demand. Demand forecasts have to cover a shorter time horizon and in this way the probability of schedule changes decreases.
- A decreased time-to-market for product innovations gives a firm the great opportunity to offer technically more sophisticated products. In an early stage a large market share can be picked and possibly product standards can be dictated to competitors.
- Shortening lead times will decline work-in-process inventories ('zero inventory') and stimulate a high inventory turn-over rate. Eventual product deterioration (perishability, fashion) is driven back.
- Shorter lead times will reduce the amount of safety stocks needed in make-to-stock production more than proportionally. The amount of safety stock depends on the variance and the mean values of the lead time and the demand rate. In general, a decrease in lead time will go along with a decrease in the lead time variance. Both effects will reduce the amount of safety stock needed. In addition, a smaller average lead time requires a shorter horizon of demand forecasts. Demand estimates become worse for longer planning horizons, so by shortening the lead time the variance in the demand estimates due to forecast errors decreases.
- In for instance MRP-systems, safety times in stead of safety stocks are often used as a buffer against forecast errors. These safety times engender early production order releases as protection against variance in completion times and will in this way increase work-in-process inventory. In correspondence with determining safety stocks, lead time reduction will lead to shorter safety times.
- The length of the cash flow cycle is decreased when the time between the expenditures for resources and material and the revenues is shortened.
- The delay between fabrication and inspection is shortened, allowing for a quicker response to quality problems.
- Lead times easily fit within the MRP-context. Taking lead times as performance measure will enable joint decision making on lot sizes and lead time values for the construction of an MPS.

- Lead time data are elementary for due date setting. The variability in lead time estimates causes a trade-off between setting a close due date that may not be met, and a due date far away that will cause an increase of average work-in-process.
- By taking lead time as performance measure at a tactical decision stage a more direct hierarchical approach is obtained: sequencing and scheduling rules are based on lead time related quantities as completion times and due dates.
- Lead time is an easily understood performance measure for shop floor workers. Their daily observations on problems that slow up the throughput rate can help to improve the performance of the production process.

Although some factors mentioned above cannot be quantified accurately, it may appear that lead time related measures can be appropriate measures of manufacturing performance.

3 The modelling approach

In the thesis a single machine in a job shop is considered. Different products have to be processed on the machine and change-overs will be needed between production of the different batches. Batches arriving at the production facility queue up in front of the machine when the machine is busy. The waiting batches are processed in the order they arrived. A batch will be transported to the next production stage not until the whole batch has been processed.
The manufacturing lead time of a batch at the single machine will be measured from the time the batch arrives at the production facility until the whole batch has been processed and released for transportation.

In modelling this situation, the basic modelling principles proposed by Karmarkar, Kekre and Kekre [6] are followed. The production facility is modelled as a queueing system. Considering a queueing system additional assumptions have to be made on:

- **The arrival process**
  It is assumed that the interarrival times of the batches at the production facility are negative exponentially distributed. The basic idea to substantiate this assumption is that the superposition of a large number of independent batch arrival streams can be approximated by a Poisson process.
  When the arrivals at the processing center originate from a large number of overlapping and uncoordinated product flows this assumptions is not unreasonable. In Chapter 6 of the thesis the justification of this assumption in various situations and its influence on model outcomes, is discussed in more detail. This assumption of Markovian arrivals is common in multi-item queueing models of production facilities, see Axsäter and Olhager [1], Cohen and Lee [2], Jönnson and Silver [3], Karmarkar [4], Karmarkar, Kekre and Kekre [5], [6], Zipkin [13].
- **The service process**
  In order to avoid superfluous complexity both the item-dependent setup times and the item-dependent processing rates are taken constant. It is assumed that every time the processing of a new batch is started a setup is executed, even when the next batch occasionally is of the same type as the one just finished.
- **The selection discipline**
  The jobs in the queue are processed according to a *First-Come, First-Served (FCFS)* rule. It should be noted that *Last-Come, First-Served (LCFS)* and the *Random Order (RO)* rule will result in the same average queueing delay although the total distribution of the queueing delay will differ in those three cases.

We take T_j as the total average lead time for item i, W as the average queueing delay, n as the number of item types, μ_i as the demand rate for item i, p_i as the processing rate for item i, τ_i as the setup time for item i, q_i as the batch size for item i and $\mathbf{q} = 1_1, \ldots q_n$. For an M/G/1 queueing system the Pollackzek-Khintchine formula can be used to derive expressions for the average queueing delay and the average total lead time for item i, respectively:

$$W(\mathbf{q}) = \frac{\sum_i^n \frac{\mu_i}{q_1}\left(\tau_i^2 + \frac{2\tau_i}{p_i}q_i + \frac{1}{p_i^2}q_i^2\right)}{2\left(1 - \sum_{j=1}^n \frac{\mu_j}{q_j}(\tau)j + \frac{q_j}{p_j}\right)}$$

$$T_i = W(\mathbf{q}) + \tau_i + \frac{q_i}{p_i}$$

4 The analysis

In Chapters 3 of the thesis the expression for the average queueing delay is analysed. This chapter sets out to find analytical expressions for the optimal batch sizes and the minimal expected queueing delay that results from optimal batching. Finding a wieldy, simple expression for the minimal expected queueing delay as a function of parameters is difficult for the general multi-item case. Yet, this is required if one wishes to study the impact of changes of problem parameters on the minimal expected queueing time in an analytical tractible manner. In chapter 3 a lower bound on the minimal expected queueing delay is derived that:

- is a wieldy, simple algebraic expression in input parameters such as setup times, production rates and demand rates;
- is shown to be very tight through extensive numerical experiments;
- is shown to share with the minimal expected queueing delay several structural properties concerning the dependency on setup times and concerning capacity expansion;
- in particular captures the concave dependency of the expected queueing delay on setup times;

The ingenuousness of these expressions in the setup times and utilization levels of the different items, allows an interpretation in terms of product variety and product flexibility. In addition, it is shown how this bound can be used as an approximation of the minimal expected queueing delay in order to support decision making on setup time reduction and capacity expansion.

The final part of Chapter 3 is devoted to the setup utilization (the fraction of available production time spent on executing setups) as production planning and control tool. A characteristic feature of the model introduced in Chapter 3 is that at optimal batch sizes setup utilization is maximal for the case that the setup times for all items are equal. This model property is used to derive a simple rule-of-thumb that gives the maximal setup utilization, see also Kuik and Tielemans [12]. In practice, these kind of rules can be used to guide multi-item batching decisions.

As is shown in the definition of T_i given above, the total manufacturing lead time of a batch is computed by adding the setup time and the real processing time of a batch to the queueing delay. In this way an item type dependent total lead time is obtained. The average lead time can be defined by taking a weighted average over the item types. The weighting factors can be defined on the basis of total work-in-process inventory or total inventory cost considerations. In Chapter 4 of the thesis, the average lead time is taken as performance criterion for batching decisions. For the case that the setup times and

the quotient of the weighting factor and the productive utilization, are item independent, exact closed form analytical expressions can be found. However, when these conditions do not hold one has to rely, again, on approximation techniques. Two approaches are compared. In the first approach, the results of Chapter 3 are used by quantifying the influence of a batching decision based on queueing delay minimization on total average lead time performance. As can be expected this approach is satisfying for high levels of total utilization but won't suffice for low utilization levels. The second approach uses the relative insensitivity of the optimal solution for small changes in the total setup utilization, to derive an approximate expression for the minimal average lead time. On the basis of a large numerical test it is shown that this approximation performs well for high as well as low utilization levels (average deviation less than 0.5 %).

Especially in due date bargaining and safety stock determination, also the variability of the manufacturing lead time will be important. In Chapter 5 of the thesis, it is studied how batch sizes influence the variance in the lead time for the case of identical items. The difference between the batch size that minimizes the variance on the one hand and the batch size that minimizes the queueing delay or the time in system on the other hand can mount to 100% for extremely low levels of utilization but disappears asymptotically for utilization levels approaching 1. The difference between the minimal variance and the variance that occurs at a batching decision based on time in system analysis, diverges for a total utilization level approaching zero. The analogue difference applying a batching decision based on queueing delay minimization is bounded to 18.5% for a utilization level approaching zero. For an increasing utilization level, the difference between the minimal variance and the variance at batch sizes that minimize the queueing delay or the total time in system, decreases and asymptotically disappears.

Chapter 6 discusses one of the basic assumptions of the model. As was stated above in a queueing model an assumption has to be made on the arrival processes of the batches at the production facility. In Chapters 3-5 of the thesis, it is assumed that the batches arrive according to a Poisson process. Although this assumption is quite common and is reasonable in many cases, instances may occur where this assumption is less proper. In Chapter 6 the single item case is considered, where the batches arrive according to an Erlang distribution with the batch size as shape parameter. A well-performing closed form analytical approximation is derived for the total time in system that can be used easily to calculate the optimal batch size and the minimal lead time. The difference in outcomes between the models using the Erlang and the Poisson arrival pattern is huge, see also Tielemans and Kuik [12].

5 Discussion

Of course, some shortcomings of the models studied in the thesis can be identified. Future research is needed to investigate the influence of these limitations on the model outcomes and to determine the urge of model adaptations or extensions.

In a dynamic environment with rapidly changing demand patterns or product mixes, the applicability of stationary queueing models will be restricted. In literature a very small amount of studies can be found that consider a queueing system with a time-dependent arrival rate. However, these studies do not give (approximate) expressions for the time-varying queueing delay. So, simulation studies will be needed to check the performance of the stationary queueing models in situations with time-varying demand, for instance seasonal demand.

The results in Chapter 6 indicate, by studying a simple single item case, that the modelling of the job arrival process has a major impact on the model solutions. So it will be interesting to investigate also for the multi-item case the impact of different job arrival processes on model outcomes. To this end simulation studies can be executed, or the heavy traffic approximation for the waiting times in an GI/G/1 queueing systems can be used as a basis for an analytical study.

As already was mentioned above also the selection discipline that determines which job will be processed after the machine becomes idle, influences the average queueing delay. So also the relationship between the minimal average queueing delay and the different selection disciplines is an interesting research topic. Especially, it will be valuable to extend the study of Kekre [7], in which the impact of a look-ahead selection rule on the average minimal queueing delay is investigated for the homogeneous items case, to the case of heterogeneous items.

The models discussed so far take into account only a single capacity restriction, namely the total available production time. However, in practice, also restrictions on, for instance tool- or manpower availability can occur. These additional constraints certainly will erode the lead time performance. It will be interesting to get insight into the order of magnitude of this reduction in performance.

References

1) Axsäter, S. and J. Olhager. (1985). The Impact of Capacity Investments on Work-in-process and Inventories. *Engineering Costs and Production Economics* **9**, pp 119-124.
2) Cohen, M.A. and H.L. Lee. (1988). Strategic analysis of integrated production-distribution systems: Models and methods. *Operations research* **36** (2), pp. 216-228.
3) Jönsson, H. and E.A. Silver. (1985). Impact of Processing and Queueing Times on Order Quantities. *Material Flow* **2**, pp 221-230.
4) Karmarkar, U.S. (1987). Lotsizes, Lead times and In-process Inventories. *Management Science* **33** (3), pp. 409-419.
5) Karmarkar, U.S., S. Kekre and S. Kekre (1985). Lotsizing in Multi-item Multi- machine Job Shops. *IIE Transactions* **17** (3), 290-298.
6) Karmarkar, U.S., S. Kekre and S. Kekre (1992). Multi-item Batching Heuristics for Minimization of Queueing Delays. *European Journal of Operational Research* **58**, pp. 99-111.
7) Kekre, S. (1987). Performance of a Manufacturing Cell with Increased Product Mix. *IIE Transactions* **19** (3), pp. 326-339.
8) Kuik, R. and P.F.J. Tielemans. (1997). Setup Utilization as a Performance Indicator in Production Planning and Control. To appear in *The International Journal of Production Economics.*
9) Salomon, M. (1991). *Deterministic Lotsizing Models for Production Planning.* Springer-Verlag, Berlin.
10) Stalk, G. (1988). Time - The next source of competitive advantage. *Harvard Business Review* **66** (4), pp 41-51.
11) Tielemans, P.F.J. (1996). *Lead Time Performance in Manufacturing Systems.* Eburon, Delft.
12) Tielemans, P.F.J. and R. Kuik. (1996). An Exploration of Models that Minimize Lead Times through Batching of Arrived Orders. *European Journal of Operational Research* **95**, pp 374-389.

13) Zipkin, P.H. Models for Design and Control of Stochastic Multi-item Batch Production Systems. *Operations Research* **34** (1), pp 91-104.

Performance Analysis of Repairable Systems

Pieter Wartenhorst
McKinsey & Company
Amsterdam
E-mail: pieter_wartenhorst@mckinsey.com

1. Introduction

Almost any system or technical installation in our society is subject to failure of one or more of its components. Failures may severely affect system performance. Nowadays people become more and more convinced of the fact that the influence of failures on system performance is not just to be taken for granted. By adjusting system configuration, by introducing preventive maintenance schemes, and by improving component quality, major improvements on system performance can be achieved.

To assess the impact of various design proposals, or to choose the best from a number of proposals, systematic quantitative methods are needed. Since component failure behaviour is random in nature, stochastic mathematical models have been developed for this purpose.

This chapter contains a summary of the authors PhD thesis, which is a further contribution to the development and refinement of such models. The remainder of this chapter contains

- An example illustrating the importance of mathematical modelling,
- An description of the main drivers of the performance of repairable systems, and
- An overview of the results presented in the thesis.

2. Importance of Mathematical Modelling

The importance of mathematical modelling of repairable systems is illustrated by a simple example showing the impact of redundancy on the availability of a production system. Consider a single-machine system which has to provide service continuously. The machine is subject to failure, and the average lifetime of the machine is assumed to be equal to ten months. Repair of a broken machine takes two weeks (half a month) on average. Now, the fraction of time during which this single-machine system is not available to provide service (the unavailability of the system) is equal to 0.5 / (10 + 0.5) * 100 = 4.8 percent.

---

[1] This chapter contains a summary of the author's PhD thesis "Performance Analysis of Repairable Systems", 1993, Tilburg University, ISBN 90-9006121-5. The research has been carried out at CWI (Centre of Mathematics and Computer Science), Amsterdam (1988-1993), and at Koninklijke/Shell- Laboratorium, Amsterdam (1990-1991).

Compare this system with a so-called 1-out-of-2 cold standby system: a second machine is acquired, which is put into operation as soon as the first machine fails and is sent to the repair facility. So, the overall system stays 'up' at failure of the first machine. At repair completion, the first machine takes the standby position until the second machine fails and is sent to the repair facility, and so on. The system breaks down only when *both* machines are in a failed condition.

It is clear that the system with redundancy performs better than the system without redundancy. However, the remaining question is *how much better does the second system perform?* Assuming that life and repair times are exponentially distributed, further mathematical analysis shows that the average length of a system up-period is increased from 10 months to 210 months. The unavailability of the system is now equal to 0.5 / (210 + 0.5) * 100 = 0.2 percent. This implies a reduction of the unavailability of more than 95 percent!

Mathematical models are an excellent tool for comparing various design alternatives and their impact on system performance. This statements holds even in situations where it is difficult to obtain precise estimates of the distributions of life- and repairtimes. For instance, in our example, even if the average lifetime of a component deviates strongly from ten months, the model will still provide us with the insight that addition of a second machine enormously improves the performance of the system.

3. Drivers of Performance of Repairable Systems

A repairable system is defined as: 'a system which, after failure to perform at least one of its required functions, can be restored to performing all of its required functions by any method, including full system replacement'.

One or more performance criteria should be selected to determine *how well the system is functioning.* The system may be judged with respect to production throughput, effectiveness, reliability, availability, safety, or some general cost/reward function. The *reliability* of a system is defined as the probability that the system will function without failure during a certain specified period of time. The *availability* of a system is the probability that the system is functioning at a certain point in time (point availability), or the fraction of time during which the system is functioning over a certain time interval (interval availability; the system is allowed to be non-functioning once in a while).

System performance is mainly driven by:

- Failure mechanisms,
- System configuration, and
- Maintenance.

3.1. Failure mechanisms

During operation components may fail or, more generally, the system may enter a certain state of degradation. Failure mechanisms can be broadly grouped in overstress mechanisms

and wearout mechanisms, because failures are due to a complex set of interactions between:

- Stresses to which the system is exposed, either externally or internally, and
- Materials / elements of the system.

An obvious way to improve system performance is by replacing components by more reliable (more expensive) ones.

3.2. System configuration

System performance may be even more strongly improved by adjusting the configuration. In our example we have seen already the great impact of redundancy. Similar improvements can hardly be obtained by installation of better components, or by application of preventive maintenance policies (however sophisticated such policies may be).

Various configurations of the components of a system exist. The most familiar ones are: series, parallel redundancy, and standby redundancy. Combinations are possible as well.

Series configuration can only be used in a situation with highly reliable components: the entire system breaks down as soon as one of the individual components breaks down.

Parallel redundancy is often applied in process industry, where several machines perform the same task. At breakdown of one machine, the remaining machines will take over the failed machine's contribution, such that the entire production process will hardly ever be fully disturbed. Another example of parallel redundancy is provided by the engines of an airplane. If one engine fails, the airplane will still be able to safely reach the ground operated by its remaining engine(s).

Standby redundancy is often applied to systems which have a critical function to their environment. Such systems are not allowed to be in a down-state for a substantial period of time, because of safety or continuation reasons. One should think of power generators in a hospital, cooling-water pumps in a nuclear power-station, critical computer systems (e.g. in a space shuttle), and brakes of a high-speed train (TGV).

Systems with redundancy are much more reliable than systems without redundancy, because the system will only break down when its components are in a failed state all at the same time.

Note that the incorporation of redundancy should be considered mainly during the design phase of the system. It will be more difficult and expensive to consider installation of redundant components later on.

3.3. Maintenance

A maintenance policy determines parameters such as the intervals (measured in operational or calendar time) between system inspections, the age of the components at which they have to be replaced by new ones, and the critical value of some prognostic characteristic of the interior state of the system.

In the example the simplest type of maintenance has been applied, which is *corrective maintenance*: repair is carried out after failure of a component. The only direct way to improve system performance with respect to corrective maintenance is by adjusting the repair rate.

Instead of waiting for components to break down, one might use *preventive maintenance*: by replacing or revising components before failure, one might prevent a system from suddenly breaking down, thereby avoiding high corrective maintenance costs or long and expensive down-periods. Preventive maintenance of single components may be *time-based* (e.g. periodical), *age-based* (e.g. if component lifetime exceeds a certain threshold value), or *condition-based* (e.g. if too much vibration is observed). In some situations it is cost-effective to apply *group-maintenance*. For instance, at failure of a single light-bulb in a large building, this bulb is correctively replaced by a new one. However, since it is undesirable (economically or organizationally) to meet with too much corrective replacements, one might decide to replace all bulbs by new ones either if the number of corrective replacements in a fixed time period exceeds a certain threshold value, or at a regular time-base. Another example of group maintenance is found when an entire (sub-)system is replaced at first failure of one of its components. Here one might think of a gear-box of a crane. At failure of only a minor bearing, which takes quite some time to replace, one cannot afford to keep the crane out of operation for such a long time. Therefore the entire gear-box of the crane is replaced by a new or revised one. The gear-box is moved to a repair shop where the failed bearing is replaced (possibly together with some other components subject to wear) – not affecting the functioning of the crane.

Preventive maintenance is usually carried out during planned time intervals on a regular time base (e.g. during a yearly system shutdown: *campaign type maintenance*). Additionally, it is often applied during sudden breakdown of the system (*opportunity maintenance*).

Usually, for a given system configuration, one tries to choose the maintenance policy such that the system performance is optimized. However, there should not be a strict separation: during system design phase (when the configuration isdetermined) the future maintenance scheme should be taken into account already.

4. Overview of the Thesis

We have analyzed models with various assumptions on failure mechanisms, system configuration and maintenance policies.

4.1. Parallel Queueing Systems with Server Breakdown and Repair

In Chapter 2 we present a queueing model that can be used to study the influence of machine breakdown and limited repair capacity on the performance of a system that has to provide service continuously.We consider a system consisting of a number of stations, each serving its own stream of customers. The servers of the stations are subject to breakdown. Broken servers are repaired by a joint repair facility with a limited number of repairmen. With this model various design issues can be investigated such as the number of repairmen that is needed to maintain a pool of machines, or the number of machines that can be assigned to a certain crew of repairmen.

Our model is a two level combination of reliability theory and queueing theory. At level 1 the servers and the repairmen constitute an ordinary machine repair model (a well-known model in reliability theory). At level 2 customers are served by servers that are subject to breakdown. Interaction between broken servers at the repair facility (level 1) influences the behaviour of the queue lengths at the various stations (level 2).

An exact matrix-geometric solution is given for the steady-state distribution of the number of customers present at a particular station. The matrix-geometric method has great modelling flexibility and allows for several generalizations of the model. Since the matrix-geometric method may become rather time- and memory-consuming, stochastic decompositions are employed to obtain simple and accurate approximations.

From sensitivity analysis we conclude that it is important to take down-periods of the server explicitly into account, even when approximating the model. Furthermore, the queue length of a service station subject to server breakdown is heavily influenced by the speed at which customers pass this station, whereas the queue length of a station without server breakdown is not (customers move faster when both interarrival and service times are shorter).

4.2. A Two-Machine Repair Model with Variable Repair Rate

Chapter 3 considers the optimal control of the repair rate for a two-unit standby system with one repairman. The repairman works either at a fast or at a slow rate. In the literature, the direct control of the repair rate in machine repair models is considered by several authors. They all assume that the repair rate depends only on the number of broken units. We assume the repair rate to depend on the actual amount of work that is to be performed as well. The model is formulated as a semi-Markov decision process. From the corresponding optimality equations conclusions are derived on the structure of the optimal policy which minimizes long-run average costs. If there are no fixed costs associated with overall system breakdowns, then the optimal policy is a two- dimensional control limit rule (also called threshold policy). If fixed costs are incurred every time the system breaks down, then the optimal policy is not necessarily of control limit type. This is illustrated by an example where a four- region policy is the optimal one. Furthermore, we present several performance measures for this system controlled by a two-dimensional control limit rule.

4.3. A Two-Unit Standby System with Markovian Degrading Units

In Chapter 4, again we study a two-unit standby system with a single repair facility. However, there is much more detail in this model as compared to the model in Chapter 3. Instead of two simple units which are either functioning or non-functioning, we consider two Markovian degrading units: the units themselves can be considered as complex fault tolerant configurations consisting of many (not necessarily identical) components; under the absence of repair the condition of the working unit deteriorates according to a continuous time Markov process. Next to corrective repair of a unit (after failure) we have the option of preventive repair (before failure). Repair times depend on the actual state of the unit. Preventive repair on the working unit is carried out according to a control limit rule. For several variations of this model we study the distribution of the lengths of the up- and down-periods of the system, by applying a regenerative approach and by

using results from Markov decision theory. Such measures give much more information about the performance of a system, than average measures such as the long-run average availability. Apart from the cold standby structure the emphasis on transient analysis is a major point of difference with existing literature.

4.4. Bounds for the Interval Availability Distribution

Chapter 5 considers the numerical evaluation of the probability distribution of the interval availability of a two-state single-component system. This distribution is known to be given by an infinite summation of convolutions. In Chapter 5 we present bounds that can be used to truncate the infinite summation properly. We also present computational schemes for the case that both the lengths of up- and down-periods are distributed according to mixed Erlang distribution functions. The mixed Erlang distribution is not fully determined by its mean and variance. Therefore it is a useful distribution for sensitivity analysis. The interval availability distribution of a single-component system can be used to obtain approximations of the interval availability distribution of more general systems, for instance the ones that are studied in Chapters 3 and 4.

4.5. Transient Failure Behaviour of Repairable Systems

Chapter 6 contains the results of the work done during a six-month visit of the author to Koninklijke/Shell-Laboratorium, Amsterdam (KSLA). In this chapter we investigate the failure behaviour of repairable systems over a finite time interval, as well as the influence of various preventive maintenance policies. Due to a great difference in time scale we ignore repair times. The repairable system is modelled as a stochastic point process. Our approach is mainly based on techniques from renewal theory. The failure behaviour is studied by counting the number of failures and replacements that occur during finite time intervals. Since with each component failure or preventive replacement a certain amount of work is related, the techniques presented in this chapter are of direct use fort he analysis of maintenance workload.

The results of this chapter provide major advances in analyzing repairable systems, particularly by avoiding the usual steady state assumption and by incorporating the effects of preventive maintenance. The techniques presented in this chapter have been implemented in a PC-software package (EMMA) that has been developed at KSLA for the systematic assessment of minimum manning levels for existing and future oil-production installations.

Part V

Other contributions

OR-consultants at CQM: Make Optimization Work!

