Reviewer: John G. Pierce
California State University, Fullerton

Introduction to **MATLAB** with Numerical Preliminaries

As the title implies, this text is intended to provide an introduction both to **MATLAB** and to scientific computation with an emphasis on numerical analysis. Three of the seven chapters deal primarily with the basics of **MATLAB**. The remaining chapters are devoted to applications, where the necessary **MATLAB** commands and procedures are introduced as needed. The text is generally well written and reader-friendly. It is filled with excellent examples, large sets of good exercises, interesting historical notes, and insightful and thorough discussions. There is also a sizeable collection of "Exercises for the Reader", with answers provided in the back of the book. These encourage students to immediately try out new ideas and test their understanding.

Chapter 1 begins with a very brief tutorial in **MATLAB**, followed by a discussion of vectors and component-wise operations with application to two-dimensional plotting. Recursion formulas are then introduced, motivated by a problem involving repayment of a loan. These formulas are implemented in **MATLAB** using a `while` loop. Early introduction of graphing and recursion allows the student to very quickly get a sense of the power of **MATLAB**, and is an excellent motivator. But in my experience, beginning students need a much more leisurely introduction to become comfortable with **MATLAB**. Such students also typically find it much easier to start with `for` loops for recursion. Chapter 2 opens with a brief but well-done discussion of some basic concepts in numerical analysis, including approximation and absolute and relative errors. Taylor polynomials are introduced to illustrate these ideas. Graphical experimentation is used to estimate approximation errors, and Taylor's theorem to obtain rigorous error bounds. Programs using `while` loops are developed to bring errors within specified tolerances. Chapter 3 discusses script and function m-files, and the differences between them, including the concepts of global and local variables. In particular, the reader is shown how to create an m-file for a mathematical function, and then use this file as input into built-in **MATLAB** functions for numerical integration, root finding and optimization.

Chapter 4 is concerned with programming in **MATLAB**. After a brief discussion of mathematical logic, `for` loops and conditionals are introduced. There is a section on writing good programs, including some experiments demonstrating the dramatic improvements that can

be achieved by vectorizing calculations whenever possible. Applications include programs for the Collatz Problem from number theory, and for Monte Carle Simulation. A program that calls itself is developed to compute determinants via cofactor expansion. But this is shown to be very inefficient when compared with the built-in function `det`, based on Gauss elimination. Chapter 5 contains a very nice discussion of floating-point arithmetic and the propagation of roundoff errors, and includes a number of excellent examples and exercises. Chapter 6 begins with an interesting account of the history of root finding, followed by a detailed discussion of the bisection, Newton, and secant methods. The last section deals with error analysis, and limitations and comparisons of the three methods.

Chapter 7 is devoted to matrices and linear systems and their applications. It consists of seven sections that together make up roughly half the total length of the text. The first section deals with matrices and their basic properties, as well as with matrix computations and manipulations in MATLAB. In the exercises, students are encouraged to decide whether or not proposed matrix properties are true by experimentation. Section 2 contains a really nice treatment of linear transformations and their applications to computer graphics and animation. Included are detailed discussions of making movies in MATLAB and of creating fractals. Several interesting examples are given. I believe that students of linear algebra will find this material very motivational. Many readers will consider it the high point of the text. The next four sections deal with linear systems of equations. Various approaches for solving a general linear system using built-in MATLAB functions are described and then compared for efficiency and accuracy on a sensitive problem. Gauss elimination and LU factorization are then treated in detail, including such practical issues as pivoting, operations counts and numerical sensitivity. Several very nice applications of linear systems are given in the examples and exercises. In Section 6, matrix norms are introduced, the condition number of a matrix is defined, and error estimates are obtained. The chapter closes with a section on iterative methods for linear systems, including Jacobi, Gauss-Seidel and SOR iteration. These methods are nicely compared from both a theoretical and experimental point of view. The text concludes with two appendices. The first describes how to use MATLAB's Symbolic Toolbox; the second gives complete solutions to all the "Exercises for the Reader".

Overall, this is a very good book, and the author's obvious enthusiasm for mathematics and computation makes it a pleasure to read. There is much here to motivate a strong beginner or an above-average student with some background in programming or in numerical analysis. The text would also serve as a very nice supplement for a course in linear algebra, numerical analysis, or general problem solving using the computer. However, I would not recommend it as a stand-alone text for the average student in a first exposure to both computer programming and numerical analysis.

Reviewer:

John G. Pierce
California State University
Department of Mathematics
Fullerton, CA 92834
E-mail: jpierce@fullerton.edu