

Preface

This book is about using MATLAB to obtain numerical solutions to problems in engineering and science. The emphasis is on the application of standard numerical techniques, not on the analysis of those techniques. Mathematical theory is used primarily to develop the logic of the algorithms and to identify the strengths and weaknesses of different numerical methods. In most cases, the behavior of the methods is exposed by numerical experiment.

The intended audience is students in engineering and science learning how to use numerical methods in their own disciplines. I also hope the book will be useful to practicing engineers and scientists who have a specific numerical task and wish to implement a solution in MATLAB. The reader needs no previous experience in numerical analysis. A solid understanding of basic calculus is assumed, as is some previous exposure to linear algebra, differential equations, and computer programming.

This is not a “how to . . . ” book. Although there are many examples that show how to use MATLAB to solve practical problems, the “how” is accompanied by a strong dose of “why.” The basic theory of each method is developed before that method is implemented in MATLAB. If you are interested in a set of recipes for plugging numbers into MATLAB statements, you should put this book down

and select another one from the shelf. If you are interested in developing an understanding of how certain numerical methods work and how different methods used to accomplish the same task compare, read on.

Not every routine presented here is recommended for solving practical problems. Elementary algorithms are implemented and their performance is investigated. Advanced techniques are then developed as improvements on the elementary algorithms. By progressing from the simple to the sophisticated, the reader is given an understanding of the fundamentals, as well as guidance on how to apply more reliable and efficient algorithms to practical problems. Numerous sample problems are provided in each chapter. Where applicable, the same problem is solved with different methods, and the results are compared. Simple examples allow the reader to focus on the fundamental aspects of the numerical methods. More sophisticated examples demonstrate the solution to realistic and complicated practical problems.

The first part of the book provides an introduction to MATLAB as an interactive computing system and as a programming language. It is assumed that the reader has some prior experience with computer programming. The material is presented as a first course in numerical methods, not a first course in programming. An ambitious and determined person with significant computer experience could probably use this material as a condensed introduction to programming, although, for that purpose, there are better books.

The routines and data files presented in the book constitute the Numerical Methods with MATLAB (NMM) toolbox, which is available at <http://www.prenhall.com/recktenwald>. Once the NMM toolbox is installed on your computer, you will be able to interactively execute all of the sample problems in this book, as well as apply the NMM toolbox routines to practical problems in your own work.

To the Student

Learning to use computers can be an extremely frustrating experience. It is very easy, in a state of exhaustion and despair, to conclude that numerical methods are (choose one or all of the following) (1) a waste of effort, (2) solely a means of torture, (3) only to be learned by computer geeks, (4) making me less, not more, productive. When I teach numerical methods, I warn my students that they can expect an initial drop in their productivity as they learn a new tool such as MATLAB. It is inevitable that mastering something as powerful and multifaceted as MATLAB will take patience and commitment. The rewards, however, are significant.

The best way to learn a computer tool is to use it. With the knowledge that it may at times be frustrating, jump in. Try it. Let mistakes happen. You will not damage the computer by entering an incorrect MATLAB command. Have this book open when you are running MATLAB. Enter the commands that appear in the text,

and explain the outcome in your own words. Try not to rush through the examples. Do not treat them as recipes that must never be altered. Experiment. See if you can break the examples by changing the input parameters. The failure of a numerical solution is a learning experience. Why does one set of inputs work when another does not? What is the smallest difference between two set of inputs that differentiates success from failure? Does the method fail because of an inherent weakness, or are the inputs not valid? Will a more powerful method work for the same inputs?

Use the study guides, lecture notes, solutions to selected end-of-chapter exercises, and other supplemental learning material at <http://www.prenhall.com/recktenwald> to complement the information in the book. The study guides are designed to help you master the material presented in each chapter, and contain learning goals that will be most useful if consulted after reading the chapter. Your instructor may highlight specific learning goals from the study guides. In each chapter solutions to selected end-of-chapter exercises, designated by SOLUTION, are provided.

To the Instructor

Various subsets of the material in this book may be organized into college-level courses. To help orient the reader, the beginning of each chapter contains a brief outline of the major topics covered. Study guides for each chapter provide a series of learning objectives divided into basic, intermediate, and advanced topics. Recognizing that the classification of topics is subjective, I am making the study guides available as PDF and as plain text documents at <http://www.prenhall.com/recktenwald>. Instructors are encouraged to download and adapt the study guides to their own selection of topics. Other course materials, including transparency masters, are also available at <http://www.prenhall.com/recktenwald>.