Dick den Hertog and Benjamin Jansen
CQM b.v., P.O. Box 414
5600 AK Eindhoven
The Netherlands
E-mail: denhertog@cqm.nl
E-mail: jansen@cqm.nl

1 Introduction

Make optimization work. That is the mission we stand for as OR-consultants at the Center for Quantitative Methods (CQM b.v.). After obtaining a PhD-degree in mathematics, and in the possession of the LNMB certificate, we both decided for an OR-job strongly emphasizing applications of OR. We did, and do, this in the belief that OR models and techniques are helpful in a lot of situations, for a large diversity of customers and organizations. This diversity, and the strong focus on OR, led us to CQM. In our contribution to this book we like to describe our life as OR-consultants at CQM (Section 2). We will give a sketch of the typical projects we are carrying out, where OR is really required to help solve the customer's problem (Section 3). Our theoretical academic background (including the LNMB courses) is indispensable and of great value for our consultancy practice (Section 4). Since several years, discussions have been going on the value of OR in business applications. We will give examples to show the great value worth of OR in practice! Important positive developments will be pointed out. It is now the challenge both for OR-practitioners and OR-academics to exploit these chances (Section 5). We will end with some words of thanks to LNMB.

2 Center for Quantitative Methods (CQM)

CQM, located in Eindhoven, is a consultancy bureau that provides services to industry and government in solving technical and organizational problems. In most cases quantitative models and methods from the disciplines statistics and OR form the base of the CQM contribution.

CQM was founded in 1979 as a consultancy bureau within Philips. Since 1993 CQM is independent of Philips by a management-buyout. At this moment CQM employs a team of approximately 40 professionals with a scientific background in statistics and OR. They consider the conversion of high-grade knowledge and experience in practical results as a challenge. CQM aims to be a top bureau in Europe in its own field and with its specific approach. During the past 18 years, CQM gained experience in over 1.700 projects. CQM's projects take place in research, development, production, distribution, service, finance and marketing.

3 Some of our typical OR-projects

The focus of the OR-group of CQM is on the following four, what we call, Product Market Combinations:

- vehicle routing and container management
- time-tabling
- design optimization
- strategic physical distribution.

In all these areas we try to focus on the OR-contribution as much as possible. Our mission is summarized in the slogan "Make Optimization work". We are experts in making optimization models and algorithms. However, it is not our prime objective to make a model that has nice theoretical properties, or for which we can design a polynomial time (approximation) algorithm. If possible, this is fine of course. However, the most important task we stand for is to develop a model that incorporates *all* the relevant practical aspects of the problem in such a way that a solution to the model is also meaningful in practice. Given this constraint, we try to make an algorithm that performs as good and as fast as possible. However, given the short time span most customers like their problem to be tackled it is often not possible to develop and implement the best algorithm we can think of. A trade off between quality and invested time has to be made each time.

"Make optimization work" also means that we use a diversity of techniques from optimization, without having one favorite method we solve all problems with. The knowledge of and experience in how to apply linear and nonlinear optimization, local search, tabu search, simulated annealing, genetic algorithms, neural networks, has evolved in our studies, customer projects, internal projects, keeping up to date with scientific literature, and contacts with universities and research institutes.

Of course in many projects the optimization part itself is not sufficient for a successful project, but we avoid carrying out work which does not fit with our OR-expertise. For such work other experts can be found, who can do it faster and better. To give some examples:

- in principle we will not develop (user) interfaces ourselves;
- we do not build database applications;
- we will not do the general logistics or management consultancy.

A consequence of this choice is that we often collaborate with third parties.

To give some examples of our projects:

- Optimization module in EVO Ritplan package. The Dutch 'Ondernemersorganisatie voor logistiek en transport' (EVO) is a branch organization for companies involved in transportation, with about 40000 members in The Netherlands. Some years ago, its IT-department developed an administrative package to support the planning and routing of vehicles. At a given moment EVO asked us to develop an optimization module, which was missing up till then. The module should be able to solve vehicle routing problems with a heterogeneous fleet, various capacity constraints, multiple time windows, and many other constraints. We could focus on the speed and efficiency of the optimization-algorithm, since the user interface was already developed. Moreover, and

not less important, EVO has a good insight in all the practical issues and constraints, and a wide network of prospective customers! In this way a very effective tool has been developed, which is used by many companies in the Netherlands.

- Customized vehicle routing planning software For a very large distributor with a special vehicle routing situation, a customized vehicle routing planning system has been developed in close cooperation with a software company. Again, we focused on the modeling and algorithms, the software company focused on the user interface, the database, the interface with existing databases, and so on. Moreover, the question was not whether we could build such a system, but whether we could build it in 6 weeks. The client was very satisfied with the result, and was surprised that we were able to develop such a system in a relatively short time.

- Automatic time-tabling In the same way as the EVO Ritplan package, we recently developed an optimization kernel for the school time-tabling software package Roosterfact, which is in use at about 350 secondary schools in the Netherlands. For these schools just an automatic time-tabling routine would have been of little value, for instance since links with administration-systems are required.

- Strategic distribution project together with a logistics consultancy bureau. Recently, a project has been carried out for a multi-national to improve its distribution layout and way-of-working. In this project we carried out the quantitative modeling work, in close cooperation with logistics consultants from another bureau, who looked at the qualitative and organizational aspects. This cooperation appeared to be extremely fruitful, since our expertises are complementary.

- Design Optimization projects In recent years many projects have been carried out in the area of Design Optimization, where we improve and speed up the design of new products and processes. The techniques underlying these applications are from nonlinear programming and statistics. Naturally, a cooperation between the statistics and OR departments of CQM takes place.

In all projects we focus on the OR-aspects of the problem as much as possible. This has several advantages.

- We are experts in OR, not in making interfaces or in logistics, to give some examples. Working with a multi-disciplinary team with experts from these diverse fields will result in optimal quality.

- After finishing the project it is clear for the client what our special OR-contribution is. In the future he can easier detect when our OR-assistance is necessary for him.

- By concentrating on the OR-contribution in a project, we can concentrate on following the new OR-developments.

- Cooperating with other consultancy firms or IT-companies often makes the acquisition process easier.

- It is easier to get highly qualified OR-employees, since the contents of the projects are real OR!

4 The advantages of a profound academic background

Before working as consultants at CQM, we both have carried out research on a rather theoretical subject ("Interior Point Methods"). Still we believe that this profound theoretical background is indispensable for our consultancy work. Not that we use the specific specialist knowledge we built up on the PhD-subject, but we use a lot of other things we learned during our PhD-period:

- an open mind for getting acquainted with and/or developing new ideas and techniques
- a broad knowledge of available models and methods
- a wide entrance in the literature, as well as a network in universities.

Although some people told us (and this is told hitherto) that "in practice you do not need all these difficult OR-methods, your common sense is enough", we discovered that the contrary is true! We can say that almost in each project these "difficult OR methods" were necessary, and were essential in the improvements we established. Of course this is also due to CQM's focusing on the OR-core ("make optimization work").

In this context also the LNMB courses were of great value for us. In this way also the OR-fundamentals for other OR-fields than the subject of the PhD-study were learned. Moreover, we got contacts with OR-researchers and OR-practitioners, which are very useful in our consultancy practice.

To say it in other words: we do not see our theoretical research in the past and our current practical consultancy work as two different and separated worlds. Of course there are differences, but the one cannot breathe without the other.

Another advantage of a profound OR background is that because of having an easier entrance in the literature and a good overview of the OR field and many contacts with OR-researchers, we are able to find the right track of solution at an early stage, and avoid developing (parts of) models and methods which have already been developed.

However, it is true that some other extra capabilities are necessary for applying OR-consultancy successfully for customers. Communicative skills and a feeling for practical solutions are also essential. But, in our view starting with a profound analytical OR background is more important than with a good education in consultancy. If the basic attitude is OK, the consultancy capabilities can be acquired by "learning by doing" and by courses. However, learning the OR-theory and methods while working as a consultant is very difficult.

5 Challenges for the OR community

In our daily consultancy practice we observe that OR becomes more and more important. In our view the most important reason for this are the developments in Information Technology (IT). Both on the hardware as on the software side rapid developments have been going on in the past 10 years, and is still in progress. This has both direct and indirect implications for OR.

The development of information systems in many companies nowadays has several effects that are important for OR. Since a lot of information is available in such systems, people are asking for added value. They want more than just the administrative, controlling and accounting functionality of the information system: data should become information, and information should be used to optimize processes. Optimization, data warehousing, and decision support are becoming more and more important. On the other hand, an argument

often used against OR that its models cannot be fed with realistic data, has become weaker due to the availability of huge amounts of data. This also makes it easier to use OR in the decision support on tactical and strategic level, as one of the conditions for OR-success (data) is being fulfilled. To give an example: OR is more and more used for determining the network layout and the strategic physical distribution. The need for a quantitative approach is recognized by customers as well as management consultants. The OR contribution is really worthwhile since we can use detailed information on orderline level, instead of aggregate estimates.

Developers of standard information systems also recognize the need to improve their systems by incorporating functionalities that use the data. Planning modules, routing modules, rostering functions are being included. An important example are the developments in the Enterprise Resource Planning software, playing a very important role in both large and small size companies. The producers (especially the larger ones as SAP, Oracle and Baan) now see the value of OR, and try to implement OR functionalities in their software. These are not just add-ons that are nice to have, but fundamental and essential to have. We think that this development will give a great impulse to OR-practitioners and researchers.

Evidently, faster personal computers having more internal memory and disk space have opened the way to solving larger OR-models much faster than before. For instance, solving realistic vehicle routing problems with many practical constraints (multiple time windows, heterogeneous fleet, various capacity constraints, etc.) can be done within minutes (example: 1200 clients, 75 vehicles in 20 minutes). As a second example, scenario models can be fed with realistic data up to the level of orderlines. That this may require over 100MB internal memory makes clear that such models would be out of the question even a few years ago. Here we think that it is quite important that *personal* computers have these capabilities nowadays. Many people have such a computer on their desk, and are acquainted using it. This makes the acceptance of OR models much better, than when it has to be done on a 'black box' as a main frame. Using computers every day have made people more accustomed to facts, numbers and statistics, which leads to a positive position towards quantitative, fact-based OR techniques. For developers of OR-models the introduction of user friendly optimization packages (e.g. AIMMS, AMPL) has made their job easier. The development time in OR-projects is shorter, more possibilities are available, user interfaces can be built quite easily. For users of OR-models, working with applications in one of these packages is very convenient. Also, in spreadsheet packages like EXCEL and Quattro Pro OR-algorithms have been included, and can sometimes be of much help.

Playing a role in this lively environment is challenging for this. It is the task of the OR-community to convince (potential) customers that a quantitative approach adds value to business just as a qualitative approach. As well, it is a challenge to make automated planning and optimization modules not just nice to have, but to make them that useful that they become the core of planning systems.

A good Public Relation for OR is especially now very important, since there are a lot of opportunities. We hope that in the near future more people (especially the managers) will know the capabilities of OR, when to use OR or when to call for an OR-consultant, just as now they know where and when to call for a management or a logistics consultant. This is also the drive of the Nederlandse Genootschap voor Besliskunde (Dutch Society of Operations Research) for their 'PRomotOR' activities. Examples of their activities are: publishing in newspapers, non-OR journals, magazines, and handbooks, etc. In this way the NGB tries to promote OR in the Netherlands.

Also, we like to get more people involved: people in industry, researchers and students at

universities, but also very importantly students at secondary schools. For the latter, basics of OR are incorporated in their mathematics curricula, but we doubt that they are made clear that and how these techniques can be used many diverse practical problems. Recently, two price contests have been out in the Netherlands: material routing by NS, CWI and traveling salesman in New York by CMG, TUE and De Telegraaf. More of those initiatives are highly welcome.

6 To LNMB

To end our contribution we like to congratulate the LNMB with its 10th anniversary. We thank LNMB for its contribution to our OR-education through courses as well as workshops. For the first author an LNMB workshop on the role of OR in practice was a great stimulus to go for applying OR after finishing his PhD thesis. We hope LNMB will also be playing an active and constructive role in coming years and continue its activities for many years.

A Short Survey on Semidefinite Programming*

E. de Klerk, C. Roos and T. Terlaky

Faculty of Technical Mathematics and Informatics,

Delft University of Technology,

E-mail: e.deklerk@twi.tudelft.nl, c.roos@twi.tudelft.nl,

t.terlaky@twi.tudelft.nl

Abstract

Semidefinite programming (SDP) is one of the fastest developing branches of mathematical programming. The reason is twofold: efficient solution algorithms for SDP have come to light in the past few years, and SDP finds applications in combinatorial optimization and engineering. In this short survey we show how SDP duality theory can be used to prove classical results, and review the development of interior point algorithms for SDP.

1 Introduction

One could easily be led to believe that the field of semidefinite programming (SDP) originated in this decade. A glance at a bibliography of SDP papers indeed indicates an explosion of research effort, starting around 1991. A closer look reveals that interest in this class of problems is somewhat older, and dates back to the 1960's (see e.g. [6]). A paper on SDP from 1981 is descriptively named *Linear Programming with Matrix Variables* [11], and this apt title may be the best way to introduce the problem.

The goal is to minimize the inner product

$$\langle C, X \rangle := \mathbf{Tr}\,(CX),$$

of two $n \times n$ symmetric matrices, a constant matrix C and a variable matrix X, subject to a set of constraints, where '**Tr** ' denotes the trace (sum of diagonal elements) of a matrix.[1] The first of the constraints are linear:

$$\langle A_i, X \rangle = b_i, \quad i = 1, \ldots, m,$$

*Some authors prefer to use the term 'optimization' instead of 'programming'.

[1]This inner product corresponds to the familiar Euclidean inner product of two vectors – if the columns of the two matrices C and X are stacked to form vectors $\mathbf{vec}(X)$ and $\mathbf{vec}(C)$, then $\mathbf{vec}(C)^T\mathbf{vec}(X) = \mathbf{Tr}\,(CX)$.

where the A_i's are given symmetrix matrices, and the b_i's given scalars. Up to this point, the stated problem is merely a linear programming (LP) problem with the entries of X as variables. We now add the convex, nonlinear constraint that X must be symmetrix positive semidefinite, denoted by $X \succeq 0$.[2] The convexity follows from the convexity of the cone of positive semidefinite matrices. The problem under consideration is therefore

$$\min_X \{\mathbf{Tr}\,(CX) \,:\, \mathbf{Tr}\,(A_iX) = b_i\ (i = 1, \ldots, m),\ X \succeq 0\}$$

The Lagrangian dual of our problem takes the form

$$\max_{y,S} \left\{ b^T y \,:\, \sum_{i=1}^{m} y_i A_i + S = C,\ S \succeq 0,\ y \in \mathbb{R}^m \right\}.$$

The *weak duality theorem* therefore implies that $\mathbf{Tr}\,(CX) - b^T y \geq 0$ for all feasible solutions. Equality holds at optimality if both problems have feasible sets with nonempty interiors (*strong duality*) (see e.g. [5]). The duality theory for SDP is weaker than that of LP. Difficulties associated with general convex programming can occur if the strict feasibility condition is not met; thus a problem can be solvable although its Lagrangian dual is infeasible, or both problems can be solvable but with a positive duality gap at optimality.

SDP problems are of interest for a number of reasons, including

- SDP contains important classes of problems as special cases;
- important applications exist in combinatorial optimization and engineering;
- efficient solution strategies have emerged in the past few years (explaining the resurgence in research interest).

Each of these considerations will be discussed briefly. The discussion will be such as to limit the overlap with previous surveys. We will mainly focus on recent developments of interior point methods for SDP, and on the usefulness of SDP duality theory as a technique of proof.

An excellent survey by Vandenberghe and Boyd [58] deals with basic theory, diverse applications, and potential reduction algorithms (up to 1995). Two more recent surveys which focus more on applications of SDP in combinatorial optimization are by Alizadeh [2] and Ramana and Pardalos [48]. The former also deals with interior point methodology, whilst the latter contains surveys of geometric properties of the SDP feasible set (so-called spectrahedra), as well as complexity and duality theory.

2 Special cases of SDP

If the matrix X is restricted to be diagonal, then the requirement $X \succeq 0$ reduces to the requirement that the diagonal elements of X must be nonnegative. In other words, we once

[2] By definition, for symmetric X one has $X \succeq 0$ if $z^T X z \geq 0$, $\forall z \in \mathbb{R}^n$, or equivalently, if all eigenvalues of X are nonnegative.

again have an LP problem. Optimization problems with convex quadratic constraints are likewise special cases of SDP.[3] This follows from the well-known *Shur complement* trick: if

$$X = \begin{bmatrix} A & B \\ B^T & C \end{bmatrix}$$

for an invertible A, then the matrix

$$S_{hur} := C - B^T A^{-1} B$$

is called the Shur complement of A in X. One has

$$\begin{aligned} X \succ 0 \quad &\text{if and only if} \quad A \succ 0 \text{ and } S_{hur} \succ 0 \\ \text{If } A \succ 0, \text{ then } X \succeq 0 \quad &\text{if and only if} \quad S_{hur} \succeq 0. \end{aligned}$$

It follows that we can represent the quadratic constraint

$$(Az + b)^T (Az + b) - (c^T z + d) \leq 0, \quad z \in \mathbb{R}^n$$

by the semidefinite constraint

$$\begin{bmatrix} I & Az + b \\ (Az + b)^T & c^T z + d \end{bmatrix} \succeq 0.$$

In the same way, we can represent the *second order cone*, $\|x\|^2 \leq (\sum_{i=1}^n x_i)^2$, by

$$\begin{bmatrix} (\sum_{i=1}^n x_i)\, I & x \\ x^T & \sum_{i=1}^n x_i \end{bmatrix} \succeq 0.$$

Another nonlinear example which arises frequently is

$$\min \left\{ \frac{(c^T x)^2}{d^T x} \; : \; Ax \geq b \right\},$$

where it is known that $d^T x > 0$ if $Ax \geq b$. An equivalent SDP problem is:[4]

$$\min \left\{ t \; : \; \begin{bmatrix} t & c^T x & 0 \\ c^T x & d^T x & 0 \\ 0 & 0 & \operatorname{diag}(Ax - b) \end{bmatrix} \succeq 0 \right\}.$$

Several problems involving matrix norm or eigenvalue minimization may be stated as SDP's. An extensive list of such problems may be found in [58]. A simple example is the classical problem of finding the largest eigenvalue $\lambda_{\max}(A)$ of a symmetric matrix A. The key observation here is that $t \geq \lambda_{\max}(A)$ if and only if $tI - A \succeq 0$. The SDP problem therefore becomes

$$\min \{ t \; : \; tI - A \succeq 0, \quad t \in \mathbb{R} \}.$$

An SDP algorithm for this problem is described in [26].

[3] This includes the well-known convex quadratic programming (QP) problem.

[4] We use the notation 'diag' as follows: for a matrix X, $\operatorname{diag}(X)$ is the vector obtained by extracting the diagonal of X; for a vector x, $\operatorname{diag}(x)$ is the diagonal matrix with the coordinates of x as diagonal elements.

3 Applications in combinatorial optimization

General quadratic optimization problems allow SDP relaxations. The key observation is that

$$x^T Q x = \mathbf{Tr}\,(Q x x^T),$$

for a given matrix Q and vector x. The rank one matrix $X = xx^T$ is positive semidefinite. We can therefore relax the condition $X := xx^T$ to $X \succeq 0$. This relaxation is originally due to Shor [53].

Combinatorial optimization problems can in turn be written as quadratic optimization problems. The condition $x_i \in \{-1, 1\}$ is equivalent to $x_i^2 = 1$, for example.

Lovász and Schrijver [35] considered the generic combinatorial problem

$$q^{\max} = \max\left\{x^T Q x \;:\; x_i \in \{-1,1\}\ (\forall i)\right\} \tag{1}$$

and suggested the relaxation

$$\bar{q} = \max\left\{\mathbf{Tr}\,(QX) \;:\; \mathrm{diag}\,(X) = e,\ X \succeq 0\right\}. \tag{2}$$

For this general relaxation Nesterov [42] recently proved that

$$\bar{q} - \underline{q} \geq q^{\max} - q^{\min} \geq \frac{4-\pi}{\pi}\left(\bar{q} - \underline{q}\right)$$

where $(q^{\min}, q^{\max})$ is the range of feasible objective values in (1), and $(\underline{q}, \bar{q})$ is the range of feasible values in the relaxation problem (2). Moreover, a random feasible solution x to (1) can be computed from the solution to the relaxation. The expected objective value of x, say $E(x)$, satisfies[5]

$$\frac{q^{\max} - E(x)}{q^{\max} - q^{\min}} < \frac{4}{7}.$$

For specific problems this bound can be improved. The showcase example is the *maximal cut problem*, i.e. the problem of finding a cut of maximal weight through a graph with weighted edges. In a pioneering article, Goemans and Williamson [21] proved that $\bar{q} \leq 1.14 q^{\max}$ in this case. They moreover devised a randomized algorithm which produces a cut with expected value greater than $0.878 q^{\max}$. Similar improvements were also reported in [21] for satisfiability problems.

The SDP relaxations are not always useful, though. Cases where the SDP relaxation is no stronger than the usual LP relaxation are reviewed in [48].

SDP offers more than just a numerical tool to generate lower and upper bounds on optimal values. It also provides a technique of proof via duality theory. We consider the classical *sandwich theorem*, and give a proof (which is new to the best of our knowledge) using strong duality theory. The theorem relates three characterizing numbers of a graph: the colouring number[6] $\chi(G)$, the maximal clique number[7] $\omega(G)$, and the Lovász number $\theta(G)$, which will be defined presently.

[5] The same bounds were obtained by Ye [60] for the 'box-constrained' problem where $x_i \in \{-1, 1\}$ is replaced by $-1 \leq x_i \leq 1$ in problem (1).

[6] Number of colours needed to colour all vertices so that no two adjacent vertices share the same colour.

[7] The cardinality of the maximal clique (connected subgraph).

For a graph $G = (V, E)$, a maximal clique is a subset $C \subset V$ with

$$\forall i, j \in C (i \neq j) \ : \{i, j\} \in E,$$

such that $|C|$ is maximal. The Lovász number $\theta(G)$ can be defined[8] as the optimal value of the SDP relaxation (see [34, 23]):

$$\theta(G) := \max \mathbf{Tr} \left(ee^T X\right) = e^T X e \tag{3}$$

subject to

$$\left.\begin{array}{rcl} X_{ij} & = & 0, \ \{i, j\} \notin E \ (i \neq j) \\ \mathbf{Tr}\,(X) & = & 1 \\ X & \succeq & 0. \end{array}\right\} \tag{4}$$

The sandwich theorem states the following.

Theorem 3.1 (Lovász's Sandwich Theorem) *For any graph $G = (V, E)$ one has*

$$\omega(G) \leq \theta(G) \leq \chi(G).$$

Proof:

In order to prove the first inequality of the theorem, let x_C denote a 0-1 vector which defines a clique C of size k in G, i.e:

$$(x_C)_i = \begin{cases} 1 & \text{if } i \in C \\ 0 & \text{otherwise.} \end{cases}$$

It is easy to check that the rank one matrix

$$X := \frac{1}{k} x_C x_C^T$$

is feasible in (4) with objective value

$$e^T X e = \frac{1}{k} \left(e^T x_C\right)^2 = \frac{k^2}{k} = k.$$

We therefore have $\omega(G) \leq \theta(G)$, which is the first part of the sandwich theorem. The second part is to prove $\theta(G) \leq \chi(G)$. To this end, we write down the Lagrangian dual of the SDP relaxation (4) to obtain

$$\theta(G) = \min \lambda \tag{5}$$

subject to

$$\left.\begin{array}{rcl} Y + ee^T & \preceq & \lambda I \\ Y_{ij} & = & 0, \ \{i, j\} \in E \ (i \neq j) \\ Y_{ii} & = & 0, \ i \in V. \end{array}\right\} \tag{6}$$

[8]Strictly speaking, the definition given here is of the Lovász number of the complement of G (nodes in the complement of G are connected if and only if they are not connected in G).

Given a colouring of G with k colours, we must constuct a feasible solution for (6) with $\lambda \leq k$. Such a colouring defines a partition $V = \cup_{i=1}^{k} C_i$ where the C_i's are subsets of nodes sharing the same colour. In other words, the C_i's must be disjoint stable sets (co-cliques). Now let $\gamma_i = |C_i|$ and define

$$M_i := k\left(I_{\gamma_i} - J_{\gamma_i}\right), \quad i = 1, \ldots, k,$$

where I_{γ_i} is the $(\gamma_i \times \gamma_i)$ identity matrix, and J_{γ_i} the all-one matrix of the same size.

We will show that the block diagonal matrix

$$Y = \begin{pmatrix} M_1 & 0 & \ldots & 0 \\ 0 & M_2 & \ldots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & \ldots & M_k \end{pmatrix} \tag{7}$$

is feasible in (6) if $\lambda = k$. By construction, Y satisfies the last two constraints in (6). We must still show that $Y + ee^T \preceq kI$, i.e. the largest eigenvalue of $Y + ee^T$ must be at most k.

