From the foregoing description it can be seen that this textbook covers a wealth of material from a somewhat unusual viewpoint. A number of misprints, mostly minor, were detected. Some of the mathematical notation seems excessive, or excessively pedantic, but not more than is usually observed in texts at this level. The index failed to provide the needed page reference on a few occasions when it was consulted.

The author maintains a World Wide Web site to provide services for and obtain feedback from users. Among the services provided are programs and procedures related to topics in the book, and an errata sheet.

D.W.L.

44[65-02, 65Yxx]—*High performance computing—Problem solving with parallel and vector architectures*, Gary W. Sabot (Editor), Addison-Wesley, Reading, MA, 1995, xvi+246 pp., 24 cm, $45.14

This book attempts, by means of a number of case studies from scientific computing, to illustrate the techniques for developing efficient programs on high-performance computers. Applications from shock-wave physics and weather prediction have been included. These applications, as well as algorithms from numerical linear algebra, dynamic tree searching, graph theory, and mathematical programming have been implemented in several programming languages, and parallel architectures and their performance analyzed and reported. The chapters of the book correspond to contributed case studies whose intent is to identify and address high-performance computing issues at the application, algorithm, language, and machine levels. The issues of portability and scalability are addressed in the last chapter of the book and within each case study. The spectrum of target machines covered ranges from SIMD and MIMD massively parallel machines to vector machines and network of workstations. The parallelization techniques and methodologies presented are tightly coupled with the particular case studies selected. The material of the book could be useful to application software developers and could provide supplementary topics for an introductory graduate course in high-performance computing.

Elias N. Houstis
Department of Computer Sciences
Purdue University
West Lafayette, IN 47907-1398

45[65F05, 65F15, 65F20, 65F35, 65Y05, 65Y20, 65Y99]—*LAPACK Users' guide*, by E. Anderson, Z. Bai, C. Bischof, J. Demmel, J. Dongarra, J. du Croz, A. Greenbaum, S. Hammarling, A. McKenney, S. Ostrouchov, and D. Sorensen, second edition, SIAM, Philadelphia, PA, 1995, xx+325 pp., 28 cm, softcover, $28.50

As a successor to the software packages LINPACK and EISPACK, LAPACK provides more efficient and accurate routines for the solutions of dense systems of linear equations, least squares problems, eigenvalue and singular value problems. The second edition of the User's Guide is mainly for the September 30, 1994 release of version 2.0 of the package. In addition to what the first edition offers, the new edition provides pointers to the guides of several related packages including

LAPACK++ (C++ extension to LAPACK), CLAPACK (a C version of the entire LAPACK obtained by using the automatic Fortran to C conversion program), ScaLAPACK (a subset of LAPACK routines that run on certain distributed memory parallel computers), and some routines exploiting IEEE arithmetic. A number of algorithms are added to the new release of LAPACK and further discussed in the User's Guide. They are for the generalized nonsymmetric eigenproblems, the generalized linear least squares problems, the generalized singular value decomposition and a generalized banded symmetric definite eigenproblem. There are also new routines that implement the divide-and-conquer methods for symmetric eigenproblems.

Chapter 1 contains a brief introduction to LAPACK and its availability, including the World Wide Web URL address. It also provides references for several related packages mentioned earlier. Discussions on how to use CLAPACK and the difference in the definition, as well as memory allocation, of a two-dimensional array in Fortran and C are included in Section 1.11.2. I think this is useful for any C users of the package. There is no discussion on efficiency issues related to the conversion from Fortran to C.

Chapter 2 provides the contents of LAPACK and a short introduction for each algorithm. A number of subsections are added to describe the newly implemented routines.

Chapter 3 presents some performance measurements for LAPACK. Some of the old computers used in obtaining performance figures in the first edition of the User's Guide are replaced by newer machines. A new section, LAPACK Benchmark, is added to this chapter. It contains performance numbers for some of the most commonly used routines in numerical linear algebra on a variety of workstations, vector computers, and shared memory parallel computers.

Chapter 4 discusses accuracy and stability issues. The presentation is different from what appeared in the first edition and, in my opinion, is easier to read. Much of the detailed theory is discussed in separate sections marked with "Further Details". A few Fortran code fragments are included in this chapter to calculate the errors for certain quantities computed by LAPACK. These code fragments are useful, at least for expert users.

The rest of the book includes Chapter 5 for documentation and convention, Chapter 6 for installation, Chapter 7 for troubleshooting and Appendices A, B, C, D, E for indices, reference to BLAS, converting from LINPACK or EISPACK and a list of LAPACK Working Notes. Part 2 contains specifications of all the routines.

Overall, this is a well-written book, and I highly recommend it to those who are new to LAPACK or have used LAPACK and are interested in understanding or using the new routines released in version 2.0 of LAPACK.

Xiao-Chuan Cai
Department of Computer Science
University of Colorado
Boulder, CO 80309

46[00A20, 68Q40]—*The Maple V handbook*, by Martha L. Abell and James P. Braselton, AP Professional, Boston, MA, 1994, viii+726 pp., $23\frac{1}{2}$ cm, softcover, \$39.95

The authors describe this book as "a reference book for all users of Maple V, in