Some very important practical areas of numerical computation are not covered in the book: iterative methods for linear systems of equations, the fast Fourier transform, partial differential equations, and minimization (optimization). These topics were included in the original outline, but as the manuscript evolved, and the sabbatical allocated for writing expired, it became clear that not everything in my plan would make it into the final draft. The absence of some numerical methods will be viewed as a limitation by some instructors. This has not been a problem for the course that was the impetus for writing the book. I believe that undergraduate students are better served by providing a sound foundation in the fundamentals than by a quickly paced survey of many numerical methods. The focus of this book is on providing such a foundation.

I believe that it is counterproductive to require engineering students to develop all of their own programs from pseudocode. This is not to suggest that numerical methods should be treated superficially or that the reader use the NMM

toolbox routines as black boxes. The source code for the NMM toolbox is provided with the intention that it is to be read as an integral part of the text. Many of the end-of-chapter assignments require modifications to the source code presented in the chapter, or application of the codes to new problems. Starting with working codes allows you to challenge your students through the *application* of numerical methods to your discipline. This frees you to focus on the behavior of each numerical method without having to guide students as they reinvent standard methods. Having a library of working codes at their disposal, students are also more likely to apply numerical methods in other classes. This provides an opportunity for integrating numerical methods into the undergraduate curriculum.

A Few Words of Caution

As with any work of this scope and size, there is certainly room for improvement. If you find errors or have suggestions, I would be grateful if you take the time to point them out to me. Send all corrections and suggestions for improvements via e-mail. I will maintain errata at <http://www.prenhall.com/recktenwald>.

I cannot, of course, in any way be responsible for negative (or positive!) outcomes from the application of the techniques presented here. There are many ways to apply a correct algorithm to the wrong problem. There are also limits to applying any technique to the problems for which it was designed. Though I have tried to offer guidance, you assume all the risk of using any of the algorithms, codes, and recommendations provided in the book.

Thanks

I have accumulated many debts on the path toward completion of this book. The writers and researchers that I have drawn upon have provided a rich background that I have synthesized into my own writing. The creators of MATLAB have produced a useful and beautiful tool that has significantly advanced my understanding of numerical mathematics. I have benefited immensely from the NA-net and NA-teach electronic digests. My teachers were inspiring and patient as I began my training in engineering and numerical problem-solving. My students endured experiments in my development of the draft manuscript. My colleagues at Portland State provided much encouragement and stimulation. In particular, I would like to thank Dave Turcic for providing feedback on the book as I was writing it and for using an early draft as a textbook in a course on numerical methods for engineers.

Many people at Prentice Hall have given me encouragement, sound advice, and judicious prodding. I am indebted to Denise Penrose and Rob Merino for encouragement at the beginning of this project. I am grateful to Susan Slater and Michael Slaughter for editorial advice and gentle prodding in the middle phase of manuscript development. I am especially appreciative of Eric Frank for his sound

understanding of textbooks, and his enthusiastic support of my vision for what this book could become. I owe many thanks to Lisa Bell-Greer and the worker bees at Revolution Publishing in Portland for making exceptional-quality, small production runs of the draft manuscript. Final thanks, chronologically speaking, go to the staff at Integre Technical Publishing, and in particular to Leslie Galen (the midwife), for polishing my prose and Don DeLand for putting sparkle into my \LaTeX source.

The following people read drafts and provided valuable feedback: Ron Buckmire (Occidental College), Anand Cousins (Seattle University), John D'Errico (Kodak Corporation), Bahram Shahian (California State University, Long Beach), and Howard Wilson (University of Alabama, Huntsville). I am responsible, of course, for any mistakes in the book.

On a personal level, I am very grateful for my parents' support of my intellectual and geographic journeys. I am also lucky to have good friends that have encouraged me and shared a lot of fun. My son Cullen has given me abundant inspiration and joy. My wife and dear companion, Elsbeth, has given much support and affection to help me toward this and other goals. I dedicate this book to her.

Gerald Recktenwald
gerry@me.pdx.edu