The Raleigh-Ritz theorem states that for any symmetric matrix A, one has:

$$\lambda_{\max}(A) = \max\left\{x^T A x \ : \ \|x\| = 1\right\}. \tag{8}$$

It follows that the maximal eigenvalue of Y is given by

$$\lambda_{\max}(Y) = \max\left\{\sum_{i=1}^{k} \alpha_i \lambda_{\max}(M_i),\ \sum_{i=1}^{k} \alpha_i = 1,\ \alpha_i \geq 0\ (\forall i)\right\}. \tag{9}$$

Moreover one has $\lambda_{\max}(M_i) = k$, so that (9) yields $\lambda_{\max}(Y) = k$. The eigenvector corresponding to k is orthogonal to the all-one vector e. To see this, note that $Yx = \lambda x$ implies

$$-k(\gamma_i - 1) \sum_{j \in C_i} x_j = \lambda \sum_{j \in C_i} x_j, \quad i = 1, \ldots, k,$$

so that $\sum_{j \in C_i} x_j = 0$ $(i = 1, \ldots, k)$ if $\lambda > 0$. In particular, $e^T x = 0$ from which it follows that k is also an eigenvalue of $Y + ee^T$. Assuming that k is not the largest eigenvalue of $Y + ee^T$, then the largest eigenvalue must have an eigenspace orthogonal to the eigenspace of k. The orthogonal complement of the eigenspace of k is spanned by the vectors

$$\left(x_{C_i}\right)_j := \begin{cases} 1 & \text{if } j \in C_i \\ 0 & \text{otherwise,} \end{cases}$$

where $i = 1, \ldots, k$. The maximal eigenvalue of $Y + ee^T$ can therefore be computed from (8):

$$\begin{aligned} \lambda_{\max}\left(Y + ee^T\right) &= \max_x \left\{x^T\left(Y + ee^T\right)x \ : \ x \in \operatorname{span}\left\{x_{C_1}, \ldots, x_{C_k}\right\},\ \|x\| = 1\right\} \\ &= \max_\alpha \left\{x^T Y x + \left(e^T x\right)^2 \ : \ x = \sum_{i=1}^{k} \alpha_i x_{C_i},\ \sum_{i=1}^{k} \gamma_i \alpha_i^2 = 1\right\} \end{aligned}$$

Substituting the expression for x, and using the construction of Y simplifies this to

$$\begin{aligned}\lambda_{\max}\left(Y+ee^T\right) &= \max_{\alpha}\left\{-k\sum_{i=1}^{k}\alpha_i^2\left(\gamma_i^2-\gamma_i\right)+\left(\sum_{i=1}^{k}\gamma_i\alpha_i\right)^2 \;:\; \sum_{i=1}^{k}\gamma_i\alpha_i^2=1\right\}\\ &= k+\max_{\alpha}\left\{-k\sum_{i=1}^{k}(\alpha_i\gamma_i)^2+\left(\sum_{i=1}^{k}\gamma_i\alpha_i\right)^2 \;:\; \sum_{i=1}^{k}\gamma_i\alpha_i^2=1\right\}.\end{aligned}$$

The expression in brackets is nonpositive, since it is of the form

$$-kz^Tz+(e^Tz)^2\leq -kz^Tz+(\|e\|\|z\|)^2=-kz^Tz+k\|z\|^2=0,$$

where $z_i=\alpha_i\gamma_i$, $(i=1,\ldots,k)$. This leads to the contradiction $\lambda_{\max}\left(Y+ee^T\right)\leq k$.

We conclude that $\lambda_{\max}\left(Y+ee^T\right)=k$, as required. □

Moreover, we have given a proof of the equivalence of two different definitions of $\theta(G)$ via (3) and (5).[9]

4 Engineering applications

The richest field of application of SDP is currently *system and control theory*. The standard reference for these problems is Boyd et al. [10]. Introductory examples are given in [58] and [45].

An application which receives less attention is *structural design*, where the best known SDP problem involves optimal truss[10] design. Two variants are:

1. minimize the weight of the structure such that its fundamental frequency remains above a critical value;
2. minimize the worst-case compliance ('stored energy') of the truss given a set of forces which the structure has to withstand.

The second of these problems allows another nice application of SDP duality theory. The problem may be stated as

Displacement formulation

$$\min_{t,x_1,\ldots,x_k}\max_j x_j^Tf_j,\quad j=1,\ldots,k$$

subject to

$$\begin{aligned}\left(\sum_{i=1}^{m}t_ib_ib_i^T\right)x_j &= f_j,\quad j=1,\ldots,k\\ \sum_{i=1}^{m}t_i &= V,\quad t\geq 0,\end{aligned}$$

[9] These and other equivalent definitions of $\theta(G)$ are discussed in [23].

[10] A truss is here defined as a structure of bars which connect a fixed ground structure of nodes. The design is fixed once the sizes of the bars have been decided.

where the t_i's are the bar volumes (design variables), and the f_j's are the set of forces which the truss has to withstand. The displacement of the nodes subject to force f_j is given by the vector x_j. The fixed vectors b_i depend only on the layout of the nodes and on the material properties (Young moduli) of the bars. The first constraint requires equilibrium of the structure and the second fixes its total volume. The objective is to minimize the worst-case compliance.

The name 'displacement' formulation stems from the displacement variables x_j. From engineering considerations, the problem may also be stated by using the forces in the bars as variables.

Bar forces formulation

$$\min_{\beta,t} \max_{j=1,\ldots,k} \sum_{\substack{i=1 \\ t_i>0}}^{m} \frac{\beta_{ij}^2}{t_i}$$

subject to

$$\begin{aligned} \sum_{i=1}^{m} t_i &= V \\ f_j &= \sum_{i=1}^{m} \beta_{ij} b_i, \quad j = 1,\ldots,k \\ t_i &\geq 0, \quad i = 1,\ldots,m \\ \beta_{ij} = 0 \quad \text{if } t_i &= 0, \quad i = 1,\ldots,m, \quad j = 1,\ldots,k, \end{aligned}$$

where β_{ij} is proportional to the reaction force in bar i due to f_j. The second constraint simply requires static equilibrium, i.e. a 'balance of forces'.

From a purely mathematical point of view it is far from obvious that the two formulations are equivalent. This equivalence can be shown using SDP duality. We will sketch the proof here. Using the Shur complement trick, the displacement formulation can be written as an SDP problem (for details, see [12]).

SDP reformulation of the displacement formulation

$$\min \tau$$

subject to

$$\begin{aligned} \begin{bmatrix} \tau & f_j^T \\ f_j & \left(\sum_{i=1}^{m} t_i b_i b_i^T\right) \end{bmatrix} &\succeq 0, \quad j = 1,\ldots,k \\ \sum_{i=1}^{m} t_i &= V, \quad t \geq 0. \end{aligned}$$

The equivalence proof is now done in three steps:

1: write down the dual of the SDP reformulation and simplify it;

2: obtain the dual of the resulting problem from Step 1;

3: reduce the problem obtained in Step 2 to the 'bar forces' formulation.

This sequence of steps is described detail in [41]. A survey of these and related formulations is given in [12], with emphasis on SDP formulations. Other structural design problems which may be formulated as SDP's include sandwich plate design [8], optimization of variable thickness sheets [51], and minimal compliance design with optimized materials [49]. Good reviews of interior point methods in truss topology design are Bendsøe et al. [9], and Jarre et al. [28] (see also [7]).

Other engineering applications of SDP include: VLSI transistor sizing, pattern recognition using ellipsoids, and logarithmic Chebychev approximation (see [58]).

5 Efficient solution strategies

Bearing the links between LP and SDP in mind, it may come as little surprise that interior point algorithms for LP have been successfully extended to SDP.

The field of interior point methods for LP more or less started with the famous paper by Karmarkar [30] in 1984, and in the following decade more that a thousand papers appeared on this topic. Some recent review papers include [20] and [26]. Several new books on the subject have also appeared recently, including [50] and [59].

The first extension of interior point algorithms from LP to SDP was by Nesterov and Nemirovskii [43], and independently by Alizadeh [1] in 1991. Nesterov and Nemirovskii actually considered a more general class of convex optimization problems, where the nonlinearity is 'banished' to a convex cone, like $X \succeq 0$. They show that such conic optimization problems can be solved by sequential minimization techniques, where the conic constraint is discarded and a barrier term is added to the objective. Suitable barriers are called *self-concordant.* These barriers go to infinity as the boundary of the cone is approached, an can be minimized efficiently by Newton's method.[11] The function

$$f_{bar}(X) = -\log\det(X)$$

is such a barrier for the cone of semidefinite matrices. Using this barrier, several classes of algorithms may be formulated which have polynomial worst-case iteration bounds for the computation of ϵ-optimal solutions.

5.1 Logarithmic barrier methods

Primal log-barrier methods use Newton's method to solve a sequence of problems of the form

$$\min_X \{\mathbf{Tr}\,(CX) - \mu\log\det(X) \;:\; \mathbf{Tr}\,(A_iX) = b_i\;(i = 1,\ldots,m)\}$$

where the parameter μ is sequentially decreased to zero. Such algorithms were analysed by Faybusovich in [17, 18] and later by other authors in [24] and [4]. Note that the condition $X \succeq 0$ has been replaced by adding a 'barrier term' to the objective.[12] The condition $X \succeq 0$

[11] The definition of self-concordant barriers is omitted here; a well-written introductory text is [27].

[12] This idea actually dates back to the 1960's and the work of Fiacco and McCormick [19]; the implications for complexity theory only became clear almost three decades later.

is maintained by controlling the Newton process carefully – large decreases of μ necessitate damped Newton steps (see e.g. [4]), while small updates allow full Newton steps (see e.g. [24]).

Following the trend in LP, so-called *primal-dual* methods soon became more popular. These methods minimize the duality gap

$$\mathbf{Tr}\,(CX) - b^T y = \mathbf{Tr}\,(XS)$$

and employ the combined primal-dual barrier function

$$f_{pd} := -\left(\log\det(X) + \log\det(S)\right) = -\log\det(XS).$$

This means that a sequence of problems of the following form are solved

$$\min_{X,y,S}\left\{\mathbf{Tr}\,(XS) - \mu\log\det(XS) \;:\; \mathbf{Tr}\,(A_i X) = b_i\;(i=1,\ldots,m),\; \sum_{i=1}^{m} y_i A_i + S = C\right\}. \quad (10)$$

The first order optimality conditions for (10) are

$$\left.\begin{array}{rcll} \mathbf{Tr}\,(A_i X) &=& b_i, & i=1,\ldots,m \\ \sum_{i=1}^m y_i A_i + S &=& C & \\ XS &=& \mu I & \\ X, S &\succ& 0. & \end{array}\right\} \quad (11)$$

This system has a unique positive definite solution pair, denoted by $X(\mu) \succ 0$ and $S(\mu) \succ 0$.[13] Primal-dual log-barrier methods solve the system (11) approximately, followed by a reduction in μ. Ideally, the goal is to obtain primal and dual steps ΔX and ΔS, respectively, which satisfy $X + \Delta X \succeq 0$, $S + \Delta S \succeq 0$ and

$$\begin{array}{rcll} \mathbf{Tr}\,(A_i \Delta X) &=& 0, & i=1,\ldots,m \\ \sum_{i=1}^m \Delta y_i A_i + \Delta S &=& 0 & \\ (X+\Delta X)(S+\Delta S) &=& \mu I. & \end{array} \quad (12)$$

The last equation is nonlinear, and primal-dual methods differ with regard to how it is linearized. Moreover, care must be taken that the solution matrices ΔX and ΔS are symmetrical. Zhang [61] suggested to replace the nonlinear equation by

$$L\left(\Delta X S + X \Delta S\right) L^{-1} + \left[L\left(\Delta X S + X \Delta S\right) L^{-1}\right]^T = 2\mu I - L\left(XS\right) L^{-1} + \left[L(XS)L^{-1}\right]^T,$$

where the matrix L determines the symmetrization strategy. Some popular choices for L are listed in Table 1. The proof of the existence and uniqueness of each of the resulting search directions was done by Shidah et al. in [52].[14] Other properties (such as scale-invariance) are compared by Todd et al. in [56].

[13] These solutions give a parametric representation of a smooth curve, called the *central path*, which tends to the *analytic center* of the primal-dual optimal sets as $\mu \to 0$. This was proved by Goldfarb and Scheinberg in [22].

[14] For $L = I$ uniqueness is not always guaranteed; a sufficient condition for uniqueness is $XS + SX \succeq 0$.

L	Reference
$\left[X^{\frac{1}{2}}\left(X^{\frac{1}{2}}SX^{\frac{1}{2}}\right)^{-\frac{1}{2}}X^{\frac{1}{2}}\right]^{\frac{1}{2}}$	Nesterov and Todd [44];
$X^{-\frac{1}{2}}$	Monteiro [38], Kojima et al. [33];
$S^{\frac{1}{2}}$	Monteiro [38], Helmberg et al. [25], Kojima et al. [33];
I	Alizadeh, Haeberley and Overton [3];

Table 1: Choices for the linearization matrix L.

The conspicious entry $L = \left[X^{\frac{1}{2}}\left(X^{\frac{1}{2}}SX^{\frac{1}{2}}\right)^{-\frac{1}{2}}X^{\frac{1}{2}}\right]^{\frac{1}{2}}$ in Table 1 warrants some comment. Nesterov and Todd [44] showed[15] that for each pair $X \succ 0$, $S \succ 0$ there exists a matrix D such that

$$f''_{bar}(D)X = S.$$

It can be shown that $f''_{bar}(D)$ is the linear operator which satisfies $f''_{bar}(D) : X \mapsto D^{-1}XD^{-1}$. It follows that $X = DSD$, from which it easily follows that $D = L^2$. In this way we obtain the *symmetric primal-dual scaling* $L^{-1}XL^{-1} = LSL$. This symmetry explains the usefulness of D in symmetrization.

Algorithms differ in how μ is updated, and how the symmetrized equations are solved. Methods with use large reductions of μ followed by several damped Newton steps are called long step (or large update) methods. These are analysed in [29], [38], and [55].

Methods which use dynamic updates of μ include the popular *predictor-corrector* methods. References include [3, 31, 32, 46, 54]. Other dynamic μ-updates are described in [14]. Superlinear convergence properties of predictor-corrector schemes are studied in [31, 37].

5.2 Primal-dual potential reduction methods

These algorithms are based on the potential function

$$\phi(X, S) = (n + \nu\sqrt{n})\mathbf{Tr}\,(XS) - \log\det(XS) - n\log n,$$

where $\nu \geq 1$. In order to obtain a polynomial complexity bound it is sufficient to show that ϕ can be reduced by an absolute constant at each iteration [57]. A survey of algorithms which achieve such a reduction is given in [58].

[15]This result was proved in the more general setting of optimization problems where the variable is restricted to a self-dual cone which allows a special type of self-concordant barrier, namely self-scaled barriers. The interested reader is referred to [44].

5.3 Affine–scaling methods

The primal affine–scaling direction for SDP minimizes the primal objective over an ellipsoid which is inscribed in the primal feasible region. Surprisingly, Muramatsu [39] has shown that an algorithm using this search direction may converge to a non-optimal point, regardless of which step length is used. This is in sharp contrast to the LP case, and shows that extension of algorithms from LP to SDP cannot always be taken for granted.

Two primal-dual variants of the affine scaling methods were extended by De Klerk et al. in [15] from LP to SDP. These algorithms minimize the duality gap over ellipsoids in the scaled primal-dual space, where the matrix $L = D^{\frac{1}{2}}$ is used for the scaling. The primal-dual method fails if either of the scalings $L = X^{\frac{1}{2}}$ or $L = S^{\frac{1}{2}}$ from Table 1 is used [40].

5.4 Infeasible start methods

Several infeasible start algorithms have been suggested. A review of traditional big-M initialization strategies may be found in [58]. One of the first infeasible-start predictor-corrector algorithms was by Potra and Sheng [46]. Other references include [31, 37].

The idea of embedding the SDP problem in a self-dual problem with known feasible starting point was investigated for SDP in [13] and [36]. A solution of the self-dual embedding gives information about the solution of the original problem. This analysis was extended in [16] to include pathological cases caused by the weaker duality theory of SDP (as compared to LP). In the latter case the stronger ELSD (extended Lagrange-Slater) dual problem is used in the embedding. These duals have better properties than the usual Lagrangean duals, and were formulated by Ramana [47].

6 Further information

An up-to-date list of publications dealing with SDP may be found in the *semidefinite programming homepage*, maintained by Christolph Helmberg. The address is

`http://www.zib-berlin.de/~bzfhelmb/semidef.html`

Available SDP software can also be accessed via this address.

References

[1] F. Alizadeh. *Combinatorial optimization with interior point methods and semi–definite matrices.* PhD thesis, University of Minnesota, Minneapolis, USA, 1991.

[2] F. Alizadeh. Interior point methods in semidefinite programming with applications to combinatorial optimization. *SIAM Journal on Optimization* **5**, 13–51, 1995.

[3] F. Alizadeh, J.-P.A. Haeberley, and M.L. Overton. Primal-dual methods for semidefinite programming: convergence rates, stability and numerical results. Technical report 721, NYU Computer Science Dept, New York University, New York, NY, 1996.

[4] K.M. Anstreicher and M. Fampa. A long-step path following algorithm for semidefinite programming problems. Working Paper, Department of Management Sciences, University of Iowa, Iowa City, USA, 1996.

[5] M.S. Bazarraa, H.D. Sherali, and C.M. Shetty. *Nonlinear Programming: Theory and Algorithms.* John Wiley and Sons, New York, 1993.

[6] R. Bellman and K. Fan. On systems of linear inequalities in hermitian matrix variables. In V.L. Klee, editor, *Convexity*, Vol. 7 Proc. Symposia in Pure Mathematics, pages 1–11. Amer. Math. Soc., Providence, RI, 1963.

[7] A. Ben-Tal and A. Nemirovskii. Potential reduction polynomial time method for truss topology design. *SIAM Journal on Optimization* **4**(3), 596–612, 1994.

[8] M.P. Bendsøe. *Optimization of Structural Topology, Shape, and Material.* Springer, 1995.

[9] M.P. Bendsøe, A. Ben-Tal, and J. Zowe. Optimization methods for truss geometry and topology design. *Structural Optimization* **7**, 141–159, 1994.

[10] S.E. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear matrix inequalities in system and control theory.* SIAM Studies in Applied Mathematics, Vol. 15. SIAM, Philadelphia, USA, 1994.

[11] B. Craven and B. Mond. Linear programming with matrix variables. *Linear Algebra Appl.* **38**, 73–80, 1981.

[12] E. de Klerk, C. Roos, and T. Terlaky. Semi-definite problems in truss topology optimization. Technical Report 95–128, Faculty of Technical Mathematics and Computer Science, Delft University of Technology, Delft, The Netherlands, 1995.

[13] E. de Klerk, C. Roos, and T. Terlaky. Initialization in semidefinite programming via a self-dual, skew-symmetric embedding. Technical Report 96–10, Faculty of Technical Mathematics and Computer Science, Delft University of Technology, Delft, The Netherlands, 1996. (To appear in *OR Letters*).

[14] E. de Klerk, C. Roos, and T. Terlaky. On primal-dual path-following algorithms in semidefinite programming. Technical Report 96–102, Faculty of Technical Mathematics and Computer Science, Delft University of Technology, Delft, The Netherlands, 1996. (To appear *Proceedings of XIIIth Int. Conf. Math. Prog.*, Kluwer).

[15] E. de Klerk, C. Roos, and T. Terlaky. Polynomial primal-dual affine scaling algorithms in semidefinite programming. Technical Report 96–42, Faculty of Technical Mathematics and Computer Science, Delft University of Technology, Delft, The Netherlands, 1996. (To appear *J. Comb. Opt.*).

[16] E. de Klerk, C. Roos, and T. Terlaky. Infeasible-start semidefinite programming algorithms via self-dual embeddings. Technical Report 97–10, Faculty of Technical Mathematics and Computer Science, Delft University of Technology, Delft, The Netherlands,

1997. (To appear in Proceedings of the Workshop on Semidefinite Programming and Interior-Point Approaches for Combinatorial Optimization Problems, FIELDS Institute, May 15-17, 1996).

[17] L. Faybusovich. On a matrix generalization of affine–scaling vector fields. *SIAM J. Matrix Anal. Appl.* **16**, 886–897, 1995.

[18] L. Faybusovich. Semi–definite programming: a path–following algorithm for a linear–quadratic functional. *SIAM Journal on Optimization*, pages 1007–1024, 1996.

[19] A.V. Fiacco and G.P. McCormick. *Nonlinear programming: sequential unconstrained minimization techniques.* John Wiley & Sons, New York, 1968. (Reprint: Volume 4 of *SIAM Classics in Applied Mathematics*, SIAM Publications, Philadelphia, USA, 1990).

[20] R.M. Freund and S. Mizuno. Interior point methods: current status and future directions. *OPTIMA* **51**, 1996.

[21] M.X. Goemans and D.P. Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of Assoc. Comput. Mach.* **42**, 1115–1145, 1995.

[22] D. Goldfarb and K. Scheinberg. Interior point trajectories in semidefinite programming. Working Paper, Dept. of IEOR, Columbia University, New York, NY, 1996.

[23] M. Grötschel, L. Lovász, and A. Schrijver. *Geometric Algorithms and Combinatorial Optimization.* Springer–Verlag, Berlin, 1988.

[24] B. He, E. de Klerk, C. Roos, and T. Terlaky. Method of approximate centers for semi–definite programming. Technical Report 96–27, Faculty of Technical Mathematics and Computer Science, Delft University of Technology, Delft, The Netherlands, 1996. (To appear in *Optimization Methods and Software*).

[25] C. Helmberg, F. Rendl, R.J. Vanderbei, and H. Wolkowicz. An interior-point method for semidefinite programming. *SIAM Journal on Optimization* **6**, 342–361, 1996.

[26] B. Jansen, C. Roos, and T. Terlaky. Interior point methods: a decade after Karmarkar. A survey, with application to the smallest eigenvalue problem. *Statistica Neerlandica* **50**, 1995.

[27] F. Jarre. Interior-point methods for classes of convex programs. In T. Terlaky, editor, *Interior point methods of mathematical programming*, pages 255–296. Kluwer, Dordrecht, The Netherlands, 1996.

[28] F. Jarre, M. Kočvara, and J. Zowe. Optimal truss design by interior point methods. Technical Report 173, Institut für Angewandte Mathematik, Universität Erlangen-Nürnberg, Erlangen, Germany, 1996.

[29] J. Jiang. A long step primal–dual path following method for semidefinite programming. Technical Report, Dept of Applied Mathematics, Tsinghua University, Beijing, China, 1996. (To appear in *OR Letters*).

[30] N.K. Karmarkar. A new polynomial–time algorithm for linear programming. *Combinatorica* **4**, 373–395, 1984.

[31] M. Kojima, M. Shida, and S. Shindoh. Global and local convergence of predictor-corrector infeasible-interior-point algorithms for semidefinite programs. Technical Report B-305, Department of Mathematical and Computing Sciences, Tokyo Institute of Technology, Tokyo, Japan, 1995.

[32] M. Kojima, M. Shidah, and S. Shindoh. A predictor-corrector interior point algorithm for the semidefinite linear complementarity problem using the Alizadeh-Haeberley-Overton search direction. Technical Report B-311, Department of Mathematical and Computing Sciences, Tokyo Institute of Technology, Tokyo, Japan, 1996.

[33] M. Kojima, S. Shindoh, and S. Hara. Interior point methods for the monotone semidefinite linear complementarity problems. Technical Report No. 282, Department of Mathematical and Computing Sciences, Tokyo Institute of Technology, Tokyo, Japan, 1994. (To appear in *SIAM Journal on Optimization*).

[34] L. Lovász. On the Shannon capacity of a graph. *IEEE Trans. on Information Theory* **25**, 1-7, 1979.

[35] L. Lovász and A. Schrijver. Cones of matrices and set-functions and 0-1 optimization. *SIAM Journal on Optimization* **1**, 166-190, 1991.

[36] Z.-Q. Luo, J.F. Sturm, and S. Zhang. Duality and self-duality for conic convex programming. Technical Report 9620/A, Tinbergen Institute, Erasmus University Rotterdam, 1996.

[37] Z.-Q. Luo, J.F. Sturm, and S. Zhang. Superlinear convergence oa a Symmetric primal-dual path following algorithm for semidefinite programming. Technical Report 9607/A, Tinbergen Institute, Erasmus University Rotterdam, 1996.

[38] R.D.C. Monteiro. Primal-dual algorithms for semidefinite programming. Working Paper, School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, USA, 1995. (To appear in *SIAM Journal on Optimization*).

[39] M. Muramatsu. Affine scaling algorithm fails for semidefinite programming. Working Paper 16, Dept. of Mechanical Engineering, Sophia University, Tokyo, Japan, 1996.

[40] M. Muramatsu and R.J. Vanderbei. Primal-dual affine-scaling algorithm fails for semidefinite programming. Technical Report SOR 97-04, Princeton University, NJ 08544, 1997.

[41] A. Nemirovskii. Interior point polynomial time methods in convex programming. Course notes, Optimization Laboratory, Faculty of Industrial Engineering and Management, Technion-Israel Institute of Technology, Haifa, Israel, 1994.

[42] Yu. Nesterov. Quality of semidefinite relaxation for nonconvex quadratic optimization. CORE Discussion paper 9719, Belgium, March 1997.

