low purchase price is a genuine incentive for those interested in inverse problems and numerical analysis to add this little book to their personal libraries.

C. W. GROETSCH

Department of Mathematical Sciences
University of Cincinnati
Cincinnati, Ohio 45221-0025

14[65-04, 65F05].—THOMAS F. COLEMAN & CHARLES VAN LOAN, *Handbook for Matrix Computations*, Frontiers in Applied Mathematics, Vol. 4, SIAM, Philadelphia, PA, 1988, vii+264 pp., 23 cm. Price: Softcover $24.00.

No course in numerical linear algebra is complete without laboratory assignments. There are three good reasons for this. First, the coding of efficient matrix algorithms cannot be learned in an armchair; one must actually write programs. Second, students who have not made computer runs generally believe that rounding errors are like cooties—only other people get them. Finally, this is one of the few courses where the instructor can insist on cleanly coded, well-documented programs that would be suitable for distribution in a computation center.

Unfortunately, it is not enough to give a student an assignment and a due date. Even if the student already knows a programming language, there are tricks to making it do matrix computations efficiently. Students should be encouraged to use the Basic Linear Algebra Subprograms, which, however, have complicated calling sequences with forbidding parameters. The calling sequences for the standard matrix packages, LINPACK and EISPACK, are even worse. Finally, if the instructor decides to use an interactive package, the student must learn what is essentially a new language. Helping the student master these details can eat heavily into time better spent deriving and analyzing algorithms.

The authors of this handbook have circumvented these problems by providing their students with a small book that sets out the details in concise form. It is assumed that the reader has been exposed to a high-level programming language and has had one semester of linear algebra. The book consists of four chapters. The first is an introduction to FORTRAN. The second leads the reader through the Basic Linear Algebra Subprograms (BLAS) for performing vector operations. The third is an introduction to LINPACK, a collection of subroutines for computing and applying matrix decompositions. The last chapter is an introduction to MATLAB, an interactive system for manipulating matrices. Interesting exercises are interspersed throughout the book. Let us look at each chapter in turn.

Although they are not dogmatic about it—they make a gracious nod toward the language C—the authors have chosen FORTRAN as the language of their handbook. This makes sense. Not only is most scientific computation done in FORTRAN, but FORTRAN is the language in which the best matrix packages are coded. The chapter contains a detailed description of how arrays are arranged in memory and how they appear to subroutines. This material, which is of paramount importance in matrix computations, is slighted in most treatments of FORTRAN.

The section on programming tips is well thought out. People coming from richer languages will especially appreciate the hints for implementing constructions such as **while** loops in FORTRAN.

The BLAS are a collection of FORTRAN subprograms to perform vector operations, e.g., inner products or the addition of the multiple of one vector to another. Their power lies in the fact that the "vector" can be a column or row of an array. Thus they provide a way of coding matrix algorithms at the vector rather than the scalar level. Here the BLAS are introduced in stages, with examples illustrating their use. This chapter could have been better coordinated with the previous chapter. For example, there is no reference back to the material on arrays; the reader is left to figure out on his own how the BLAS get inside a matrix. Again, in the section on programming tips the authors stress the need to avoid overflows by scaling. Then they proceed to ignore their own good advice in Example 2.2-1.

LINPACK is a collection of FORTRAN subroutines to perform various matrix computations, all loosely associated with linear systems or linear least squares. The package follows the modern trend in numerical linear algebra by first computing a matrix decomposition, which then is used as a platform for solving a number of problems. The authors are particularly good at leading the reader through the complicated calling sequences for the subroutines. However, the chapter is flawed in two ways. First, no summary is provided at the end of the chapter, so that the experienced user who merely wants to be reminded of the calling sequences finds himself forever thumbing between the index and the text. Second, the authors drop the reader *in medias res.* A couple of pages explaining the philosophy and organization of LINPACK and defining the decompositions it uses—they are not likely to be treated in the prerequisite semester of linear algebra—would make this section much more comprehensible to the novice.

MATLAB is a system for manipulating matrices at a high level. For example to solve the least squares problem of minimizing $\|b-Ax\|^2$, where $\|\cdot\|$ is the Euclidean norm, one simply enters

$$x = A\backslash b.$$

With this much power at their fingertips, students of numerical linear algebra can experiment in a way that would have been impossible less than a decade ago. The introduction to MATLAB given here is very clean; in fact, I prefer it to MATLAB's own documentation. The one important omission is the indispensible **save** and **load** commands, which allow the user to walk away from a session and later pick up where he left off.

Another omission, which is much more serious, is that of a chapter on the EISPACK routines for eigenvalue calculations. The failure to treat this very important aspect of matrix computations detracts greatly from the value of the book.

Nonetheless, it *is* a valuable book. As a reference work it is conveniently bound and attractively formatted, and for everyday work, exclusive of occasional fine points, it replaces at least three bulkier manuals. It is also a useful supplement to a course on numerical linear algebra. However, some polishing would make it more accessible to the tyro, and I hope the authors will undertake the job in revision.

G. W. S.

15[90–02].—FERENC FORGÓ, *Nonconvex Programming*, Akadémiai Kiadó, Budapest, 1988, 188 pp., $24\frac{1}{2}$ cm. Price $24.00.

The general field of finite-dimensional optimization has largely been cultivated in areas concerned with linear, integer, and convex problems, although there are also major subfields associated with stochastic, dynamic, and nondifferentiable problems.

The term "Nonconvex Programming" is generally meant to apply to optimization problems dealing with a continuous objective function and a closed constraint region (often described by continuous functions, especially linear). Such problems have traditionally been treated by algorithms developed for convex problems, often started from several different points with the hope that one will lead to a true global optimum.

There has, however, been a significant amount of work done in the last 15–20 years in the development of methods which specifically apply to problems which have proper local solutions but whose global solution is required. This book is the first to attempt a broad and extensive summary of the various aspects of those algorithms which *guarantee* to produce such global optimizers.

Integer problems, methods based on random search and unconstrained global optimization algorithms are not covered.

Chapter 1 is an especially well-written summary of the basic results of convex optimization, and Chapter 2 is devoted to the geometric notions of convex hulls and envelopes. The three basic approaches to nonconvex programming are enumeration (direct and implicit), branch and bound, and cutting plane methods. These are summarized in Chapter 3.

One of the oldest and most studied of the nonconvex problems is that of maximizing a convex function over a polytope, and there are probably a dozen distinct algorithms that have been proposed for its solution. Some of the earlier of these are detailed in Chapter 4, while Chapter 5 discusses the basic ideas involved in treating problems whose constraint region is described (at least in part) by "reverse convex constraints", i.e., inequality constraints involving convex functions which are not to be smaller than a given constant.

Chapters 6 and 7 address methods for the largest set of nonconvex problems, including the separable and quadratic varieties. The important Fixed Charge Problem is treated in Chapter 8, while the concluding part of the book collects some isolated topics such as closed form solutions and decomposition.

While there are (understandably) a number of important topics not covered, and while there are virtually no insights into the computational efficiencies of the algorithms, this book is remarkable for the sheer extent of the items covered, and