[43] Yu. Nesterov and A.S. Nemirovskii. *Interior point polynomial algorithms in convex programming.* SIAM Studies in Applied Mathematics, Vol. 13. SIAM, Philadelphia, USA, 1994.

[44] Yu. Nesterov and M.J. Todd. Self-scaled barriers and interior-point methods for convex rogramming. Technical Report 1091, School of OR and IE, Cornell University, Ithaca, New York, USA, 1994. (To appear in *Mathematics of Operations Research*).

[45] J.A. Olkin. Using semi–definite programming for controller design in active noise control. *SIAG/OPT Views and News* **8**, 1–5, Fall 1996.

[46] F.A. Potra and R. Sheng. A superlinearly convergent primal–dual infeasible–interior–point algorithm for semidefinite programming. Reports on Computational Mathematics 78, Dept. of Mathematics, The University of Iowa, Iowa City, USA, 1995.

[47] M. Ramana. An exact duality theory for semidefinite programming and its complexity implications. *Mathematical Programming* **77**(2), 1997.

[48] M.V. Ramana and P.M. Pardalos. Semidefinite programming. In T. Terlaky, editor, *Interior point methods of mathematical programming*, pages 369–398. Kluwer, Dordrecht, The Netherlands, 1996.

[49] U.T. Ringertz. On finding the optimal distribution of material properties. *Structural Optimization* **5**, 265–267, 1993.

[50] C. Roos, T. Terlaky, and J.-Ph. Vial. *Theory and Algorithms for Linear Optimization: An interior point approach.* John Wiley & Sons, New York, 1997.

[51] M.P. Rossow and J.E. Taylor. A finite element method for the optimal design of variable thickness sheets. *AIAA Journal* **11**(11), 1566–1569, 1973.

[52] M. Shida, S. Shindoh, and M. Kojima. Existence of search directions in interior–point algorithms for the SDP and the monotone SDLCP. Manuscript, 1996. (To appear in *SIAM Journal on Optimization*).

[53] N.Z. Shor. Quadratic optimization problems. *Soviet Journal of Computer and System Sciences* **25**, 1–11, 1987.

[54] J.F. Sturm and S. Zhang. Symmetric primal–dual path following algorithms for semidefinite programming. Technical Report 9554/A, Tinbergen Institute, Erasmus University Rotterdam, 1995.

[55] J.F. Sturm and S. Zhang. On the long step path–following method for semidefinite programming. Technical Report 9638/A, Tinbergen Institute, Erasmus University Rotterdam, 1996.

[56] M.J. Todd, K.C. Toh, and R.H. Tütüncü. On the Nesterov-Todd direction in semidefinite programming. Working paper, School of OR and Industrial Engineering, Cornell University, Ithaca, New York 14853–3801, 1996.

[57] L. Vandenberghe and S. Boyd. A primal–dual potential reduction algorithm for problems involving matrix inequalities. *Math. Programming* **69**, 205–236, 1995.

[58] L. Vandenberghe and S. Boyd. Semidefinite programming. *SIAM Review* **38**, 49–95, 1996.

[59] S.J. Wright. *Primal–dual interior point methods.* SIAM, Philadelphia, 1997.

[60] Y. Ye. Approximating quadratic programming with bound constraints. Working paper, Dept. of Management Science, University of Iowa, Iowa City, USA, 1997.

[61] Y. Zhang. On extending primal-dual interior point algorithms from linear programming to semidefinite programming. Technical Report, Department of Mathematics and Statistics, University of Maryland at Baltimore County, Baltimore, USA, 1995.

Global optimization: from pure adaptive search to simulated annealing

H. Edwin Romeijn
Rotterdam School of Management
Erasmus University
E-mail: E.Romeijn@fac.fbk.eur.nl

1 Introduction

Many problems can be posed as mathematical programming problems, i.e. problems in which an objective function, that depends on a number of decision variables, has to be optimized subject to a set of constraints:

$$\text{maximize } f(x)$$
$$\text{subject to } x \in S.$$

The field of *global optimization* generally restricts itself to problems where the objective function f and constraints describing the feasible region S are *nonlinear*, and the number of decision variables is *finite* (thus $S \subseteq \mathbb{R}^n$). In this chapter we will further restrict ourselves to the case where the objective and constraint functions are continuous, and where S is a compact body.

When a global optimization problem possesses some special structure, e.g. it is a quadratic programming problem, a concave minimization problem, the objective function is Lipschitz continuous with known Lipschitz constant, etc. special purpose algorithms can be constructed that use this special structure. These algorithms will usually be deterministic methods, i.e., methods that do not involve stochastic concepts. Often a convergence guarantee can be provided. See for example Horst and Tuy [9] or Horst and Pardalos [8].

In the absence of a special (known) structure, it is common to resort to a stochastic approach to the problem. Examples of stochastic methods for global optimization are Multistart and Multi Level Single Linkage (see e.g. Rinnooy Kan and Timmer [18, 19]); (Pure) Random Search methods (see e.g. Solis and Wets [25]); and the Random Function approach (see e.g. Kushner [11], Boender [4] and Mockus [13]). In this chapter we will focus on another type of method, namely Simulated Annealing.

Simulated Annealing is a sequential random search technique. That is, it proceeds by sequentially generating (feasible) points to the problem according to some scheme. In particular, this scheme is designed to avoid getting trapped in local optima by accepting, in addition to transitions corresponding to an improvement in function value, also transitions corresponding to a deterioration. These deteriorations make it possible to move away from local optima and explore the feasible region S in its entirety. In this chapter we will describe how Simulated Annealing naturally emerges from a class of conceptual algorithms, called Adaptive Search algorithms, which in turn is a generalization of the Pure Adaptive Search algorithm. We will also present convergence results for particular Simulated Annealing algorithms. For more details we refer to Romeijn [20], and Romeijn and Smith [22, 21].

2 Pure Adaptive Search

The Pure Random Search (PRS) algorithm is a well-known method that proceeds simply by randomly generating a number of points in the feasible region S, and estimates the global optimum by choosing the best of these points in terms of objective function value. Even though it is very inefficient in practice (the expected number of iterations necessary to obtain some level of accuracy grows exponentially in the problem dimension n), it does possess the desirable property that the global optimum will be found with probability one if the sample size grows to infinity.

The Pure Adaptive Search (PAS) algorithm (see Patel, Smith, and Zabinsky [14] and Zabinsky and Smith [26]) differs from the PRS algorithm in that it forces improvement in each iteration. In PAS, an iteration point is generated from the uniform distribution on the subset of points that are *improving* with respect to the previous iteration point. More formally, the algorithm reads:

Pure Adaptive Search (PAS)

Step 0. Set $k = 0$ and $y_0 = -\infty$.

Step 1. Generate x_{k+1} uniformly distributed in $S_{k+1} = \{x \in S : f(x) > y_k\}$.

Step 2. Set $y_{k+1} = f(x_{k+1})$. Increment k and return to Step 1.

Zabinsky and Smith [26] have shown that, for a large class of global optimization problems, the expected number of iterations needed by PAS to obtain a certain level of solution accuracy grows at most *linearly* in the dimension of the problem. This result suggests there is hope for an efficient random search method for global optimization. In fact, several random search algorithms have already reported *empirical* linearity in dimension for optimizing quadratic functions (see e.g. Schrack and Borowski [23], Schumer and Steiglitz [24], and Solis and Wets [25]). Unfortunately, in practice, we encounter some difficulties when trying to implement the Pure Adaptive Search algorithm:

1. Constructing the *improving region*

$$S_k = \{x \in S : f(x) > f(x_{k-1})\}.$$

2. Generating a point uniformly distributed in S_k.

We could try to avoid these difficulties by using the acceptance-rejection method for generating a point in S_k. However, by using this implementation of Step 1 of the algorithm, we obtain the PRS algorithm mentioned above. This implementation is inefficient due to the fact that the number of trial points in each PAS iteration necessary to find a point in S_k increases exponentially in the dimension of the problem.

The Adaptive Search method attempts to avoid this by, instead of generating uniform points in S, generating points from a nonuniform distribution that assigns larger probability to the improving region S_k.

3 Adaptive Search

3.1 The algorithm

We start with a family of probability distributions $\Pi = \{\pi_T; T \in \mathbf{T}\}$ over S, where $\mathbf{T} \subset \mathbb{R} \cup \{\infty\}$. Assume that $\infty \in \mathbf{T}$, and let $\pi_\infty = \varphi$, the uniform distribution on S. We assume that each member of this family satisfies the following assumption:

Assumption 3.1 *For all $T \leq T'$ $(T, T' \in \mathbf{T})$, and all $0 \leq u \leq v$*

$$\frac{\pi_T(S_u)}{\pi_T(S_v)} \geq \frac{\pi_{T'}(S_u)}{\pi_{T'}(S_v)}$$

where

$$S_u = \{x \in S : f(x) > f^* - u\}.$$

This assumption means that, if we condition on being in any level set S_v, the distribution of the objective function value under distribution π_T for X stochastically dominates the distribution of $f(X)$ under $\pi_{T'}$ for all $T \leq T'$. Now consider the following generalization of the PAS algorithm:

Adaptive Search (AS)

Step 0. Set $k = 0$, $T_0 = \infty$, and $y_0 = f_*$.

Step 1. Generate x from the distribution π_{T_k} over S. If $f(x) > y_k$, set $x_{k+1} = x$. Otherwise, repeat Step 1.

Step 2. Set $y_{k+1} = f(x_{k+1})$ and set the parameter $T_{k+1} = \tau(y_{k+1})$, where τ is a $\mathbf{T}$-valued nonincreasing function. Increment k and return to Step 1.

An important advantage of this algorithm is that we need only to sample from the feasible region S, instead of from a nested set of smaller level sets of f. The price we have to pay for this is that the distribution from which we have to sample is not simply the uniform distribution, and can change during the course of the algorithm.

Note that if we choose $\pi_T = \varphi$ for all T, then the AS algorithm reduces to the PAS algorithm, where every iteration is implemented using the acceptance-rejection method. In other words, effectively this yields the PRS algorithm. However, for the general case we can influence the number of trial points necessary in Step 1 by an appropriate choice of the parameter T in Step 2 (where this choice will depend on the particular shape of the distributions π_T). We will refer to the parameter T as the *temperature parameter*. A particular choice of temperature parameters $(T_k; k = 0, 1, \ldots)$ is called a *cooling schedule*. We allow the cooling schedule to be *adaptive*, i.e. we allow the temperature T_k in iteration k to depend on the function value in the previous iteration. More formally, the cooling schedule is given by a nonincreasing function τ which is a measurable function from $[f_*, f^*]$ to $\mathbf{T}$. The temperature T_k is then given by $T_k = \tau(f(X_k))$.

It can be shown that the number of iterations of AS is stochastically less than the number of iterations of PAS, and thus also bounded by a linear function in the problem dimension n.

3.2 The generating distribution

It will prove useful to further restrict the family distributions π_T beyond satisfying assumption 3.1. In particular, we will choose π_T to be of the following general form:

$$\pi_T(B) = \int_B g_T(x)\,\varphi(dx) \qquad \text{for all } B \in \mathcal{B}_S$$

where $\mathcal{B}_S$ is the Borel σ-field on S, and where

$$g_T(x) = h_T(f(x)).$$

Under the following condition on the functions h_T, the corresponding family Π satisfies assumption 3.1.

Assumption 3.2 *For all $T \leq T'$ $(T, T' \in \mathbf{T})$,*

$$\frac{h_T(u)}{h_{T'}(u)}$$

is a nondecreasing function.

Recall that $\pi_\infty = \varphi$, i.e. $h_\infty \equiv 1$. Thus, assumption 3.2 implies that each of the functions h_T is nondecreasing, by choosing $T' = \infty$.

3.3 The cooling schedule

As noted before, we can use the cooling schedule as a tool for influencing the number of points that have to be generated in an iteration to obtain an improving point. In particular, we would like to choose the temperature parameter T_k in every iteration of the AS algorithm in such a way that the expected number of trial points needed to generate a point in S_k is small. This leads to the definition of the

Adaptive Search cooling schedule
Choose the next temperature T_k so that the probability of generating an improvement under π_{T_k} is (at least) $1 - \alpha$.

If we choose the temperatures according to this cooling schedule, the expected number of trials in each iteration is (at most) equal to $\frac{1}{1-\alpha}$, independent of the dimension d of the problem. In principle, it is possible to compute a temperature $T(u;\alpha)$ according to the Adaptive Search cooling schedule, i.e. such that $\pi_{T(u;\alpha)}(S_u) \geq 1 - \alpha$ for $\alpha \in (0, 1]$. However, in practice this will be an extremely difficult task. In the next section, we will present a cooling schedule that, either exactly or asymptotically, satisfies this condition, for a specific choice of the h_T's and for some classes of optimization problems.

4 Simulated annealing

Simulated annealing originated from an analogy with the physical annealing process of finding low energy states of a solid in a heat bath (see Metropolis et al. [12]). Pincus [16] developed an algorithm based on this analogy for solving discretizations of continuous global optimization problems. Most of the other applications to date have been to discrete combinatorial optimization problems (see e.g. Kirkpatrick, Gelatt and Vecchi [10], Aarts and Korst

[1], and Aarts and Van Laarhoven [2]). Formulations of the simulated annealing algorithm for continuous optimization have also been proposed (see e.g. Bohachevsky, Johnson and Stein [5], Corana et al. [6] and Dekkers and Aarts [7]).

Consider a family of distributions $\{\pi_T;\ T \in [0,\infty]\}$ from the type as discussed above, i.e. such that assumption 3.1 is satisfied. Assume that the functions h_T defining the densities of the distributions π_T (for $T > 0$) satisfy the following assumption:

Assumption 4.1 *For every* $f_* \leq u < v \leq f^*$,

$$\lim_{T \downarrow 0} \frac{h_T(u)}{h_T(v)} = 0.$$

Under this assumption the distribution π_T will, for small T, "concentrate near the global maximum" of the global optimization problem. More formally, for all $\epsilon > 0$, $\lim_{T \downarrow 0} \pi_T(S_\epsilon) = 1$.

Now assume that we are given a Markov kernel $R(\cdot,\cdot)$, the *selection Markov kernel*, on $(S, \mathcal{B}_S)$. Here $\mathcal{B}_S$ denotes the Borel σ-field on S. Thus, for each $x \in S$, $R(x,\cdot)$ is a probability measure on $(S, \mathcal{B}_S)$ and for each $B \in \mathcal{B}_S$, $R(\cdot, B)$ is a measurable function. We are also given a *cooling schedule* $\tau = (\tau_k;\ k = 0, 1, \ldots)$ on $(S, \mathcal{B}_S)$. By this we mean that for each k, τ_k is a measurable function from S^{k+1} to $[0,\infty]$. The Simulated Annealing Algorithm with *state space* S, with *objective function* f, with selection Markov kernel R and with cooling schedule τ, constructs a sequence of *states* $X_0, X_1, \ldots$, a sequence of *candidate points* $Y_1, Y_2, \ldots$ and a sequence of temperatures $T_0, T_1, \ldots$ iteratively, according to the following algorithm:

Simulated Annealing

Step 0. Choose $x_0 \in S$ and $t_0 \in [0,\infty]$. Set $k = 0$.

Step 1. Select y_{k+1} according to the probability distribution $R(x_k, \cdot)$.

Step 2. Set

$$x_{k+1} = \begin{cases} y_{k+1} & \text{with probability } \beta_{t_k}(x_k, y_{k+1}) \\ x_k & \text{otherwise} \end{cases}$$

and set $t_{k+1} = \tau_{k+1}(x_0, \ldots, x_{k+1})$.

Step 3. Increment k and return to Step 1.

The *acceptance probability* $\beta_T(x,y)$ of accepting a next candidate point y, given that the current state is x and the current temperature is T, is given by

$$\beta_T(x,y) = \min\left(1, \frac{h_T(f(y))}{h_T(f(x))}\right).$$

Note that the cooling schedule is more general than in the previous section. There the temperature was only allowed to change in *improving* iterations, and was a function of only the best function value found so far.

We will now state sufficient conditions under which the sequence of function values $(f(X_n);\ n = 0, 1, \ldots)$ converges in probability to f^*:

C1. The set S is compact. Furthermore, for all $\epsilon > 0$, $\varphi(\{x \in S : f(x) > f^* - \epsilon\}) > 0$.

C2. The selection Markov kernel R is absolutely continuous (with respect to φ) and the corresponding density is uniformly bounded away from zero. That is,

$$R(x, B) = \int_B r(x, y)\, \varphi(dy)$$

with $\inf_{x,y \in S} r(x, y) > 0$.

C3. For every open subset G in S, $R(x, G)$ is continuous in x.

C4. For every choice of initial state x_0 and initial temperature t_0, the sequence of temperatures $(T_n;\ n = 0, 1, \ldots)$ converges to zero in probability.

The following theorem is due to Bélisle [3], who proved the result for a specific choice of the functions h_T and under a slightly more restrictive set of conditions.

Theorem 4.2 (cf. Bélisle [3]) *Let f be a function defined on a compact set S. Let X_1, $X_2, \ldots$ be the sequence of states induced by the simulated annealing algorithm with selection Markov kernel R and with cooling schedule τ. Assume that conditions C1 to C4 are satisfied. Let $f^* = \sup_{x \in S} f(x)$. Then, for every set of initial conditions (x_0, t_0) the sequence of function values $(f(X_k);\ k = 0, 1, \ldots)$ converges in probability to f^*.*

5 The Hide-and-Seek algorithm

In the previous section the Markov kernel R was arbitrary, subject to conditions C2 and C3. In this section we will make the additional assumption that the Markov chain defined by R converges in total variation to φ. In that case, the sequence of points generated by the Simulated Annealing algorithm at a *fixed* temperature T converges in total variation to π_T. Now let us return to the restricted cooling schedule of the form $\tau_k(x_0, \ldots, x_k) = \tau(\max_{0 \le i \le k} f(x_i))$ for some function τ. Then we see that each of the points corresponding to a *record value* of the sequence of function values generated by the Simulated Annealing algorithm is *approximately* distributed according to π_T for some value of T. In other words, the resulting algorithm, called *Hide-and-Seek*, can be seen as an approximate implementation of the Adaptive Search algorithm. Using the result that the number of iterations of the AS algorithm is at most linear in the problem dimension n, we may hope that the number of temperature changes of the Simulated Annealing algorithm is linear in dimension as well, for a certain class of global optimization problems.

Following Pincus [15] from this point on we make the following choice for the functions h_T, characterizing the generating distributions:

$$h_T(y) = \frac{e^{y/T}}{\int_S e^{f(x)/T}\, \varphi(dx)}.$$

It is easy to check that this choice for h_T satisfies all assumptions previously made. Note that the distribution π_T with this density is the Boltzmann distribution with parameter T commonly used in other Simulated Annealing algorithms. For some special cases, we can now specify the Adaptive Search cooling schedule, either exactly or asymptotically:

Special cases

CASE 1. LINEAR PROGRAMMING.
If f is linear, S is a full-dimensional polytope and the optimal solution is unique and nondegenerate, then

$$\tau(y) = \frac{f^* - y}{\gamma_{1-\alpha}(d,1)}.$$

Alternatively, if f is linear, S is the *boundary* of a full-dimensional polytope, and the optimal solution is unique an nondegenerate, then

$$\tau(y) = \frac{f^* - y}{\gamma_{1-\alpha}(d-1,1)}.$$

CASE 2. CONVEX MAXIMIZATION.
If f is convex and S is a full-dimensional polytope (or the boundary of a full-dimensional polytope), then we get asymptotically the same cooling schedules as in case 1 above.

CASE 3. CONCAVE MAXIMIZATION.
If f is a strictly concave quadratic function, and S is a level set of f, then we obtain the following cooling schedule:

$$\tau(y) = \frac{f^* - y}{\gamma_{1-\alpha}(d/2,1)}. \tag{1}$$

If we can approximate f by a (strictly concave) quadratic function, using a second order Taylor expansion, then (1) can be used as an approximation of the desired cooling schedule. Also, if the Hessian of f exists and is negative definite at the global optimum, then (1) asymptotically is the desired cooling schedule.

6 Concluding remarks

In this chapter we have derived the Simulated Annealing algorithm as an approximate implementation of a conceptual class of Adaptive Search algorithms enjoying a number of attractive properties. This derivation provides not only a more rigorous motivation for the Simulated Annealing algorithm, compared to the common motivation from the analogy with a physical annealing process. In addition, it quite naturally suggests a solution to one of the problems that is usually encountered when implementing simulated annealing, namely the choice of the cooling schedule to be used.

A remaining question for future research is to analyze the complexity of the Simulated Annealing algorithm as developed above. A first step in this direction has recently be made: Reaume [17] showed that the Adaptive Search algorithm, as well as the Simulated Annealing algorithm, can be implemented in polynomial time for convex programming problems.

References

[1] E.H.L. Aarts and J.H.M. Korst. *Simulated Annealing and Boltzmann Machines.* Wiley, Chichester, 1988.

[2] E.H.L. Aarts and P.J.M. van Laarhoven. Simulated annealing: an introduction. *Statistica Neerlandica* **43**, 31–52, 1989.

[3] C.J.P. Bélisle. Convergence theorems for a class of simulated annealing algorithms on oR^d. *Journal of Applied Probability* **29**(4), 885–895, 1992.

[4] C.G.E. Boender. *The Generalized Multinomial Distribution: a Bayesian analysis and applications.* PhD thesis, Econometric Institute, Erasmus University Rotterdam, Rotterdam, The Netherlands, 1984.

[5] I.O. Bohachevsky, M.E. Johnson, and M.L. Stein. Generalized simulated annealing for function optimization. *Technometrics* **28**(3), 209–217, 1986.

[6] A. Corana, M. Marchesi, C. Martini, and S. Ridella. Minimizing multimodal functions for continuous variables with the "simulated annealing" algorithm. *ACM Transactions on Mathematical Software* **13**, 262–280, 1987.

[7] A. Dekkers and E. Aarts. Global optimization and simulated annealing. *Mathematical Programming* **50**, 367–393, 1991.

[8] R. Horst and P.M. Pardalos, editors. *Handbook of Global Optimization.* Kluwer Academic Publishers, Dordrecht, The Netherlands, 1995.

[9] R. Horst and H. Tuy. *Global Optimization.* Springer-Verlag, Berlin, Germany, 1993.

[10] S. Kirkpatrick, C.D. Gelatt Jr., and M.P. Vecchi. Optimization by simulated annealing. *Science* **20**, 671–680, 1983.

[11] H.J. Kushner. A versatile stochastic model of a function of unknown and time-varying form. *Journal of Mathematical Analysis and Applications* **5**, 150–167, 1962.

[12] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, and E. Teller. Equations of state calculations by fast computing machines. *The Journal of Chemical Physics* **21**, 1087–1092, 1953.

[13] J. Mockus. *Bayesian Approach to Global Optimization.* Kluwer Academic Publishers, Dordrecht, The Netherlands, 1989.

[14] N.R. Patel, R.L. Smith, and Z.B. Zabinsky. Pure adaptive search in Monte Carlo optimization. *Mathematical Programming* **43**, 317–328, 1988.

[15] M. Pincus. A closed form solution for certain programming problems. *Operations Research* **16**, 690–694, 1968.

[16] M. Pincus. A Monte Carlo method for the approximate solution of certain types of constrained optimization problems. *Operations Research* **18**, 1225–1228, 1970.

[17] D.J. Reaume. *Convex and Global Optimization by Random Sampling.* PhD thesis, Department of Industrial and Operations Engineering, College of Engineering, The University of Michigan, Ann Arbor, Michigan, April 1997.

[18] A.H.G. Rinnooy Kan and G.T. Timmer. Stochastic global optimization methods; part I: clustering methods. *Mathematical Programming* **39**, 27–56, 1987.

[19] A.H.G. Rinnooy Kan and G.T. Timmer. Stochastic global optimization methods; part II: multi level methods. *Mathematical Programming* **39**, 57–78, 1987.

[20] H.E. Romeijn. *Global Optimization by Random Walk Sampling Methods.* Thesis Publishers, Amsterdam, The Netherlands, 1992.

[21] H.E. Romeijn and R.L. Smith. Simulated annealing and adaptive search in global optimization. *Probability in the Engineering and Informational Sciences* **8**, 571–590, 1994.

[22] H.E. Romeijn and R.L. Smith. Simulated annealing for constrained global optimization. *Journal of Global Optimization* **5**, 101–126, 1994.

[23] G. Schrack and N. Borowski. An experimental comparison of three random searches. In F. Lootsma, editor, *Numerical Methods for Nonlinear Optimization*, pages 137–147. Academic Press, London, United Kingdom, 1972.

[24] M.A. Schumer and K. Steiglitz. Adaptive step size random search. *IEEE Transactions on Automatic Control* **AC-13**, 270–276, 1968.

[25] F.J. Solis and R.J.-B. Wets. Minimization by random search techniques. *Mathematics of Operations Research* **6**, 19–30, 1981.

[26] Z.B. Zabinsky and R.L. Smith. Pure adaptive search in global optimization. *Mathematical Programming* **53**(3), 323–338, 1992.

Generalized Job Shop Scheduling: Complexity and Local Search

R.J.M. Vaessens
Eindhoven University of Technology
E-mail: nlklm3tm@ibmmail.com

1 Introduction

In my thesis [Vaessens, 1995] a particular scheduling problem has been studied, called the *generalized job shop scheduling* problem. Scheduling problems occur in situations where a set of activities has to be performed by a set of scarce resources [Baker and Su, 1974]. Scheduling theory is concerned with the optimal assignment of these resources to the activities over time. Its applications can be found in various areas like production planning, personnel planning, computer system control, and time tabling. Over the past decades, scheduling theory has been the subject of extensive research. Most attention has been paid to *machine scheduling problems*, in which the resources are usually called *machines* and the activities *operations*. The main restriction is that a machine can perform at most one operation at a time.

To solve a practical scheduling problem by mathematical means it is necessary to abstract a model from it. This abstraction must capture the essential elements of the practical problem in the sense that it should be possible to convert a solution obtained for the model into a solution of comparable quality for the practical problem. A major disadvantage of most models considered in the literature is that they are either too simple to reflect reality or too complex to be quickly solvable. The model introduced in my thesis intends to reduce this gap.

The remainder of this paper is organized as follows. Section 2 describes the generalized job shop scheduling problem in more detail. Since the problem is hard to solve, we studied approximative solution methods, and local search methods in particular. Section 3 gives a quick introduction in local search and neighborhood functions. Section 4 describes some neighborhood functions that could be used when applying local search to the generalized job shop scheduling problem. It also describes some features of these neighborhood functions. Finally, Section 5 gives some conclusions.

2 The generalized job shop scheduling problem

The generalized job shop scheduling problem (*GJSSP*) generalizes the models that have been studied in the literature in several ways. The most important extension of the model relaxes the requirement that each operation has to be processed by a single machine, which is known in advance, in two different ways. First, we allow an operation to be performed by one machine out of a given operation-dependent set of machines. Second, we allow that an operation may need the simultaneous cooperation of several machines. Combining these two features leads to a model, in which for each operation some *machine sets* are given, each of

which is capable of processing the operation. The selection of one such machine set for each operation is now part of the scheduling problem. Furthermore, arbitrary precedences between operations are included. On the other hand, only one optimality criterion is considered: the minimization of the maximum completion time.

The model of the GJSSP can now be described more formally. Given are a set of operations and a set of machines. For each operation, a collection of machine sets is given; each of these machine sets is capable of processing the operation. For each operation and each of its machine sets, a processing time is given. Furthermore, a binary precedence relation is given on the set of operations. A precedence between two operations denotes that the processing of the second operation cannot start before the processing of the first operation has been finished. Each operation is assumed to be available at time 0; furthermore, no preemption of operations is allowed.

A *schedule* consists of two parts: an assignment of operations to machine sets and an assignment of operations to time intervals. Obviously, given the assignment of operations to machine sets, it is sufficient to know the starting time of each operation. A schedule is called *feasible* if each operation is processed by one of its machine sets for the required duration of time, if at any time instant no machine takes part in the processing of more than one operation, if the precedences are satisfied, and if each starting time is nonnegative. Our goal is to find a feasible schedule that minimizes the maximum completion time over all operations.

To find an optimal schedule it is sufficient to look at left-justified schedules. A feasible schedule is called *left-justified* if it is not possible to complete any operation earlier such that

- each operation is processed by the same machine set, and
- on each machine the processing order remains the same.

For a given schedule there exists a unique left-justified schedule in which each operation is processed by the same machine set and in which the processing order on each machine remains the same. This left-justified schedule can be obtained from the original one by starting all operations as early as possible, that is, by shifting them to the left on a time-axis.

Assume for each operation a machine set has been chosen. Then each feasible schedule defines a unique order of the operations on all machines. On the other hand, each (feasible) order of the operations on all machines defines a unique left-justified schedule. Therefore, there is a one-to-one correspondence between the order of the operations on all machines and a left-justified schedule. So, a schedule can also be defined by the choice of a machine set for each operation and the order of the operations on the machines, and by defining its length by the length of the corresponding left-justified schedule. Note that choices for the order on one machine may imply forced choices for the order on other machines. If one would choose the reversed order of such a forced choice, the resulting schedule would be infeasible. In the remainder of this paper we will give a schedule by the choice of a machine set for each operation and the choice of the order of operations on all machines.

An operation is called *critical* in a given schedule when the length of the schedule increases if the duration of the operation is increased by one time unit (while leaving machine orderings and chosen machine sets unchanged).

3 Local search and neighborhood functions

Since the introduction of local search methods by Bock [1958] and Croes [1958], a large variety of local search algorithms has been proposed. Basically, local search algorithms work as follows. Starting from an initial solution, the algorithm visits a sequence of solutions. Each solution in the sequence is obtained from a previous one by making slight modifications. The art of finding good local search algorithms is to make good choices for the type of modifications and for the solution that should be modified. The aim is to finally find solutions that have a good quality.

To describe this more formally some notions have to be introduced. A *minimization problem* is specified by a class of problem *instances*, each of which is implicitly specified by a triple $(\mathcal{S}, \mathcal{X}, f)$. Here, the *solution space* $\mathcal{S}$ is the set of all (feasible) solutions, the *cost space* $\mathcal{X}$ is a totally ordered set of all possible cost values, and the *cost function* f is a mapping $f : \mathcal{S} \to \mathcal{X}$. The optimal cost f_{opt} of an instance is defined by $f_{\text{opt}} = \min\{f(s) | s \in \mathcal{S}\}$, and the set of optimal solutions is denoted by $\mathcal{S}_{\text{opt}} = \{s \in \mathcal{S} | f(s) = f_{\text{opt}}\}$. The objective is to find some solution $s_{\text{opt}} \in \mathcal{S}_{\text{opt}}$.

A *neighborhood function* $\mathcal{N}$ is a mapping $\mathcal{N} : \mathcal{S} \to \mathcal{P}(\mathcal{S})$, which specifies for each $s \in \mathcal{S}$ a subset $\mathcal{N}(s)$ of $\mathcal{S}$ of neighbors of s. A neighborhood function defines which modifications to a solution are allowed to make in a local search algorithm. A solution $s \in \mathcal{S}$ is called a *local minimum with respect to* $\mathcal{N}$ if $f(s) \leq f(t)$ for all $t \in \mathcal{N}(s)$. Furthermore, to distinguish between local minima and elements of $\mathcal{S}_{\text{opt}}$, we call the latter ones *global* minima.

An interesting property of neighborhood functions is *connectivity*. This property expresses to which extent solutions can be reached from an initial solution when making a sequence of transitions in which each next solution is a neighbor of the previous one.

To be more specific we have to introduce several notions. For a neighborhood function $\mathcal{N} : \mathcal{S} \to \mathcal{P}(\mathcal{S})$ we define the set $A_{\mathcal{N}}$ as

$$A_{\mathcal{N}} = \{(x, y) \mid x \in \mathcal{S}, y \in \mathcal{N}(x)\}.$$

So $A_{\mathcal{N}}$ contains pairs of solutions for which the second solution is a neighbor of the first solution. The directed graph $\mathcal{G}_{\mathcal{N}} = (\mathcal{S}, A_{\mathcal{N}})$ is called the *neighborhood graph* corresponding to $\mathcal{N}$. A neighborhood function $\mathcal{N}$ is called *strongly connected* if the corresponding neighborhood graph is strongly connected (so there is a directed path in the neighborhood between any two solutions). A neighborhood function $\mathcal{N}$ is called *optimum connected* if for each solution there exists a path to an optimal solution in the corresponding neighborhood graph. Note that each strongly connected neighborhood function is also optimum connected.

The extent to which a neighborhood function is connected has consequences for local search algorithms. For instance, if a given neighborhood function is not optimum connected, then there are solutions for which no sequence of transitions leads to an optimal solution. In this case any local search algorithm that starts with such an initial solution and makes only transitions to neighboring solutions is unable to find an optimal solution. On the other hand, there exist local search algorithms, for instance the standard simulated annealing algorithm [Van Laarhoven, 1988], that asymptotically converge to an optimal solution if the neighborhood function is optimum connected.

4 Neighborhood functions for the GJSSP

A neighborhood function for the GJSSP must be capable of modifying the chosen machine set for an operation, and it must be capable of changing for a given operation the order

relative to other operations that have a machine in common with the given operation in their chosen machine set. For the most elementary neighborhood function that satisfies these conditions a neighbor is obtained as follows: first, choose an operation and delete it from all machine orderings; next, assign to this operation a (possibly different) machine set; finally, insert this operation in the machine orderings corresponding to the new machine set, such that the resulting schedule is feasible. We assume that a solution cannot be a neighbor of itself, so that at least the machine orderings of a schedule and each of its neighbors are different.

One can think of several variants of this elementary *reinsertion* neighborhood function. First, we introduce a variant $\mathcal{N}_2$, in which an operation is reinserted on an arbitrary machine set, such that no other reinsertion on that machine set gives a smaller length. Next, we introduce a third variant $\mathcal{N}_3$, in which an operation is reinserted in the best way with the best machine set possible.

Note however, that a neighbor obtained by reinserting a non-critical operation does have a critical path that is at least as long as the critical path of the current schedule. The reason for this is that the critical path of the current schedule remains present in such a neighbor. So it may be profitable to consider only neighbors that are obtained by reinserting critical operations of the current schedule. We will denote the neighborhood functions corresponding to $\mathcal{N}_1$, $\mathcal{N}_2$, and $\mathcal{N}_3$ in which only critical operations may be reinserted by $\mathcal{N}_1^c$, $\mathcal{N}_2^c$, and $\mathcal{N}_3^c$, respectively.

In my thesis it is described how feasible neighbors can be found efficiently, since this becomes more difficult when the machine set of an operation has been changed. It is also described how a best insertion, given the machine set, can be determined efficiently.

5 Connectivity of neighborhood functions

In the following we discuss the connectivity of the neighborhood functions introduced above. A summary of the results has been depicted in Figure 1. The sign + denotes that a neighborhood function is strongly (s.c.) or optimum connected (o.c.) and the sign − that it is not. Arrows denote that a result for the connectivity of one neighborhood function implies a result for the connectivity of a second neighborhood function. Bold circles denote results that are not implied by other results.

It is not difficult to show that the neighborhood function $\mathcal{N}_1$ is strongly connected and therefore also optimum connected. To prove this result we have to show that we can construct a sequence of subsequent neighboring schedules from a given initial schedule to a given final schedule. We construct this sequence in two stages. In the first stage we modify the chosen machine sets into the machine sets of the given final solution; here, we do not pay attention to the order of the operations. In the second stage the machine set assignment remains the same for all operations and only the machine orderings are modified in such a way that the resemblance with the final schedule increases. The most difficult part of the proof is to show that each schedule constructed is a feasible one.

We leave it as an excercise for the reader that in general the neighborhood function $\mathcal{N}_2$ is not optimum connected and thus not strongly connected either. This also implies that $\mathcal{N}_3$ is not optimum connected, nor strongly connected. The proof proceeds by defining an instance of the GJSSP and an initial schedule of this instance (which of course is not optimal), such that every sequence of best insertions on an arbitrarily chosen machine set does not lead to an optimal schedule. An instance with only eight operations and one possible machine set for each operation will do the job.

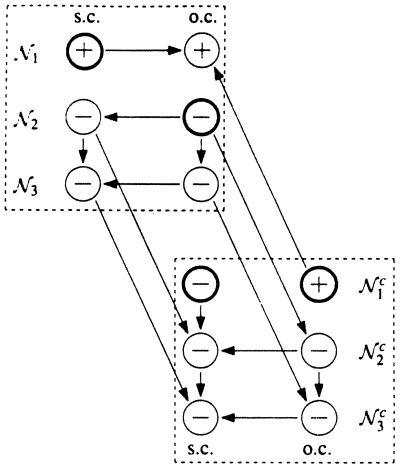


Figure 1: Connectivity of various neighborhood functions.

Now we consider the neighborhood functions in which only critical operations may be reinserted. As a consequence of the previous result the neighborhood functions $\mathcal{N}_2^c$ and $\mathcal{N}_3^c$ are in general not optimum connected nor strongly connected. So the only neighborhood function that remains to be studied is $\mathcal{N}_1^c$. It can be easily shown that that this neighborhood function is in general not strongly connected. An instance with two machines and one operation with a large processing time on the first machine and two operation with a very small processing time on the second machine will do. There are two possible schedules, but the critical path is defined by the operation on the first machine. So the two schedules cannot be reached from each other. An instance due to Van Laarhoven [1988] shows that it is also impossible to prove connectivity from any initial schedule to a schedule that is identical on the critical path of a given final schedule. However, it can be shown that $\mathcal{N}_1^c$ is optimum connected by constructing a sequence of neighboring schedules from an arbitrary initial schedule to an optimal schedule [Vaessens, 1995].

6 Conclusions

Various elementary neighborhood functions for the generalized job shop scheduling problem have been studied. Application of local search algorithms may show different results for the various neighborhood functions. For neighborhood functions that are strongly or optimum connected local search algorithms have the ability of reaching an optimal schedule when started from an arbitrary initial schedule. With other neighborhood functions an optimal schedule is not always reachable. However, the latter neighborhood functions may lead to a faster convergence to a schedule of good quality. Computational results should be obtained to verify the validity of these theoretical statements. Furthermore, it should be investigated

whether more complicated neighborhood functions can lead to good results at the expense of small amounts of compution time. Research in this direction should be motivated by the fact that the application of complicated neighborhood functions have been very successful for the ordinary job shop scheduling problem (see also [Vaessens, 1995]).

Bibliography

K.R. BAKER, Z.-S. SU (1974), Sequencing with due-dates and early start times to minimize maximum tardiness, *Naval Res. Logist. Quart.* **21**, 171-176.

F. BOCK (1958), *An Algorithm for Solving "Traveling Salesman" and Related Network Optimization Problems*, Internal report, Armour Research Foundation, Chicago, Illinois.

R.J.M. VAESSENS (1995), Generalized Job Shop Scheduling: Complexity and Local Search, Ph.D. thesis, Eindhoven University of Technology, Eindhoven, The Netherlands.

P.J.M. VAN LAARHOVEN (1988), *Theoretical and Computational Aspects of Simulated Annealing*, Ph.D. thesis, Erasmus University Rotterdam, Rotterdam, The Netherlands.

Convex approximations for stochastic programs with simple integer recourse

Maarten H. van der Vlerk
Faculty of Economics
University of Groningen
E-mail: m.h.van.der,vlerk@eco.rug.nl

Abstract

We review convex approximations for stochastic programs with simple integer recourse. Both for the case of discrete and continuous random variables such approximations are discussed, and representations as continuous simple recourse problems are given.

1 Introduction

In this paper we review results on convex approximations for certain stochastic programming models with mixed-integer variables. It summarizes part of my PhD thesis [13], which was written while I held an AIO position (supervisors W.K. Klein Haneveld and L. Stougie) financed by the LNMB.

Before presenting the simple integer recourse model, we first give a brief introduction to stochastic (linear) programming (see e.g. [4, 10]).

1.1 Stochastic programming

We consider a two-stage stochastic linear programming model with integer recourse. Such a model is at the intersection of two different branches of mathematical programming. On the one hand some of the model parameters are random, which places the problem in the field of *stochastic programming*. On the other hand some decision variables are required to be integers, so that this model also belongs to the field of *integer programming*.

Stochastic linear programming models arise from linear programming models if some of the parameters are random. For example, consider the linear program

$$\min\{cx : \bar{A}x = \bar{b},\ x \in \mathbb{R}^n_+\},$$

with $c \in \mathbb{R}^n$, $\bar{b} \in \mathbb{R}^m$ and $\bar{A} \in \mathbb{R}^{m \times n}$. In many practical situations the model builder is uncertain about the values of (some of) the parameters $(c, \bar{A}, \bar{b})$. In stochastic programming this uncertainty is modeled by replacing (some of) the fixed parameters by random variables whose probability distribution is assumed to be known. However, such a model is no longer well-defined. Therefore, additional specifications have to be made to get a meaningful model. One way to do this is as follows. Given a decision vector x, for any realization of $(c, \bar{A}, \bar{b})$ one can compute the value cx and the size of the infeasibility $\bar{b} - \bar{A}x$. It is assumed that in addition to the original model a penalty function v is used that transforms the infeasibility

$\bar{b} - \bar{A}x$ into a cost comparable with the objective function cx. Thus, taking the expectation over all realizations of the random parameters $(c, \bar{A}, \bar{b})$, one computes the expected value of $cx + v(\bar{b} - \bar{A}x)$ which is taken as the criterion for minimization.

In stochastic linear programming it is common practice to define the function v as the value function of a second linear program, which has the infeasibilities $\bar{b} - \bar{A}x$ in the right-hand side of the constraints. The variables of this linear program represent so-called *recourse actions*. By using them the feasibility is restored afterwards, so to speak, at certain cost. Therefore the second linear program usually is called the *second-stage* problem, and the corresponding stochastic linear programming problem is called a model with *recourse*. Familiar stochastic linear programming models have continuous recourse. By this we mean, that the recourse variables are continuous variables. In this paper we deal with stochastic linear programs with *integer recourse*. The property which sets these models apart, is that the recourse variables are integer variables.

Integer linear programs are NP-hard, which causes serious additional problems on top of those already present in solving stochastic programs with continuous recourse. Indeed, given a choice of the decision variables x, for each realization of the random parameters we have to solve an integer problem in order to compute the expected value of $cx + v(\bar{b} - \bar{A}x)$. Thus, it is clear that solving stochastic linear programs with general integer recourse by straightforward computation is impossible for any non-trivial problem size. Instead, we need structural properties of specific models in order to formulate more sophisticated solution strategies. In Section 2 we give an overview of relevant properties of the so-called simple integer recourse model, which will be defined in Section 1.2. In Sections 3 and 4 we use these properties to construct convex approximations.

We motivate our interest in stochastic linear programs with integer recourse by a small example problem.

Example 1.1 A firm needs to plan its production for the next period. In principle, this production must be large enough to meet the uncertain demand that becomes known only after the production has taken place. If the quantity produced falls short of the actual demand then the only possibility left is to buy a number of units from a competing firm, since every demand has to be satisfied. The objective is to minimize the total expected cost.

We assume that the production cost of one unit is 1, whereas the buying price at a competing firm equals 4 per unit. Moreover, we assume that the probability distribution of the demand is known: it is equal to 1000 with probability 0.9 and equal to 1300 with probability 0.1. The problem is: how many items should the be produced?

Below we will compare the optimal solutions of two models for this problem. First we assume that the firm can buy any quantity it needs from a competitor, which is represented in the model by a continuous recourse variable. In the second formulation it is assumed that only batches of a fixed size can be bought. Due to this additional assumption the latter model has an integer second-stage problem: the recourse variable now represents the number of batches that are bought to compensate for a shortage.

The optimal production quantity for the first model is 1000. If the demand turns out to be 1300, then 300 units have to be bought. Hence, the total expected cost is 1120, which consists of production cost 1000 and expected cost due to buying of $(1/10) \cdot 4 \cdot 300 = 120$.

Next we assume that only batches of size 250 can be bought. The price per unit remains the same, so that the price per batch is 1000. The optimal solution of this integer recourse problem is to produce 1050 units. Now one batch has to be bought in case the demand is 1300, so that the total expected cost is 1150, consisting of production cost 1050 and expected recourse cost $(1/10) \cdot 1000 \cdot 1 = 100$.

Finally, suppose that we choose to ignore the integer structure of the second-stage problem in the computations for the latter model. Then we obtain the solution of the continuous recourse model, that is, to produce 1000 units. However, the actual expected cost of this solution is $1000 + (1/10) \cdot 1000 \cdot 2 = 1200$, since we have to buy two batches if demand turns out to be 1300 units. We see that this solution is non-optimal by a margin of 50. ◁

1.2 Simple integer recourse

The *simple integer recourse model* with fixed technology matrix is defined as

$$\inf_x \left\{ cx + \mathcal{Q}(x) : Ax = b,\ x \in \mathbb{R}^{n_1}_+ \right\}, \tag{1}$$

where the *expected value function* $\mathcal{Q}$ is

$$\mathcal{Q}(x) = E_\xi\, v(\xi - Tx),$$

and v is the value function of the second-stage problem

$$\begin{aligned} v(s) = \inf_y \{ q^+y^+ + q^-y^- :\ & y^+ \geq s, \\ & y^- \geq -s, \\ & y^+, y^- \in \mathbb{Z}^{m_2}_+ \}, \qquad s \in \mathbb{R}^{m_2}. \end{aligned}$$

Here $c, A, b, (q^+, q^-)$ and T are vectors/matrices of the appropriate size, $q^+, q^- \geq 0$, $q^+ + q^- > 0$, and ξ is a random vector in $\mathbb{R}^{m_2}$.

As suggested by the name, this model has the same structure as the well-known simple *continuous* recourse model, in which the second-stage decision variables $y = (y^+, y^-)$ are non-negative reals. The expected value function of the latter problem is Lipschitz continuous and convex, so that in principle (i.e., disregarding possible difficulties in evaluating the integrals) it can be solved efficiently by standard techniques from mathematical programming, see [9]. Several special purpose algorithms exist, see e.g. [1, 4, 10, 14]. The expected value function of the integer problem lacks these favorable properties in general. For an overview of the field of stochastic integer programming we refer to [11, 12].

Using separability which is due to the simple recourse structure, $\mathcal{Q}$ is completely characterized by the one-dimensional generic function Q, given by

$$Q(z) = q^+ E_\xi \lceil \xi - z \rceil^+ + q^- E_\xi \lfloor \xi - z \rfloor^-, \quad z \in \mathbb{R},$$

where $q^+, q^- \in \mathbb{R}$, with $q^+, q^- \geq 0$, $q^+ + q^- > 0$ ξ is a random variable, and $\lceil s \rceil^+ = \max\{0, \lceil s \rceil\}$, $\lfloor s \rfloor^- = \max\{0, -\lfloor s \rfloor\}$, $s \in \mathbb{R}$.

2 Preliminary results

The results in this section are quoted from [8] and [13] where the reader is referred to for proofs.

Theorem 2.1 *Let Q be the one-dimensional expected value function of an integer simple recourse program, defined as*

$$Q(z) = q^+ E_\xi \lceil \xi - z \rceil^+ + q^- E_\xi \lfloor \xi - z \rfloor^-, \qquad z \in \mathbb{R},$$

where $q^+, q^- \geq 0$, $q^+ + q^- > 0$, and ξ is a random variable with finite expected value μ, and with right continuous cumulative distribution function (cdf) $F(s) = \Pr\{\xi \leq s\}$ and left continuous cdf $\hat{F}(s) = \Pr\{\xi < s\}$, $s \in \mathbb{R}$. Then

(a) *For all* $z \in \mathbb{R}$,

$$Q(z) = q^+ \sum_{k=0}^{\infty} \Pr\{\xi > z + k\} + q^- \sum_{k=0}^{\infty} \Pr\{\xi < z - k\}.$$

Moreover, the corresponding one-dimensional continuous simple recourse expected value function $\tilde{Q}$*, given by*

$$\tilde{Q}(z) = q^+ E_\xi(\xi - z)^+ + q^- E_\xi(\xi - z)^-, \qquad z \in \mathbb{R},$$

provides a convex lower bound as well as a convex upper bound for Q*:*

$$\tilde{Q}(z) \le Q(z) \le \tilde{Q}(z) + \max\{q^+, q^-\}, \qquad z \in \mathbb{R}.$$

For every $\alpha \in [0, 1)$ *the restriction of* Q *to* $\alpha + \mathbb{Z}$ *is convex.*

(b) *Let* $z_1 \le z_2$*. Then*

$$\begin{aligned} Q(z_2) - Q(z_1) \;=\; & -q^+ \sum_{k=0}^{\infty} \Pr\{z_1 + k < \xi \le z_2 + k\} \\ & + q^- \sum_{k=0}^{\infty} \Pr\{z_1 - k \le \xi < z_2 - k\}. \end{aligned}$$

In particular, for $n \in \{1, 2, \ldots\}$ *it holds, for all* $z \in \mathbb{R}$,

$$Q(z + n) - Q(z) = -q^+ \sum_{k=0}^{n-1} (1 - F(z + k)) + q^- \sum_{k=1}^{n} \hat{F}(z + k).$$

(c) *If* $(a, b) \subset \mathbb{R}$ *is an interval of length more than one such that* $\Pr\{a < \xi < b\} = 0$*, then* Q *is semi-periodic with period* 1 *and slope* $\gamma = -q^+ \Pr\{\xi > a\} + q^- \Pr\{\xi < b\}$ *on the interval* $[a, b] \cap \mathbb{R}$*. That is, if* $z_1 \in [a, b]$*,* $z_2 \in [a, b]$*,* $z_1 - z_2 = n \in \mathbb{Z}$*, then*

$$Q(z_2) - Q(z_1) = n\gamma.$$

In particular, if $\Pr\{\xi < \underline{\xi}\} = 0$ *for some* $\underline{\xi} > -\infty$ *then* Q *is semi-periodic with period* 1 *and slope* $-q^+$ *on* $(-\infty, \underline{\xi}]$*. Similarly, if* $\Pr\{\xi > \bar{\xi}\} = 0$ *for some* $\bar{\xi} < +\infty$ *then* Q *is semi-periodic with period* 1 *and slope* q^- *on* $[\bar{\xi}, \infty)$*.*

(d) *Suppose that* ξ *is* continuously distributed *with probability density function (pdf)* f*. Then* Q *is continuous on* $\mathbb{R}$*. If the pdf* f *has a finite total variation* $|\Delta|f$ *then* Q *is even Lipschitz continuous with a Lipschitz constant that is at most* $\max\{q^+, q^-\}(1 + |\Delta|f/2)$*. In that case* Q *has left and right derivatives everywhere, and* Q *is differentiable at* $z \in \mathbb{R}$ *if* f *is continuous at* $z + k$ *for all* $k \in \mathbb{Z}$*. Then*

$$Q'(z) = -q^+ \sum_{k=0}^{\infty} f(z + k) + q^- \sum_{k=0}^{\infty} f(z - k), \quad z \in \mathbb{R}.$$

Left and right derivatives are obtained from this expression by replacing f *by its left and right-continuous version, respectively.*

(e) *Suppose that* ξ *is* discretely distributed *with support* Ξ*. Then the function* Q *is lower semicontinuous, and the set of all discontinuity points of* Q *is given by* $\Xi + \mathbb{Z}$*. In particular,*

(i) *Q is continuous from the left but not from the right in all points of $(\Xi+\mathbb{Z})\backslash(\Xi-\mathbb{Z}_+)$. If $\bar{z}$ is such a discontinuity point then $\lim_{z\downarrow\bar{z}} Q(z) - Q(\bar{z}) = q^- \Pr\{\xi \in \bar{z} - \mathbb{Z}_+\}$.*

(ii) *Q is continuous from the right but not from the left in all points of $(\Xi-\mathbb{Z})\backslash(\Xi+\mathbb{Z}_+)$. If $\bar{z}$ is such a discontinuity point then $Q(\bar{z}) - \lim_{z\uparrow\bar{z}} Q(z) = -q^+ \Pr\{\xi \in \bar{z} + \mathbb{Z}_+\}$.*

(iii) *Q is neither continuous from the left nor continuous from the right in all points of $(\Xi + \mathbb{Z}_+) \cap (\Xi - \mathbb{Z}_+)$. If $\bar{z}$ is such a discontinuity point then $\lim_{z\downarrow\bar{z}} Q(z) - Q(\bar{z}) = q^- \Pr\{\xi \in \bar{z} - \mathbb{Z}_+\}$ and $Q(\bar{z}) - \lim_{z\uparrow\bar{z}} Q(z) = -q^+ \Pr\{\xi \in \bar{z} + \mathbb{Z}_+\}$. In particular this is true for all $\bar{z} \in \Xi$. In between two successive points in $\Xi + \mathbb{Z}$ the function Q is constant.*

(f) *Let Q^c be a convex function on $\mathbb{R}$ such that $\tilde{Q} \le Q^c \le \tilde{Q} + \max\{q^+, q^-\}$. Then*

$$W(s) = \frac{(Q^c)'_+(s) + q^+}{q^+ + q^-}, \qquad s \in \mathbb{R},$$

is a cdf, and for all $z \in \mathbb{R}$

$$Q^c(z) = q^+ E_\psi(\psi - z)^+ + q^- E_\psi(\psi - z)^- + \bar{c}.$$

Here, ψ is any random variable with cdf W, and $\bar{c} = (q^+c_2 + q^-c_1)(q^+ + q^-)$, where $c_1 = \lim_{z\to-\infty}(Q^c(z) - \tilde{Q}(z))$ and $c_2 = \lim_{z\to\infty}(Q^c(z) - \tilde{Q}(z))$. □

Note that part (f) implies that we can solve simple integer recourse problems (at least approximately) by algorithms developed for continuous simple recourse problems.

3 The convex hull

In this section we outline an algorithm that determines the convex hull of Q for the case that ξ follows a *discrete distribution with finitely many mass points.* For details we refer to [5, 6].

We assume here that ξ is a discrete random variable with support $\Xi = \{\xi^1, \ldots, \xi^p\}$, and also that the mass points are ordered, that is, $\xi^1 < \xi^2 < \ldots < \xi^p$. Then, by Theorem 2.1, the function Q is a finite, lower semicontinuous function that is discontinuous at all points of $\mathcal{D}_\infty = \bigcup_{i=1}^p \{\xi^i + \mathbb{Z}\}$. Moreover, Q is semi-periodic with period 1 on the intervals $(-\infty, \xi^1]$ and $[\xi^p, \infty)$, with slope $-q^+$ and q^-, respectively.

The crucial observation to make is that, due these properties, the convex hull of the function Q depends simply and solely on the points $(d, Q(d))$, $d \in \mathcal{D}_\infty$. It follows that *the convex hull of Q is equal to the convex hull of the piecewise linear function Q^{pl}*, specified by the points $(d, Q(d))$, $d \in \mathcal{D}_\infty$. The points where the slope of Q^{pl} changes will be called the knots.

Now we can apply an algorithm known as the Graham scan (see [3]), which determines the convex hull of a piecewise linear function f. Let f'_- and f'_+ denote the left and right derivative of f. Obviously, a necessary condition for a knot d^i to be on the convex hull is that $f'_-(d^i) \le f'_+(d^i)$. Consequently, if this condition is not satisfied then we may eliminate the knot d^i, and redefine the function f on $[d^{i-1}, d^{i+1}]$ as the linear function connecting the points $(d^{i-1}, f(d^{i-1}))$ and $(d^{i+1}, f(d^{i+1}))$. It is easy to see that by repeating this procedure until all knots satisfy the condition mentioned above, eventually we end up with the convex hull of the function f.

Of course, this algorithm only terminates in finite time if only a finite number of knots has to be to considered. For the function Q^{pl} finiteness of the initial set of knots is due to its

semi-periodicity on the intervals $(-\infty, \xi^1]$ and $[\xi^p, \infty)$, which implies that the convex hull Q^{**} is affine on $(-\infty, \xi^1 - 1]$ and $[\xi^p + 1, \infty)$ with slopes $-q^+$ and q^-, respectively. Thus, we only need to determine the convex hull of Q^{pl} restricted to $[\xi^1 - 1, \xi^p + 1]$ which depends only on knots in the set $\mathcal{D} = \mathcal{D}_\infty \cap [\xi^1 - 1, \xi^p + 1]$, which is finite indeed. It is straightforward to extend this function defined on $[\xi^1 - 1, \xi^p + 1]$ to the convex hull of Q^{pl} on $\mathbb{R}$.

Note that Graham's scan uses only one-sided derivatives. To compute these derivatives, we only need to know the difference between function values in neighboring discontinuity points of Q. Let $d^1 < d^2$ be any two such points. Then it follows from Theorem 2.1 (b) that

$$\begin{aligned} Q(d^2) - Q(d^1) &= -q^+ \sum_{i=1}^{p} \left\{ \Pr\{\xi = \xi^i\} : d^2 \in \xi^i - \mathbb{Z}_+ \right\} \\ &\quad + q^- \sum_{i=1}^{p} \left\{ \Pr\{\xi = \xi^i\} : d^1 \in \xi^i + \mathbb{Z}_+ \right\}. \end{aligned}$$

We see that computing the required function differences is merely a matter of book-keeping: no function values have to be calculated. For each discontinuity point we simply have to record by which mass point(s) it is generated, which comes at virtually no extra effort while constructing the set $\mathcal{D}$.

So far, Q^{**} is determined up to a constant. By Theorem 2.1 (f) this information is sufficient to determine the distribution of the random variable ψ that replaces ξ in the equivalent continuous simple recourse formulation. Since Q^{**} is piecewise linear, it follows that *ψ is a discrete random variable with mass points ψ^i that correspond to the knots of Q^{**}*. Moreover, $\Pr\{\psi = \psi^i\} = \delta^i/(q^+ + q^-)$, where δ^i is the increase of the slope of Q^{**} at ψ^i. It remains to compute the constant $\bar{c}$ as defined in Theorem 2.1 (f). We use that Q^{**} equals Q at each of its knots. Hence,

$$\bar{c} = Q(\psi^i) - \left(q^+ E_\psi(\psi - \psi^i)^+ + q^- E_\psi(\psi - \psi^i)^- \right),$$

where ψ^i is an arbitrary point in the support of ψ. This is the only time that we actually calculate a function value of Q.

An extended version of this algorithm, which uses that Q is semi-periodic on $[\xi^i, \xi^{i+1}]$, $i = 1, \ldots, p-1$, is strongly efficient (see [2]).

4 Convexity by perturbation of the distribution

In this section we consider convex approximations of Q for the case that ξ is a continuous random variable. The idea is to replace the distribution of ξ by a distribution such that the function Q is convex. A complete description of the class containing all distributions with this property is given in [7].

Definition 4.1 For $\alpha \in [0, 1)$ and F a cdf of a random variable, the *α-approximation* F_α of F is defined as the piecewise linear function generated by the restriction of F to the lattice $\alpha + \mathbb{Z}$. That is, for any $\bar{s} \in \alpha + \mathbb{Z}$,

$$F_\alpha(s) = F(\bar{s}) + (s - \bar{s}) \left(F(\bar{s} + 1) - F(\bar{s}) \right), \quad s \in [\bar{s}, \bar{s} + 1].$$

Since F_α is continuous and non-decreasing with $\lim_{s \to -\infty} F_\alpha(s) = 0$ and $\lim_{s \to \infty} F_\alpha(s) = 1$, it is a cdf itself. If ξ is a random variable with cdf F, any random variable ξ_α with cdf F_α will be called an α-approximation of ξ. Notice that ξ_α has a pdf even if ξ does not.

For the calculus with α-approximations we need the following generalization of the concepts of integer round down and integer round up.

Definition 4.2 For $s \in \mathbb{R}$ and $\alpha \in [0,1)$, the *round down* and the *round up* of s *with respect to* $\alpha + \mathbb{Z}$ is defined as, respectively,

$$\begin{aligned} \lfloor s \rfloor_\alpha &= \max\{\alpha + k : \alpha + k \le s,\ k \in \mathbb{Z}\} \\ \lceil s \rceil_\alpha &= \min\{\alpha + k : \alpha + k \ge s,\ k \in \mathbb{Z}\}, \end{aligned}$$

i.e., $\lfloor s \rfloor_\alpha = \lfloor s - \alpha \rfloor + \alpha$ and $\lceil s \rceil_\alpha = \lceil s - \alpha \rceil + \alpha$.

For example, $\lfloor 2.1 \rfloor_{0.9} = 1.9$ and $\lceil -3 \rceil_{0.6} = -2.4$. Notice that for all $s \in \mathbb{R}$, $\alpha \in [0,1)$, we have that $\lfloor s \rfloor_\alpha \le s \le \lceil s \rceil_\alpha$. In particular, if $s \in \alpha + \mathbb{Z}$ then $\lfloor s \rfloor_\alpha = s = \lceil s \rceil_\alpha$, and if $s \notin \alpha + \mathbb{Z}$ then $\lfloor s \rfloor_\alpha < s < \lceil s \rceil_\alpha = \lfloor s \rfloor_\alpha + 1$. Obviously, the usual integer round down and round up correspond to the case $\alpha = 0$.

The next lemma gives formulae for the α-approximation F_α of F and its pdf f_α.

Lemma 4.1 *Let F be the cdf of a random variable. For $\alpha \in [0,1)$, its α-approximation is the cdf F_α given by*

$$F_\alpha(s) = F(\lfloor s \rfloor_\alpha) + (s - \lfloor s \rfloor_\alpha)\left(F(\lceil s \rceil_\alpha) - F(\lfloor s \rfloor_\alpha)\right), \qquad s \in \mathbb{R}.$$

The right-continuous version f_α of the pdf of F_α is given by

$$f_\alpha(s) = F(\lfloor s \rfloor_\alpha + 1) - F(\lfloor s \rfloor_\alpha), \qquad s \in \mathbb{R}.$$

□

Below we discuss α-approximations Q_α of the function Q. We give a uniform bound on the error $|Q_\alpha - Q|$ for the case that the pdf f satisfies some mild regularity conditions, denoted as $f \in \mathcal{F}$. See [7] for details.

Theorem 4.1 *Let ξ be a continuous random variable with pdf f. For all $\alpha \in [0,1)$, let ξ_α be an α-approximation of ξ. For each $\alpha \in [0,1)$ define*

$$Q_\alpha(z) = q^+ E\lceil \xi_\alpha - z \rceil^+ + q^- E\lfloor \xi_\alpha - z \rfloor^-, \qquad z \in \mathbb{R},$$

where $q^+, q^- \ge 0$, $q^+ + q^- > 0$. Then

(a) For each α, the function Q_α is convex. It is related to the function $Q(z) = q^+ E\lceil \xi - z \rceil^+ + q^- E\lfloor \xi - z \rfloor^-$ by the equation

$$Q_\alpha(z) = Q(\lfloor z \rfloor_\alpha) + (z - \lfloor z \rfloor_\alpha)\left(Q(\lceil z \rceil_\alpha) - Q(\lfloor z \rfloor_\alpha)\right),$$

so that Q_α is the piecewise linear function that coincides with Q at all points $\alpha + k$, $k \in \mathbb{Z}$.

item[(b)] Assume that the pdf $f \in \mathcal{F}$. Then, for all $z \in \mathbb{R}$,

$$|Q_\alpha(z) - Q(z)| \le (q^+ + q^-)\frac{|\Delta| f}{4},$$

where $|\Delta| f$ denotes the total variation of f. In particular, if the pdf f is unimodal then

$$|Q_\alpha(z) - Q(z)| \le (q^+ + q^-)\frac{f(\nu)}{2} \quad \forall z \in \mathbb{R},$$

where ν is the mode of the distribution.

(c) *Let, for α and β different numbers in $[0,1)$, $\xi_{\alpha\beta}$ be the random variable that is equal to ξ_α or ξ_β, each with probability $1/2$. It has a piecewise constant pdf $f_{\alpha\beta}$ given by $f_{\alpha\beta}(s) = (f_\alpha(s) + f_\beta(s))/2$, $s \in \mathbb{R}$. Then $Q_{\alpha\beta}(z) = q^+ E\lceil \xi_{\alpha\beta} - z \rceil^+ + q^- E\lfloor \xi_{\alpha\beta} - z \rfloor^-$, $z \in \mathbb{R}$, is a piecewise linear convex function. If $f \in \mathcal{F}$ and in addition it holds $|\alpha - \beta| = 1/2$, then*

$$|Q_{\alpha\beta}(z) - Q(z)| \leq (q^+ + q^-)\,\frac{|\Delta| f}{8} \qquad \forall z \in \mathbb{R}.$$

Moreover, this uniform error bound can not be reduced by using other convex combinations of pdf of type f_α.

(d)

$$Q_\alpha(z) = q^+ E(\psi_\alpha - z)^+ + q^- E(\psi_\alpha - z)^- + \frac{q^+ q^-}{q^+ + q^-}, \quad z \in \mathbb{R},$$

where ψ_α is a random variable with cdf

$$W_\alpha(s) = \frac{q^+}{q^+ + q^-} F(\lfloor s \rfloor_\alpha) + \frac{q^-}{q^+ + q^-} F(\lfloor s \rfloor_\alpha + 1), \quad s \in \mathbb{R}.$$

The function $Q_{\alpha\beta}$ has a similar representation in terms of a random variable $\psi_{\alpha\beta}$ with cdf $W_{\alpha\beta} = (W_\alpha + W_\beta)/2$. □

We see that the function Q_α c.q. $Q_{\alpha\beta}$ is equal (up to a constant) to the one-dimensional expected value function of a continuous simple recourse problem, and that the distribution of its right-hand side random variable ψ_α c.q. $\psi_{\alpha\beta}$ can be computed directly from the distribution of ξ.

References

[1] Yu. Ermoliev and R.J-B. Wets. *Numerical Techniques for Stochastic Optimization.* Springer-Verlag, Berlin, 1988.

[2] M.R. Garey and D.S. Johnson. *Computers and Intractability; A Guide to the Theory of NP-Completeness.* Freeman, New York, 1979.

[3] R.L. Graham. An efficient algorithm for determining the convex hull of a finite planar set. *Information Processing Letters* **1**, 132–133, 1972.

[4] P. Kall and S.W. Wallace. *Stochastic Programming.* Wiley, Chichester, 1994.

[5] W.K. Klein Haneveld, L. Stougie, and M.H. van der Vlerk. On the convex hull of the simple integer recourse objective function. *Annals of Operations Research* **56**, 209–224, 1995.

[6] W.K. Klein Haneveld, L. Stougie, and M.H. van der Vlerk. An algorithm for the construction of convex hulls in simple integer recourse programming. *Annals of Operations Research* **64**, 67–81, 1996.

[7] W.K. Klein Haneveld, L. Stougie, and M.H. van der Vlerk. Convex simple integer recourse problems. Research Report 97A10, SOM, University of Groningen, 1997.

[8] F.V. Louveaux and M.H. van der Vlerk. Stochastic programming with simple integer recourse. *Mathematical Programming* **61**, 301–325, 1993.

[9] Yu. Nesterov and A. Nemirovski. *Interior point polynomial methods in Convex Programming: theory and applications.* SIAM Series in Applied Mathematics. SIAM, 1994.

[10] A. Prékopa. *Stochastic Programming.* Kluwer Academic Publishers, Dordrecht, 1995.

[11] R. Schultz, L. Stougie, and M.H. van der Vlerk. Two-stage stochastic integer programming: a survey. *Statistica Neerlandica* **50**(3), 404–416, 1996.

[12] L. Stougie and M.H. van der Vlerk. Stochastic integer programming. In S. Martello M. Dell'Amico, F. Maffioli, editor, *Annotated Bibliographies in Combinatorial Optimization*, chapter 8. Wiley, to appear.

[13] M.H. van der Vlerk. *Stochastic programming with integer recourse.* PhD thesis, University of Groningen, The Netherlands, 1995. (Electronic version via http://docserver.ub.rug.nl/eldoc/dis/eco/m.h.van.der.vlerk/dis.html)

[14] R.J-B. Wets. Solving stochastic programs with simple recourse. *Stochastics* **10**, 219–242, 1984.

Part VI

Some Facts

Review of the Education Activities of the LNMB

Lodewijk Kallenberg
Director LNMB
Groningen University
&
Leiden University
E-mail: kallenberg@rulwinw.leidenuniv.nl

1. General information

In 1986, the two-phase structure of academic education was introduced in the Netherlands. The first phase of the undergraduate education consists of a four year programme, followed by an optional second phase of another four years in which a student can take one's doctoral degree by writing a dissertation. During this second phase the PhD students are expected to attend graduate courses. The LNMB takes care of this education in the field of Operations Research. The graduate education programme consists of eight courses scheduled in two years, two annual workshops and approximately 16 colloquia a year.

In this review the following abbreviations are used:

CWI : Centre for Mathematics and Computer Science Amsterdam
EUR : Erasmus University Rotterdam
KUB : Tilburg University
KUN : Catholic University Nijmegen
LUW : Agricultural University of Wageningen
RUG : Groningen University
RUL : Leiden University
TUD : Delft University of Technology
TUE : Eindhoven University of Technology
UM : Maastricht University
UT : University of Twente
UU : Utrecht University
UvA : University of Amsterdam
VU : Free University Amsterdam

As of 1988 38 courses were given. These courses were attended by 228 students. On the average each student took 3.7 courses. Therefore, the total number of participations was 846. Below an overview is given of these 228 persons.

	Total	Dissertations[1]	PhD students[2]	Dropouts[3]	Others[4]
EUR	32	12	14	5	1
KUB	28	16	11	0	1
KUN	4	3	1	0	0
LUW	9	2	1	1	5
RUG	17	7	6	0	4
RUL	5	3	1	0	1
TUD	34	2	4	2	26
TUE	26	12	10	2	2
UM	17	8	6	0	3
UT	23	11	9	0	3
UvA	17	6	6	2	3
UU	6	1	1	0	4
VU	10	5	0	0	5
TOTAL	228	88	70	12	58

2. The courses

In each semester two courses are offered. Currently 38 courses have been given. The next summary shows the courses which were given in the period spring 1988 until spring 1997.

Spring 1988

S1: Stochastic Operations Research 1
Lecturers : Tijms (VU) and van der Duyn Schouten (KUB)
Participants : 22

C1: Combinatorial Optimalization 1
Lecturers : Kolen (UM) and Lenstra (TUE/CWI)
Participants : 21

Fall 1988

M1: Theory of Mathematical Programming
Lecturers : Klein Haneveld (RUG) and Ponstein (RUG)
Participants : 23

C2: Combinatorial Optimalization 2
Lecturers : Kolen (UM) and Schrijver (CWI/UvA)
Participants : 28

Spring 1989

M2: Techniques of Mathematical Programming
Lecturers : de Jong (TUE) and Lootsma (TUD)
Participants : 22

---

[1] Taken a doctoral degree.

[2] PhD students working on their dissertation.

[3] Students who stopped their PhD study.

[4] Other participants, e.g. twaio's (an education of two years), undergrade students or PhD students not in OR, but in a related area.

Spring 1989

G1: Game Theory
Lecturers : Tijs (KUB) and Vrieze (UM)
Participants : 21

Fall 1989

A1: Artificial Intelligence and Operations Research
Lecturers : Aarts (TUE), Hummel (New York) and van Hee (TUE)
Participants : 26

S1: Stochastic Operations Research 1
Lecturers : Tijms (VU) and van Dijk (VU)
Participants : 20

Spring 1990

S2: Stochastic Operations Research 2
Lecturers : van der Wal (TUE) and Wessels (TUE)
Participants : 17

C1: Combinatorial Optimalization 1
Lecturers : Kolen (UM) and Lenstra (TUE/CWI)
Participants : 20

Fall 1990

M1: Theory of Mathematical Programming
Lecturers : Evers (TUD), van Maaren (TUD) and Klein Haneveld (RUG)
Participants : 24

C2: Combinatorial Optimalization 2
Lecturers : Schrijver (CWI/UvA) and Stougie (UvA)
Participants : 24

Spring 1991

M2: Techniques of Mathematical Programming
Lecturers : de Jong (TUE) and Lootsma (TUD)
Participants : 22

G1: Game Theory
Lecturers : Tijs (KUB), van Damme (KUB) and Vrieze (UM)
Participants : 21

Fall 1991

A2a: Optimalization and Finance
Lecturer : Vorst (EUR)
Participants : 25

A2b: Simplicial Algorithms
Lecturer : Talman (KUB)
Participants : 25

S1: Stochastic Operations Research 1
Lecturers : Tijms (VU) and van der Duyn Schouten (KUB)
Participants : 27

Spring 1992

S2: Stochastic Operations Research 2
- Lecturers : van der Wal (TUE) and Boxma (CWI/KUB)
- Participants : 27

C1: Combinatorial Optimalization 1
- Lecturers : Kolen (UM) and Lenstra (TUE/CWI)
- Participants : 35

Fall 1992

M1: Theory of Mathematical Programming
- Lecturers : van Maaren (TUD) and Klein Haneveld (RUG)
- Participants : 20

C2: Combinatorial Optimalization 2
- Lecturers : Schrijver (CWI/UvA) and Stougie (UvA)
- Participants : 30

Spring 1993

M2: Techniques of Mathematical Programming
- Lecturers : Roos (TUD) and Lootsma (TUD)
- Participants : 20

G1: Game Theory
- Lecturers : Tijs (KUB), van Damme (KUB) and Vrieze (UM)
- Participants : 21

Fall 1993

S1: Stochastic Operations Research 1
- Lecturers : Tijms (VU) and Kallenberg (RUG/RUL)
- Participants : 21

C1: Combinatorial Optimalization 1
- Lecturers : Gerards (CWI) and Stougie (UvA)
- Participants : 19

Spring 1994

S2: Stochastic Operations Research 2
- Lecturers : Wessels (TUE) and Boxma (CWI/KUB)
- Participants: 21

C2: Combinatorial Optimalization 2
- Lecturers : Lenstra (TUE/CWI), van Hoesel (EUR) and Stougie (UvA)
- Participants : 27

Fall 1994

A3a Reliability Theory and Maintenance
- Lecturers : van der Duyn Schouten (KUB) and Dekker (EUR)
- Participants : 36

A3b: Production Management
- Lecturers : de Kok (TUE) and Zijm (UT)
- Participants : 36

Fall 1994

M1: Theory of Mathematical Programming
- Lecturers : van Maaren (TUD) and Klein Haneveld (RUG)
- Participants : 31

Spring 1995

M2: Capita Selecta Mathematical Programming
- Lecturers : Roos (TUD), Terlaky (TUD) and Lootsma (TUD)
- Participants : 13

G1: Game Theory
- Lecturers : Borm (KUB), Tijs (KUB), van Damme (KUB) and Vrieze (UM)
- Participants : 14

Fall 1995

S1: Stochastic Operations Research 1
- Lecturers : Tijms (VU) and Kallenberg (RUG/RUL)
- Participants : 17

C1: Combinatorial Optimalization 1
- Lecturers : Gerards (CWI) and Kern (TU)
- Participants : 19

Spring 1996

S2: Stochastic Operations Research 2
- Lecturers : Wessels (TUE) and Boxma (CWI/KUB)
- Participants : 15

C2: Combinatorial Optimalization 2
- Lecturers : Lenstra (TUE/CWI), van Hoesel (EUR) and Stougie (UvA)
- Participants : 18

Fall 1996

A4a: Randomized Algorithms
- Lecturer : Stougie (UvA)
- Participants : 31

A4b: Distribution and Logistics
- Lecturers : Lenstra (TUE) and Dekker (EUR)
- Participants : 33

M1: Theory of Mathematical Programming
- Lecturers : Balder (UU) and van Maaren (TUD)
- Participants : 25

Spring 1997

M2: Mathematical Programming 2
- Lecturers : Roos (TUD) and Terlaky (TUD)
- Participants : 18

G1: Game Theory
- Lecturers : Borm (KUB) and Vrieze (UM)
- Participants : 12

Number of PhD students which attended the courses (by university)

	1988	1989	1990	1991	1992	1993	1994	1995	1996	1997*	TOTAL
CWI	11	13	10	9	8	2	2	2	6	1	64
EUR	9	7	12	11	8	16	25	8	16	4	116
KUB	5	4	4	10	13	9	14	6	8	6	79
KUN	2	0	0	5	8	0	0	2	2	1	20
LUW	10	2	0	1	5	3	3	0	0	0	24
RUG	10	4	14	6	4	0	4	5	3	0	50
RUL	4	2	2	6	6	3	1	0	2	2	28
TUD	14	23	8	2	17	13	22	9	6	2	116
TUE	8	2	10	20	20	8	13	5	18	9	113
UM	8	8	6	5	6	4	9	15	7	0	68
UT	3	10	10	5	12	5	10	4	5	3	67
UU	0	0	0	1	0	5	3	0	3	0	12
UvA	3	5	7	2	5	7	6	7	10	1	53
VU	7	9	2	1	0	6	5	0	5	1	36
TOTAL	94	89	85	84	112	81	117	63	91	30	846

3. The workshops

Each year The LNMB organizes two workshops. The first workshop was held in December 1989. Below a survey is presented of the 18 workshops which were organized up until now with a list of the speakers and the topics of their papers.

Maastricht (December 1989)

Filar (University of Maryland)	: Game Theory
Kelly (University of Cambridge)	: Stochastic OR
Möhring (Technical University of Berlin)	: Combinatorial Optimization
Rockafellar (University of Washington)	: Optimization Theory

Groningen (May 1990)

Anthonisse (AKB-ORES)	: Case study OR
Boender (EUR)	: Case study OR
Dik (Volvo Car, Helmond)	: Case study OR
Hordijk (RUL)	: Markov decision theory
Kallenberg (RUL)	: Markov decision theory
van Laarhoven (CQM, Philips, Eindhoven)	: Case study OR
Telgen (UT)	: Case study OR
Vorst (EUR)	: Case study OR
Wijngaard (TUE)	: Case study OR

Amsterdam (December 1990)

Maschler (Hebrew University)	: Game Theory
Schweitzer (University of Rochester)	: Network of Queues
Wolsey (CORE)	: Polyhedral Combinatorics
Wets (University of California)	: Variational Problems

* only the spring courses

Eindhoven (June 1991)
The topic of this workshop was Local Search. The following lectures were given:

Aarts (Philips, Eindhoven/TUE)	: Neural Networks
Johnson (AT&T Bell Labs)	: Implementation Issues
Kern (UT)	: Probabilistic Analysis
Kolen (UM)	: Genetic Algorithms
van Laarhoven (CQM, Philips, Eindhoven)	: Simulated Annealing
Lenstra (TUE)	: Introduction
Papadimitriou (Mass. Institute of Technology)	: Computational Complexity
Savelsbergh (TUE)	: Local Search in Graphs

Lunteren (January 1992)

Grötschel (Konrad-Zuse Centre, Berlin)	: Combinatorial Optimization
Robinson (University of Wisconsin)	: Variational Problems
Stidham Jr (University of North Carolina)	: Control of Queues
Dyer (University of Leeds)	: Probabilistic Analysis
Fisher (University of Pennsylvania)	: Two OR applications in Industry
Gibbens (University of Cambridge, UK)	: Communication Networks
Rousseau (University of Montreal)	: Crew scheduling

Tilburg (May 1992)

Curiel (University of Maryland, Baltimore)	: Sequencing Games
van der Laan (VU)	: Bimatrix Games
Owen (Naval Postgrad. School, Monterey)	: Production Games
Peters (UM)	: Collective Decisions
Storcken (KUB)	: Collective Decisions
Talman (KUB)	: Simplicial Algorithms

Lunteren (December 1992)

Evstigneev (Academy of Sciences, Moscou)	: Stochastic Models
Goemans (Mass. Institute of Technology)	: LP relaxations for CO
Keeney (University of Southern California)	: Decision Analysis
Lemaréchal (INRIA, Le Chesnay)	: Convex Optimization
Liebling (Institute of Technology, Lausanne)	: Some Applications of OR
Steele (University of Pennsylvania)	: Probability and comput. problems
Weber (University of Cambridge, UK)	: Stochastic Scheduling

Delft (May 1993)

Ben-Tal (Technion, Haifa)	: Nonlinear programming
Grauer (University of Siegen)	: Distributed Computing
Kallenberg (RUG/RUL)	: Mean-variance in MDP's
Lootsma (TUD)	: Production Allocation
Nesterov (Academy of Sciences, Moscou)	: Decomposition Methods
Roos (TUD)	: Interior Point Methods
Terlaky (TUD)	: Implementation Issues

Lunteren (January 1994)

Bixby (Rice University/CPLEX)	: Computational State of the Art for LP
Coffman Jr. (AT&T Bell Labs, Murray Hill)	; Stochastic Matching Theory
Diaconis (Harvard University)	: Geometry and Markov Chains
Fourer (Northwestern University, Evanston)	: Design of Modeling Languages
Frank (Eötvös Loránd University, Budapest)	: Combinatorial Optimization
Jörnsten (Norwegian School of Economics)	: Applications in Oil Industry
Latouche (University of Brussels)	: Markovian Models
Peleg (Hebrew University of Jerusalem)	: Game Theory

During this conference there also was a Mini Symposium on "Modelling and Optimization" with contributions by Bixby, Fourer and Jörnsten.

Twente (May 1994)

Abate (AT&T Bell Labs, Warren)	: Queueing Theory
Blanc (KUB)	: Queueing Theory
Choudhury (AT&T Bell Labs, Holmdel)	: Queueing Theory
Kern (UT)	: Complexity of Cost Allocation
Neuts (University of Arizona, Tuczon)	: Queueing Theory
de Smit (UT)	: Queueing Theory
Turàn (University of Illinois, Chicago	: On-line scheduling
van de Velde (UT)	: On-line scheduling
Whitt (AT&T Bell Labs, Warren)	: Queueing Theory
Zijm (UT)	: Manufacturing Systems

Lunteren (January 1995)

Asmussen (Aalborg University)	: Stochastic Models
Cook (University of Bonn)	: Computational Aspects of CO
Cornuejols (Carnegie Mellon University)	: Balanced Matrices
Federgruen (Columbia University)	: Inventory Theory
Sobel (State University of New York)	: Structure in Stochastic Models
Wallace (University of Trondheim)	: Stochastic Programming

The Mini Symposium had as topic: "Management for Distribution and Vehicle-Routing". Contributions were given by Dekker (EUR), Federgruen (Columbia University), van der Ham (Europe Combined Terminals, Rotterdam) and Paixao (University of Lisboa).

Rotterdam (October 1995)

Axsäter (University of Lund)	: Inventory Control
Boender (EUR)	: Asset Liability Management
Dekker (EUR)	: Distribution Systems
Fleischmann (University of Augsburg)	: Freight Traffic Networks
de Kok (TUE)	: Installation Stock Models
Salomon (EUR)	: Production and Inventory
St"ahly (University of St. Gallen)	: Distribution of Dairy Products
Tüshaus (University of St. Gallen)	: Location Analysis
Vorst (EUR)	: Investment Horizons

Lunteren (Januari 1996)

Aarts (TUE)	: Neural Networks and CO
Kannan (Carnegie Mellon University)	: Learning Theory and OR

Shanthikumar (University of California)	: Dynamic Resource Allocation
Shwartz (Technion, Haifa)	: Large Deviations
Wagner (Universität Konstanz)	: Combinatorial Problems
Ye (University of Iowa)	: Interior Point Methods

There was a Mini-Symposium on "New Software Developments for Real-World Decision Problems". The lectures were given by Bisschops (Paragon Decision Technology, Haarlem) and Cornelissens (Beyers Innovative Software, Brasschaat).

Amsterdam (June 1996)

Fiat (Tel Aviv University)	: On-line problems
Fu (University of Maryland)	: Simulation and Perturbation
Leonardi (University of Roma)	: On-line Routing
Nobel (VU)	: Control of Queues
Ridder (VU)	: Simulation of Rare Events
Wöginger (Graz University of Technology)	: On-line Scheduling

Lunteren (Januari 1997)

Goldberg (NEC Research Institute, Princeton)	: HOT Priority Queues
Nemirowski (Technion, Haifa)	: IPM in Convex Programming
Papadimitriou (University of California)	: On-line Algorithms
Schmidt (University of Ulm)	: Queueing Theory
Sorin (Ecole Polytechnique, Paris)	: Repeated Games
Weiss (University of Haifa)	: Fluid Models for Queueing Networks

The Mini-Symposium on "OR for Financial Markets" was given by Dempster (University of Cambridge), Dert (ABN-AMRO Bank, Amsterdam) and Salomon (Rabo Bank, Zeist).

Leiden (June 1997)

Altman (INRIA, Sophia-Antipolis)	: Multimodularity and Sequential Optimization
Feltkamp (University of Alicante, Spain)	: Combinatorial Games
Gaujal (INRIA, Sophia-Antipolis)	: Multimodularity and Sequential Optimization
Hamers (KUB)	: Combinatorial Games
Hordijk (RUL)	: Multimodularity and Sequential Optimization
Kallenberg (RUL)	: Markov Decision Problems
Koole (VU)	: Control of Queuing Systems
Puterman (University of British Columbia)	: Markov Decision Problems
Thomas (University of Edinburgh)	: Parallel Computations for Markov Decision Problems
Vrieze (UM)	: Competitive Markov Decision Problems

4. Dissertations

Below the dissertations written by the PhD students of the LNMB are listed distinguished by university. Together with the title of the doctoral thesis, the candidate's name, the date, and the promotor(s) are mentioned.

Agricultural University Wageningen (LUW) (2 dissertations)

BLOEMHOF-RUWAARD, JACQUELINE (03-06-96), Integration of Operational Research and Environment (van Beek/ van Wassenhove).

HUIRNE, RUUD (18-12-90), Computerized management support for swine breeding farms (van Beek)

Eindhoven University of Technology (TUE) (12 dissertations)

ADAN, IVO (19-11-91), A compensation approach for queueing problems (Wessels/Zijm).

AKKER, MARJAN VAN DEN (21-12-94), LP-based solution methods for single-machine scheduling problems (Lenstra).

EENIGE, MICHÈL VAN (17-09-96), Queueing systems with periodic service (Wessels).

EIJL, CLEOLA VAN (14-06-96), A polyhedral approach to the discrete lot-sizing and scheduling problem (Lenstra).

HOOGEVEEN, HAN (07-02-92), Single-machine bicriteria scheduling (Lenstra)

HOUTUM, GEERT-JAN VAN (07-02-95), New approaches for multi-dimensional queueing systems (Wessels/Zijm).

SOL, MARC (08-11-94), Column generation techniques for pickup and delivery problems (Lenstra).

VAESSENS, ROB (22-09-95), Generalized job-shop scheduling: complexity and local search (Lenstra/Aarts).

VELTMAN, BART (18-05-93), Multiprocessor scheduling with communication delays (Lenstra).

VERRIJDT, JOS (30-01-97), Design and control of service part distribution systems (de Kok).

WENNINK, MARC (15-09-95), Algorithmic support for automated planning boards (Lenstra).

ZWIETERING, PIETER (15-04-94), The complexity of multi-layered perceptrons (Wessels/Aarts).

Erasmus University Rotterdam (EUR) (12 dissertations)

BARROS, ISABELLE DE (26-01-95), Discrete and fractional programming techniques for location models (Rinnooy Kan).

CHEN, BO (09-06-94), Worst case performance of scheduling heuristics (van Wassenhove).

DERT, CEES (30-11-95), Asset Liability Management for Pension Funds; a multi-stage chance constrained programming approach (Rinnooy Kan/Boender).

GROMICHO, JOAQIUM (26-01-95), Quasiconvex optimization and location models (Rinnooy Kan).

MELO, TERESA (17-10-96), Stochastic Lot-Sizing in Production Planning (Dekker).

ROMEIJN, EDWIN (03-09-92), Global optimization by random walk sampling methods (Rinnooy Kan/Boender).

SMITH, MARCEL (10-04-97), On the availability of failure prone systems (Dekker).

TIELEMANS, PETER (28-03-96), Lead Time Performance in Manufacturing Systems (van Nunen).

VERBEEK, PETER (30-05-91), Learning about decision support systems: two case studies on manpower planning in an airline (Kolen/Rinnooy Kan).

VLIET, ANDRÉ VAN (16-02-95), Lower and Upper Bounds for On-line Packing and Scheduling Heuristics (Rinnooy Kan).

VLIET, MARIO VAN (06-06-91), Optimization of manufacturing system design (Rinnooy Kan/Boxma).

WILDEMAN, RALPH (10-10-96), The Art of Grouping Maintenance (Dekker).

Free University Amsterdam (VU) (5 dissertations)

BOUCHERIE, RICHARD (21-05-92), Product-Form in Queueing Networks (Van Dijk/Tijms).

GOUWELEEUW, FRANK (28-11-96), A General Approach to Computing Loss Probabilities (Tijms).

MANDJES, MICHEL (03-12-96), Rare Event Analysis of Communication Networks (Tijms).

OMMEREN, JAN-KEES (15-06-89), Asymptotic analysis of queueing systems (Tijms).

SMEITINK, ERIC (16-04-92), Stochastic models for repairable systems (Tijms).

Leiden University (RUL) (3 dissertations)

KOOLE, GER (24-06-92), Stochastic Scheduling and Dynamic Programming (Hordijk).

LOEVE, ANNEKE (22-03-95), Markov decision chains with partial information (Hordijk).

PASSCHIER, OLAF (13-11-96), The theory of Markov games and queueing control (Hordijk).

Nijmegen University (KUN) (3 dissertations)

JURG, PETER (10-11-93), Some topics in the theory of bimatrix games (Tijs).

REIJNIERSE, HANS (12-06-95), Games, graphs and algorithms (Tijs).

VERMEULEN, DRIES (05-08-95), Stability in Non-Cooperative Game Theory (Tijs).

Tilburg University (KUB) (16 dissertations)

BORST, SEM (04-11-94), Polling systems (Boxma).

COMBÉ, MARCO (27-01-95), Queueing models with dependence structures (Boxma).

DAM, EDWIN VAN (4-19-96), Graphs with Few Eigenvalues (Tijs).

DANG, CHUANGYIN (15-03-91), The D_1 triangulation in simplicial algorithms (Talman).

EIJS, MARC VAN (08-10-93), Coordinated replenishment policies (van der Duyn Schouten).

FELTKAMP, VINCENT (04-05-95), Cooperation in Controlled Network Structures (Tijs).

HAMERS, HERBERT (29-09-95), Sequencing and Delivery Situations (Tijs).

HOUT, WILBERT VAN DEN (27-03-96), The Power-Series Algorithm (Boxma).

MEI, ROB VAN DER (07-04-95), Polling Systems and the Power-Series Algorithm (Boxma).

NOUWELAND, ANNE VAN DE (05-02-93), Games and graphs in economic situations (Tijs).

OTTEN, GERT-JAN (24-03-95), On decision making in cooperative situations (Tijs).

PEETERS, RENÉ (09-06-95), Ranks and structure of graphs (Tijs).

VANNESTE, STEPHAN (07-12-92), Maintenance policies for complex systems (van der Duyn Schouten).

WARTENHORST, PIETER (07-02-93), Performance Analysis of Repairable Systems (van der Duyn Schouten).

WESTSTRATE, JAN (27-02-92), Analysis and optimization of polling models (Boxma).

YANG, ZAIFU (12-01-96), Simplicial fixed point algorithms and applications (Talman).

Twente University (UT) (11 dissertations)

AARTS, HARRY (11-11-94), Minimum cost spanning tree games and set games (Hoede).

BERG,YEROON VAN DEN (20-12-96), Planning and control of warehousing systems (Zijm).

GADEMANN, NOUD (04-02-93), Linear optimization in random polynomial time (Faigle).

GARBE, RENATE (06-10-94), Algorithmic aspects of interval orders (Faigle).

JACOBS, ALBERT (30-11-93), Timespan minimization for storm surge barrier (van Harten/Heemink).

MEESTER, GEATSE (02-02-96), Multi-recourse shop floor scheduling (Zijm).

SCHRIJNER, PAULINE (22-09-95) Quasi-stationarity of discrete-time Markov chains (de Smit/van Doorn).

SCHUTTEN, MARCO (20-06-96), Shop floor scheduling with setup times (Zijm).

SPIEKER, BORIS (27-05-94), The power of rank bounds in communication complexity (Faigle).

WORM, JACQUES (08-04-94), Model based decision support for multi-period road maintenance planning (van Harten).

ZEE, DURK-JOUKE VAN DER (17-01-97), Simulation as a tool for logistic management (van Harten).

University of Amsterdam (UvA) (6 dissertations)

COELHO DE PINA, JOSÉ (19-12-95), Applications of shortest path methods (Schrijver).

GRAAF, MAURITS DE (23-06-94), Graphs and curves on surfaces (Schrijver).

HOLST, HEIN VAN DER (24-09-96), Topological and spectral graph characterizations (Schrijver).

KORT, JEROEN DE (17-12-92), Edge-disjointness in combinatorial optimization (de Leve).

PIERSMA, NANDA (25-06-93), Combinatorial optimization and empirical processes (Steele/de Leve).

SLUIS, ERIK VAN DER (07-09-93), Heuristics for complex inventory systems. Deterministic and stochastic problems (de Leve).

University of Delft (TUD) (2 dissertations)

HERTOG, DICK DEN (15-09-92), Interior point approach to linear, quadratic and convex programming: algorithms and complexity (Lootsma/Roos).

JANSEN, BENJAMIN (08-01-96), Interior point techniques in optimization: complementarity, sensitivity and algorithms (Lootsma/Roos).

University of Groningen (RUG) (7 dissertations)

BAKKER, EVERT-JAN (30-01-92), Rainfall and risk in India's agriculture; an ex-ante evaluation of rainfall insurance (Schweigman).

BOORSMA, ANNE (17-05-90), Mogelijkheden voor akkerbouwbedrijven in de veenkoloniën (Schweigman).

BROENS, DOUWE-FRITS (26-01-95), Investment Evaluation with respect to Commercial Uncertainty (Klein Haneveld).

DAL, RENÉ VAN (03-09-92), Special cases of the traveling salesman problem (Sierksma/Ponstein).

DAM, PETER VAN (21-09-95), Scheduling packaging lines in the process industry (Sierksma/Gaalman).

VEEN, JACK VAN DER (24-09-92), Solvable cases of the traveling salesman with various objection functions (Sierksma/Klein Haneveld).

VLERK, MAARTEN VAN DER (13-04-95), Stochastic Programming with Integer Recourse (Klein Haneveld).

University of Maastricht (UM) (8 dissertations)

JANSSEN, MARCO (29-11-96), Meeting targets: tools to support integrated assessment modeling of global change (Vrieze).

JOOSTEN, REINOUD (12-12-96), Dynamics, equilibria and values (Peters/Vrieze).

KLUNDERT, JORIS VAN DER (07-06-96), Scheduling Problems In Automated Manufacturing (Kolen/Crama).

KUIPERS, JEROEN (15-06-94), Combinatorial methods in cooperative game theory (Vrieze).

OERLEMANS, ALWIN (06-12-96), Production planning for flexible manufacturing systems (Kolen).

OOSTEN, MAARTEN (06-12-96), A Polyhedral Approach to Grouping Problems (Kolen/Vrieze).

SPIEKSMA, FRITS (21-02-92), Assignment and scheduling algorithms in automated manufacturing (Vrieze).

THUIJSMAN, FRANK (23-06-89), Optimality and Equilibria in Stochastic Games (Vrieze/Tijs).

Utrecht University (UU) (1 dissertation)

FLUITER, BABETTE DE (19-03-97), Algorithms for graphs of small treewidth (van Leeuwen).

Summary (by year and by university)

	1991	1992	1993	1994	1995	1996	1997	TOTAL
EUR	2	1	0	1	4	3	1	12
KUB	1	2	3	1	6	3	0	16
KUN	0	0	1	0	2	0	0	3
LUW	1	0	0	0	0	1	0	2
RUG	1	3	0	0	3	0	0	7
RUL	0	1	0	0	1	1	0	3
TUD	0	1	0	0	0	1	0	2
TUE	1	1	1	3	3	2	1	12
UM	1	1	0	1	0	5	0	8
UT	0	0	2	4	1	3	1	11
UvA	0	1	2	1	1	1	0	6
UU	0	0	0	0	0	0	1	1
VU	1	2	0	0	0	2	0	5
TOTAL	8	13	9	11	21	22	4	88

CWI TRACTS

1 D.H.J. Epema. *Surfaces with canonical hyperplane sections.* 1984.
2 J.J. Dijkstra. *Fake topological Hilbert spaces and characterizations of dimension in terms of negligibility.* 1984.
3 A.J. van der Schaft. *System theoretic descriptions of physical systems.* 1984.
4 J. Koene. *Minimal cost flow in processing networks, a primal approach.* 1984.
5 B. Hoogenboom. *Intertwining functions on compact Lie groups.* 1984.
6 A.P.W. Böhm. *Dataflow computation.* 1984.
7 A. Blokhuis. *Few-distance sets.* 1984.
8 M.H. van Hoorn. *Algorithms and approximations for queueing systems.* 1984.
9 C.P.J. Koymans. *Models of the lambda calculus.* 1984.
10 C.G. van der Laan, N.M. Temme. *Calculation of special functions: the gamma function, the exponential integrals and error-like functions.* 1984.
11 N.M. van Dijk. *Controlled Markov processes; time-discretization.* 1984.
12 W.H. Hundsdorfer. *The numerical solution of nonlinear stiff initial value problems: an analysis of one step methods.* 1985.
13 D. Grune. *On the design of ALEPH.* 1985.
14 J.G.F. Thiemann. *Analytic spaces and dynamic programming: a measure theoretic approach.* 1985.
15 F.J. van der Linden. *Euclidean rings with two infinite primes.* 1985.
16 R.J.P. Groothuizen. *Mixed elliptic-hyperbolic partial differential operators: a case-study in Fourier integral operators.* 1985.
17 H.M.M. ten Eikelder. *Symmetries for dynamical and Hamiltonian systems.* 1985.
18 A.D.M. Kester. *Some large deviation results in statistics.* 1985.
19 T.M.V. Janssen. *Foundations and applications of Montague grammar, part 1: Philosophy, framework, computer science.* 1986.
20 B.F. Schriever. *Order dependence.* 1986.
21 D.P. van der Vecht. *Inequalities for stopped Brownian motion.* 1986.
22 J.C.S.P. van der Woude. *Topological dynamix.* 1986.
23 A.F. Monna. *Methods, concepts and ideas in mathematics: aspects of an evolution.* 1986.
24 J.C.M. Baeten. *Filters and ultrafilters over definable subsets of admissible ordinals.* 1986.
25 A.W.J. Kolen. *Tree network and planar rectilinear location theory.* 1986.
26 A.H. Veen. *The misconstrued semicolon: Reconciling imperative languages and dataflow machines.* 1986.
27 A.J.M. van Engelen. *Homogeneous zero-dimensional absolute Borel sets.* 1986.
28 T.M.V. Janssen. *Foundations and applications of Montague grammar, part 2: Applications to natural language.* 1986.
29 H.L. Trentelman. *Almost invariant subspaces and high gain feedback.* 1986.
30 A.G. de Kok. *Production-inventory control models: approximations and algorithms.* 1987.
31 E.E.M. van Berkum. *Optimal paired comparison designs for factorial experiments.* 1987.
32 J.H.J. Einmahl. *Multivariate empirical processes.* 1987.
33 O.J. Vrieze. *Stochastic games with finite state and action spaces.* 1987.
34 P.H.M. Kersten. *Infinitesimal symmetries: a computational approach.* 1987.
35 M.L. Eaton. *Lectures on topics in probability inequalities.* 1987.
36 A.H.P. van der Burgh, R.M.M. Mattheij (eds.). *Proceedings of the first international conference on industrial and applied mathematics (ICIAM 87).* 1987.
37 L. Stougie. *Design and analysis of algorithms for stochastic integer programming.* 1987.
38 J.B.G. Frenk. *On Banach algebras, renewal measures and regenerative processes.* 1987.
39 H.J.M. Peters, O.J. Vrieze (eds.). *Surveys in game theory and related topics.* 1987.
40 J.L. Geluk, L. de Haan. *Regular variation, extensions and Tauberian theorems.* 1987.
41 Sape J. Mullender (ed.). *The Amoeba distributed operating system: Selected papers 1984-1987.* 1987.
42 P.R.J. Asveld, A. Nijholt (eds.). *Essays on concepts, formalisms, and tools.* 1987.
43 H.L. Bodlaender. *Distributed computing: structure and complexity.* 1987.
44 A.W. van der Vaart. *Statistical estimation in large parameter spaces.* 1988.
45 S.A. van de Geer. *Regression analysis and empirical processes.* 1988.
46 S.P. Spekreijse. *Multigrid solution of the steady Euler equations.* 1988.
47 J.B. Dijkstra. *Analysis of means in some non-standard situations.* 1988.
48 F.C. Drost. *Asymptotics for generalized chi-square goodness-of-fit tests.* 1988.
49 F.W. Wubs. *Numerical solution of the shallow-water equations.* 1988.
50 F. de Kerf. *Asymptotic analysis of a class of perturbed Korteweg-de Vries initial value problems.* 1988.
51 P.J.M. van Laarhoven. *Theoretical and computational aspects of simulated annealing.* 1988.
52 P.M. van Loon. *Continuous decoupling transformations for linear boundary value problems.* 1988.
53 K.C.P. Machielsen. *Numerical solution of optimal control problems with state constraints by sequential quadratic programming in function space.* 1988.
54 L.C.R.J. Willenborg. *Computational aspects of survey data processing.* 1988.
55 G.J. van der Steen. *A program generator for recognition, parsing and transduction with syntactic patterns.* 1988.
56 J.C. Ebergen. *Translating programs into delay-insensitive circuits.* 1989.
57 S.M. Verduyn Lunel. *Exponential type calculus for linear delay equations.* 1989.
58 M.C.M. de Gunst. *A random model for plant cell population growth.* 1989.
59 D. van Dulst. *Characterizations of Banach spaces not containing l^1.* 1989.
60 H.E. de Swart. *Vacillation and predictability properties of low-order atmospheric spectral models.* 1989.

61 P. de Jong. *Central limit theorems for generalized multilinear forms.* 1989.
62 V.J. de Jong. *A specification system for statistical software.* 1989.
63 B. Hanzon. *Identifiability, recursive identification and spaces of linear dynamical systems, part I.* 1989.
64 B. Hanzon. *Identifiability, recursive identification and spaces of linear dynamical systems, part II.* 1989.
65 B.M.M. de Weger. *Algorithms for diophantine equations.* 1989.
66 A. Jung. *Cartesian closed categories of domains.* 1989.
67 J.W. Polderman. *Adaptive control & identification: Conflict or conflux?.* 1989.
68 H.J. Woerdeman. *Matrix and operator extensions.* 1989.
69 B.G. Hansen. *Monotonicity properties of infinitely divisible distributions.* 1989.
70 J.K. Lenstra, H.C. Tijms, A. Volgenant (eds.). *Twenty-five years of operations research in the Netherlands: Papers dedicated to Gijs de Leve.* 1990.
71 P.J.C. Spreij. *Counting process systems. Identification and stochastic realization.* 1990.
72 J.F. Kaashoek. *Modeling one dimensional pattern formation by anti-diffusion.* 1990.
73 A.M.H. Gerards. *Graphs and polyhedra. Binary spaces and cutting planes.* 1990.
74 B. Koren. *Multigrid and defect correction for the steady Navier-Stokes equations. Application to aerodynamics.* 1991.
75 M.W.P. Savelsbergh. *Computer aided routing.* 1992.
76 O.E. Flippo. *Stability, duality and decomposition in general mathematical programming.* 1991.
77 A.J. van Es. *Aspects of nonparametric density estimation.* 1991.
78 G.A.P. Kindervater. *Exercises in parallel combinatorial computing.* 1992.
79 J.J. Lodder. *Towards a symmetrical theory of generalized functions.* 1991.
80 S.A. Smulders. *Control of freeway traffic flow.* 1996.
81 P.H.M. America, J.J.M.M. Rutten. *A parallel object-oriented language: design and semantic foundations.* 1992.
82 F. Thuijsman. *Optimality and equilibria in stochastic games.* 1992.
83 R.J. Kooman. *Convergence properties of recurrence sequences.* 1992.
84 A.M. Cohen (ed.). *Computational aspects of Lie group representations and related topics. Proceedings of the 1990 Computational Algebra Seminar at CWI, Amsterdam.* 1991.
85 V. de Valk. *One-dependent processes.* 1994.
86 J.A. Baars, J.A.M. de Groot. *On topological and linear equivalence of certain function spaces.* 1992.
87 A.F. Monna. *The way of mathematics and mathematicians.* 1992.
88 E.D. de Goede. *Numerical methods for the three-dimensional shallow water equations.* 1993.
89 M. Zwaan. *Moment problems in Hilbert space with applications to magnetic resonance imaging.* 1993.
90 C. Vuik. *The solution of a one-dimensional Stefan problem.* 1993.
91 E.R. Verheul. *Multimedians in metric and normed spaces.* 1993.
92 J.L.M. Maubach. *Iterative methods for non-linear partial differential equations.* 1994.
93 A.W. Ambergen. *Statistical uncertainties in posterior probabilities.* 1993.
94 P.A. Zegeling. *Moving-grid methods for time-dependent partial differential equations.* 1993.
95 M.J.C. van Pul. *Statistical analysis of software reliability models.* 1993.
96 J.K. Scholma. *A Lie algebraic study of some integrable systems associated with root systems.* 1993.
97 J.L. van den Berg. *Sojourn times in feedback and processor sharing queues.* 1993.
98 A.J. Koning. *Stochastic integrals and goodness-of-fit tests.* 1993.
99 B.P. Sommeijer. *Parallelism in the numerical integration of initial value problems.* 1993.
100 J. Molenaar. *Multigrid methods for semiconductor device simulation.* 1993.
101 H.J.C. Huijberts. *Dynamic feedback in nonlinear synthesis problems.* 1994.
102 J.A.M. van der Weide. *Stochastic processes and point processes of excursions.* 1994.
103 P.W. Hemker, P. Wesseling (eds.). *Contributions to multigrid.* 1994.
104 I.J.B.F. Adan. *A compensation approach for queueing problems.* 1994.
105 O.J. Boxma, G.M. Koole (eds.). *Performance evaluation of parallel and distributed systems - solution methods. Part 1.* 1994.
106 O.J. Boxma, G.M. Koole (eds.). *Performance evaluation of parallel and distributed systems - solution methods. Part 2.* 1994.
107 R.A. Trompert. *Local uniform grid refinement for time-dependent partial differential equations.* 1995.
108 M.N.M. van Lieshout. *Stochastic geometry models in image analysis and spatial statistics.* 1995.
109 R.J. van Glabbeek. *Comparative concurrency semantics and refinement of actions.* 1996.
110 W. Vervaat, H. Holwerda (ed.). *Probability and lattices.* 1997.
111 I. Helsloot. *Covariant formal group theory and some applications.* 1995.
112 R.N. Bol. *Loop checking in logic programming.* 1995.
113 G.J.M. Koole. *Stochastic scheduling and dynamic programming.* 1995.
114 M.J. van der Laan. *Efficient and inefficient estimation in semiparametric models.* 1995.
115 S.C. Borst. *Polling models.* 1996.
116 G.D. Otten. *Statistical test limits in quality control.* 1996.
117 K.G. Langendoen. *Graph reduction on shared-memory multiprocessors.* 1996.
118 W.C.A. Maas. *Nonlinear $\mathcal{H}_\infty$ control: the singular case.* 1996.
119 A. Di Bucchianico. *Probabilistic and analytical aspects of the umbral calculus.* 1997.
120 M. van Loon. *Numerical methods in smog prediction.* 1997.
121 B.J. Wijers. *Nonparametric estimation for a windowed line-segment process.* 1997.
122 W.K. Klein Haneveld, O.J. Vrieze, L.C.M. Kallenberg (editors). *Ten years LNMB – Ph.D. research and graduate courses of the Dutch Network of Operations Research.* 1997.

MATHEMATICAL CENTRE TRACTS

1 T. van der Walt. *Fixed and almost fixed points.* 1963.

2 A.R. Bloemena. *Sampling from a graph.* 1964.

3 G. de Leve. *Generalized Markovian decision processes, part I: model and method.* 1964.

4 G. de Leve. *Generalized Markovian decision processes, part II: probabilistic background.* 1964.

5 G. de Leve, H.C. Tijms, P.J. Weeda. *Generalized Markovian decision processes, applications.* 1970.

6 M.A. Maurice. *Compact ordered spaces.* 1964.

7 W.R. van Zwet. *Convex transformations of random variables.* 1964.

8 J.A. Zonneveld. *Automatic numerical integration.* 1964.

9 P.C. Baayen. *Universal morphisms.* 1964.

10 E.M. de Jager. *Applications of distributions in mathematical physics.* 1964.

11 A.B. Paalman-de Miranda. *Topological semigroups.* 1964.

12 J.A.Th.M. van Berckel, H. Brandt Corstius, R.J. Mokken, A. van Wijngaarden. *Formal properties of newspaper Dutch.* 1965.

13 H.A. Lauwerier. *Asymptotic expansions.* 1966, out of print: replaced by MCT 54.

14 H.A. Lauwerier. *Calculus of variations in mathematical physics.* 1966.

15 R. Doornbos. *Slippage tests.* 1966.

16 J.W. de Bakker. *Formal definition of programming languages with an application to the definition of ALGOL 60.* 1967.

17 R.P. van de Riet. *Formula manipulation in ALGOL 60, part 1.* 1968.

18 R.P. van de Riet. *Formula manipulation in ALGOL 60, part 2.* 1968.

19 J. van der Slot. *Some properties related to compactness.* 1968.

20 P.J. van der Houwen. *Finite difference methods for solving partial differential equations.* 1968.

21 E. Wattel. *The compactness operator in set theory and topology.* 1968.

22 T.J. Dekker. *ALGOL 60 procedures in numerical algebra, part 1.* 1968.

23 T.J. Dekker, W. Hoffmann. *ALGOL 60 procedures in numerical algebra, part 2.* 1968.

24 J.W. de Bakker. *Recursive procedures.* 1971.

25 E.R. Paërl. *Representations of the Lorentz group and projective geometry.* 1969.

26 European Meeting 1968. *Selected statistical papers, part I.* 1968.

27 European Meeting 1968. *Selected statistical papers, part II.* 1968.

28 J. Oosterhoff. *Combination of one-sided statistical tests.* 1969.

29 J. Verhoeff. *Error detecting decimal codes.* 1969.

30 H. Brandt Corstius. *Exercises in computational linguistics.* 1970.

31 W. Molenaar. *Approximations to the Poisson, binomial and hypergeometric distribution functions.* 1970.

32 L. de Haan. *On regular variation and its application to the weak convergence of sample extremes.* 1970.

33 F.W. Steutel. *Preservations of infinite divisibility under mixing and related topics.* 1970.

34 I. Juhász, A. Verbeek, N.S. Kroonenberg. *Cardinal functions in topology.* 1971.

35 M.H. van Emden. *An analysis of complexity.* 1971.

36 J. Grasman. *On the birth of boundary layers.* 1971.

37 J.W. de Bakker, G.A. Blaauw, A.J.W. Duijvestijn, E.W. Dijkstra, P.J. van der Houwen, G.A.M. Kamsteeg-Kemper, F.E.J. Kruseman Aretz, W.L. van der Poel, J.P. Schaap-Kruseman, M.V. Wilkes, G. Zoutendijk. *MC-25 Informatica Symposium.* 1971.

38 W.A. Verloren van Themaat. *Automatic analysis of Dutch compound words.* 1972.

39 H. Bavinck. *Jacobi series and approximation.* 1972.

40 H.C. Tijms. *Analysis of (s,S) inventory models.* 1972.

41 A. Verbeek. *Superextensions of topological spaces.* 1972.

42 W. Vervaat. *Success epochs in Bernoulli trials (with applications in number theory).* 1972.

43 F.H. Ruymgaart. *Asymptotic theory of rank tests for independence.* 1973.

44 H. Bart. *Meromorphic operator valued functions.* 1973.

45 A.A. Balkema. *Monotone transformations and limit laws.* 1973.

46 R.P. van de Riet. *ABC ALGOL, a portable language for formula manipulation systems, part 1: the language.* 1973.

47 R.P. van de Riet. *ABC ALGOL, a portable language for formula manipulation systems, part 2: the compiler.* 1973.

48 F.E.J. Kruseman Aretz, P.J.W. ten Hagen, H.L. Oudshoorn. *An ALGOL 60 compiler in ALGOL 60, text of the MC-compiler for the EL-X8.* 1973.

49 H. Kok. *Connected orderable spaces.* 1974.

50 A. van Wijngaarden, B.J. Mailloux, J.E.L. Peck, C.H.A. Koster, M. Sintzoff, C.H. Lindsey, L.G.L.T. Meertens, R.G. Fisker (eds.). *Revised report on the algorithmic language ALGOL 68.* 1976.

51 A. Hordijk. *Dynamic programming and Markov potential theory.* 1974.

52 P.C. Baayen (ed.). *Topological structures.* 1974.

53 M.J. Faber. *Metrizability in generalized ordered spaces.* 1974.

54 H.A. Lauwerier. *Asymptotic analysis, part 1.* 1974.

55 M. Hall, Jr., J.H. van Lint (eds.). *Combinatorics, part 1: theory of designs, finite geometry and coding theory.* 1974.

56 M. Hall, Jr., J.H. van Lint (eds.). *Combinatorics, part 2: graph theory, foundations, partitions and combinatorial geometry.* 1974.

57 M. Hall, Jr., J.H. van Lint (eds.). *Combinatorics, part 3: combinatorial group theory.* 1974.

58 W. Albers. *Asymptotic expansions and the deficiency concept in statistics.* 1975.

59 J.L. Mijnheer. *Sample path properties of stable processes.* 1975.

60 F. Göbel. *Queueing models involving buffers.* 1975.

63 J.W. de Bakker (ed.). *Foundations of computer science.* 1975.

64 W.J. de Schipper. *Symmetric closed categories.* 1975.

65 J. de Vries. *Topological transformation groups, 1: a categorical approach.* 1975.

66 H.G.J. Pijls. *Logically convex algebras in spectral theory and eigenfunction expansions.* 1976.

68 P.P.N. de Groen. *Singularly perturbed differential operators of second order.* 1976.

69 J.K. Lenstra. *Sequencing by enumerative methods.* 1977.

70 W.P. de Roever, Jr. *Recursive program schemes: semantics and proof theory.* 1976.

71 J.A.E.E. van Nunen. *Contracting Markov decision processes.* 1976.

72 J.K.M. Jansen. *Simple periodic and non-periodic Lamé functions and their applications in the theory of conical waveguides.* 1977.

73 D.M.R. Leivant. *Absoluteness of intuitionistic logic.* 1979.

74 H.J.J. te Riele. *A theoretical and computational study of generalized aliquot sequences.* 1976.

75 A.E. Brouwer. *Treelike spaces and related connected topological spaces.* 1977.

76 M. Rem. *Associons and the closure statements.* 1976.

77 W.C.M. Kallenberg. *Asymptotic optimality of likelihood ratio tests in exponential families.* 1978.

78 E. de Jonge, A.C.M. van Rooij. *Introduction to Riesz spaces.* 1977.

79 M.C.A. van Zuijlen. *Empirical distributions and rank statistics.* 1977.

80 P.W. Hemker. *A numerical study of stiff two-point boundary problems.* 1977.

81 K.R. Apt, J.W. de Bakker (eds.). *Foundations of computer science II, part 1.* 1976.

82 K.R. Apt, J.W. de Bakker (eds.). *Foundations of computer science II, part 2.* 1976.

83 L.S. van Benthem Jutting. *Checking Landau's "Grundlagen" in the AUTOMATH system.* 1979.

84 H.L.L. Busard. *The translation of the elements of Euclid from the Arabic into Latin by Hermann of Carinthia (?), books vii-xii.* 1977.

85 J. van Mill. *Supercompactness and Wallmann spaces.* 1977.

86 S.G. van der Meulen, M. Veldhorst. *Torrix I, a programming system for operations on vectors and matrices over arbitrary fields and of variable size.* 1978.

88 A. Schrijver. *Matroids and linking systems.* 1977.

89 J.W. de Roever. *Complex Fourier transformation and analytic functionals with unbounded carriers.* 1978.

90 L.P.J. Groenewegen. *Characterization of optimal strategies in dynamic games.* 1981.

91 J.M. Geysel. *Transcendence in fields of positive characteristic.* 1979.

92 P.J. Weeda. *Finite generalized Markov programming.* 1979.

93 H.C. Tijms, J. Wessels (eds.). *Markov decision theory.* 1977.

94 A. Bijlsma. *Simultaneous approximations in transcendental number theory.* 1978.

95 K.M. van Hee. *Bayesian control of Markov chains.* 1978.

96 P.M.B. Vitányi. *Lindenmayer systems: structure, languages, and growth functions.* 1980.

97 A. Federgruen. *Markovian control problems; functional equations and algorithms.* 1984.

98 R. Geel. *Singular perturbations of hyperbolic type.* 1978.

99 J.K. Lenstra, A.H.G. Rinnooy Kan, P. van Emde Boas (eds.). *Interfaces between computer science and operations research.* 1978.

100 P.C. Baayen, D. van Dulst, J. Oosterhoff (eds.). *Proceedings bicentennial congress of the Wiskundig Genootschap, part 1.* 1979.

101 P.C. Baayen, D. van Dulst, J. Oosterhoff (eds.). *Proceedings bicentennial congress of the Wiskundig Genootschap, part 2.* 1979.

102 D. van Dulst. *Reflexive and superreflexive Banach spaces.* 1978.

103 K. van Harn. *Classifying infinitely divisible distributions by functional equations.* 1978.

104 J.M. van Wouwe. *GO-spaces and generalizations of metrizability.* 1979.

105 R. Helmers. *Edgeworth expansions for linear combinations of order statistics.* 1982.

106 A. Schrijver (ed.). *Packing and covering in combinatorics.* 1979.

107 C. den Heijer. *The numerical solution of nonlinear operator equations by imbedding methods.* 1979.

108 J.W. de Bakker, J. van Leeuwen (eds.). *Foundations of computer science III, part 1.* 1979.

109 J.W. de Bakker, J. van Leeuwen (eds.). *Foundations of computer science III, part 2.* 1979.

110 J.C. van Vliet. *ALGOL 68 transput, part I: historical review and discussion of the implementation model.* 1979.

111 J.C. van Vliet. *ALGOL 68 transput, part II: an implementation model.* 1979.

112 H.C.P. Berbee. *Random walks with stationary increments and renewal theory.* 1979.

113 T.A.B. Snijders. *Asymptotic optimality theory for testing problems with restricted alternatives.* 1979.

114 A.J.E.M. Janssen. *Application of the Wigner distribution to harmonic analysis of generalized stochastic processes.* 1979.

115 P.C. Baayen, J. van Mill (eds.). *Topological structures II, part 1.* 1979.

116 P.C. Baayen, J. van Mill (eds.). *Topological structures II, part 2.* 1979.

117 P.J.M. Kallenberg. *Branching processes with continuous state space.* 1979.

118 P. Groeneboom. *Large deviations and asymptotic efficiencies.* 1980.

119 F.J. Peters. *Sparse matrices and substructures, with a novel implementation of finite element algorithms.* 1980.

120 W.P.M. de Ruyter. *On the asymptotic analysis of large-scale ocean circulation.* 1980.

121 W.H. Haemers. *Eigenvalue techniques in design and graph theory.* 1980.

122 J.C.P. Bus. *Numerical solution of systems of nonlinear equations.* 1980.

123 I. Yuhász. *Cardinal functions in topology - ten years later.* 1980.

124 R.D. Gill. *Censoring and stochastic integrals.* 1980.

125 R. Eising. *2-D systems, an algebraic approach.* 1980.

126 G. van der Hoek. *Reduction methods in nonlinear programming.* 1980.

127 J.W. Klop. *Combinatory reduction systems.* 1980.

128 A.J.J. Talman. *Variable dimension fixed point algorithms and triangulations.* 1980.

129 G. van der Laan. *Simplicial fixed point algorithms.* 1980.

130 P.J.W. ten Hagen, T. Hagen, P. Klint, H. Noot, H.J. Sint, A.H. Veen. *ILP: intermediate language for pictures.* 1980.

131 R.J.R. Back. *Correctness preserving program refinements: proof theory and applications.* 1980.

132 H.M. Mulder. *The interval function of a graph.* 1980.

133 C.A.J. Klaassen. *Statistical performance of location estimators.* 1981.

134 J.C. van Vliet, H. Wupper (eds.). *Proceedings international conference on ALGOL 68.* 1981.

135 J.A.G. Groenendijk, T.M.V. Janssen, M.J.B. Stokhof (eds.). *Formal methods in the study of language, part I.* 1981.

136 J.A.G. Groenendijk, T.M.V. Janssen, M.J.B. Stokhof (eds.). *Formal methods in the study of language, part II.* 1981.

137 J. Telgen. *Redundancy and linear programs.* 1981.

138 H.A. Lauwerier. *Mathematical models of epidemics.* 1981.

139 J. van der Wal. *Stochastic dynamic programming, successive approximations and nearly optimal strategies for Markov decision processes and Markov games.* 1981.

140 J.H. van Geldrop. *A mathematical theory of pure exchange economies without the no-critical-point hypothesis.* 1981.

141 G.E. Welters. *Abel-Jacobi isogenies for certain types of Fano threefolds.* 1981.

142 H.R. Bennett, D.J. Lutzer (eds.). *Topology and order structures, part 1.* 1981.

143 J.M. Schumacher. *Dynamic feedback in finite- and infinite-dimensional linear systems.* 1981.

144 P. Eijgenraam. *The solution of initial value problems using interval arithmetic; formulation and analysis of an algorithm.* 1981.

145 A.J. Brentjes. *Multi-dimensional continued fraction algorithms.* 1981.

146 C.V.M. van der Mee. *Semigroup and factorization methods in transport theory.* 1981.

147 H.H. Tigelaar. *Identification and informative sample size.* 1982.

148 L.C.M. Kallenberg. *Linear programming and finite Markovian control problems.* 1983.

149 C.B. Huijsmans, M.A. Kaashoek, W.A.J. Luxemburg, W.K. Vietsch (eds.). *From A to Z, proceedings of a symposium in honour of A.C. Zaanen.* 1982.

150 M. Veldhorst. *An analysis of sparse matrix storage schemes.* 1982.

151 R.J.M.M. Does. *Higher order asymptotics for simple linear rank statistics.* 1982.

152 G.F. van der Hoeven. *Projections of lawless sequencies.* 1982.

153 J.P.C. Blanc. *Application of the theory of boundary value problems in the analysis of a queueing model with paired services.* 1982.

154 H.W. Lenstra, Jr., R. Tijdeman (eds.). *Computational methods in number theory, part I.* 1982.

155 H.W. Lenstra, Jr., R. Tijdeman (eds.). *Computational methods in number theory, part II.* 1982.

156 P.M.G. Apers. *Query processing and data allocation in distributed database systems.* 1983.

157 H.A.W.M. Kneppers. *The covariant classification of two-dimensional smooth commutative formal groups over an algebraically closed field of positive characteristic.* 1983.

158 J.W. de Bakker, J. van Leeuwen (eds.). *Foundations of computer science IV, distributed systems, part 1.* 1983.

159 J.W. de Bakker, J. van Leeuwen (eds.). *Foundations of computer science IV, distributed systems, part 2.* 1983.

160 A. Rezus. *Abstract AUTOMATH.* 1983.

161 G.F. Helminck. *Eisenstein series on the metaplectic group, an algebraic approach.* 1983.

162 J.J. Dik. *Tests for preference.* 1983.

163 H. Schippers. *Multiple grid methods for equations of the second kind with applications in fluid mechanics.* 1983.

164 F.A. van der Duyn Schouten. *Markov decision processes with continuous time parameter.* 1983.

165 P.C.T. van der Hoeven. *On point processes.* 1983.

166 H.B.M. Jonkers. *Abstraction, specification and implementation techniques, with an application to garbage collection.* 1983.

167 W.H.M. Zijm. *Nonnegative matrices in dynamic programming.* 1983.

168 J.H. Evertse. *Upper bounds for the numbers of solutions of diophantine equations.* 1983.

169 H.R. Bennett, D.J. Lutzer (eds.). *Topology and order structures, part 2.* 1983.