FUNDAMENTALS OF ENGINEERING NUMERICAL ANALYSIS

Engineers need hands-on experience in solving complex technical problems with computers. This text introduces numerical methods and shows how to develop, analyze, and use them. A thorough and practical book, it is intended as a first course in numerical analysis, primarily for beginning graduate students in engineering and physical science. Along with mastering the fundamentals of numerical methods, students will learn to write their own computer programs using standard numerical methods, or to use popular software packages such as MATLAB or programs in *Numerical Recipes* (Cambridge University Press). They will learn what factors affect accuracy, stability, and convergences, and they will also come to view critically the numerical output spewed from a computer. Numerous examples and exercises give students first-hand experience. The material is based on what Professor Moin found to be useful in teaching a course in numerical analysis and in his own career as a computational physicist/engineer.

With the availability of ever more powerful computers, numerical simulation of physical phenomena has become more practical and more widespread. This introductory text will guide students as well as practicing engineers in the use of computational teachings in their work.

Parviz Moin is Franklin P. and Caroline M. Johnson Professor of Engineering and Director of the Center for Turbulence Research, Stanford University. He pioneered the use of numerical simulation techniques for the study of turbulence physics, control, and modeling concepts. Professor Moin is a Fellow of the American Physical Society and a Member of the National Academy of Engineering.

FUNDAMENTALS OF ENGINEERING NUMERICAL ANALYSIS

PARVIZ MOIN
Stanford University

CAMBRIDGE
UNIVERSITY PRESS

PUBLISHED BY THE PRESS SYNDICATE OF THE UNIVERSITY OF CAMBRIDGE
The Pitt Building, Trumpington Street, Cambridge, United Kingdom

CAMBRIDGE UNIVERSITY PRESS
The Edinburgh Building, Cambridge CB2 2RU, UK
40 West 20th Street, New York, NY 10011-4211, USA
10 Stamford Road, Oakleigh, VIC 3166, Australia
Ruiz de Alarcón 13, 28014 Madrid, Spain
Dock House, The Waterfront, Cape Town 8001, South Africa

http://www.cambridge.org

First published 2001

Printed in the United States of America

Typefaces Times New Roman 10.75/13.5 pt., Melior and Helvetica Neue *System* LaTeX 2ε [TB]

A catalog record for this book is available from the British Library.

Library of Congress Cataloging in Publication Data

Moin, Parviz,

Fundamentals of engineering numerical analysis / Parviz Moin.

p. cm.

ISBN 0-521-80140-0 – ISBN 0-521-80526-0 (pb.)

1. Engineering mathematics. 2. Numerical analysis. I. Title.

TA335 .M65 2001

620′.001′5194 – dc21 00-052933

ISBN 0 521 80140 0 hardback
ISBN 0 521 80526 0 paperback

To Linda

Contents

Preface

With the advent of faster computers, numerical simulation of physical phenomena is becoming more practical and more common. Computational prototyping is becoming a significant part of the design process for engineering systems. With ever-increasing computer performance the outlook is even brighter, and computer simulations are expected to replace expensive physical testing of design prototypes.

This book is an outgrowth of my lecture notes for a course in computational mathematics taught to first-year engineering graduate students at Stanford. The course is the third in a sequence of three quarter-courses in computational mathematics. The students are expected to have completed the first two courses in the sequence: numerical linear algebra and elementary partial differential equations. Although familiarity with linear algebra in some depth is essential, mastery of the analytical tools for the solution of partial differential equations (PDEs) is not; only familiarity with PDEs as governing equations for physical systems is desirable. There is a long tradition at Stanford of emphasizing that engineering students learn numerical analysis (as opposed to learning to run canned computer codes). I believe it is important for students to be educated about the fundamentals of numerical methods. My first lesson in numerics includes a warning to the students not to believe, at first glance, the numerical output spewed out from a computer. They should know what factors affect accuracy, stability, and convergence and be able to ask tough questions before accepting the numerical output. In other words, the user of numerical methods should not leave all the "thinking" to the computer program and the person who wrote it. It is also important for computational physicists and engineers to have first-hand experience with solving real problems with the computer. They should experience both the power of numerical methods for solving non-trivial problems as well as the frustration of using inadequate methods. Frustrating experiences with a numerical method almost always send a competent numerical analyst to the drawing board and force him or her to ask good questions about the choice and parameters of the method, which should have been asked before going to the computer in the first place. The exercises at the end of each chapter are intended to give these important experiences with numerical methods.

Along with mastering the fundamentals of numerical methods, the students are expected to write their own programs to solve problems using standard numerical methods. They are also encouraged to use standard (commercial) software whenever possible. There are several software libraries with well-documented programs for basic computational work. Recently, I have used the *Numerical Recipes* by Press et al. (Cambridge) as an optional supplement to my lectures. *Numerical Recipes* is based on a large software library that is well documented and available on computer disks. Some of the examples in this book refer to specific programs in *Numerical Recipes*.

Students should also have a simple (x, y) plotting package to display their numerical results. Some students prefer to use MATLAB's plotting software, some use the plotting capability included with a spreadsheet package, and others use more sophisticated commercial plotting packages. Standard well-written numerical analysis programs are generally available for almost everything covered in the first four chapters, but this is not the case for partial differential equations discussed in Chapter 5. The main technical reason for this is the large variety of partial differential equations, which requires essentially tailor-made programs for each application.

No attempt has been made to provide complete coverage of the topics that I have chosen to include in this book. This is not meant to be a reference book, rather it contains the material for a first course in numerical analysis for future practitioners. Most of the material is what I have found useful in my career as a computational physicist/engineer. The coverage is succinct, and it is expected that all the material will be covered sequentially. The book is intended for first-year graduate students in science and engineering or seniors with good post-calculus mathematics backgrounds. The first five chapters can be covered in a one-quarter course, and Chapter 6 can be included in a one-semester course.

Discrete data and numerical interpolation are introduced in Chapter 1, which exposes the reader to the dangers of high-order polynomial interpolation. Cubic splines are offered as a good working algorithm for interpolation. Chapter 2 (finite differences) and Chapter 3 (numerical integration) are the foundations of discrete calculus. Here, I emphasize systematic procedures for constructing finite difference schemes, including high-order Padé approximations. We also examine alternative, and often more informative, measures of numerical accuracy. In addition to introducing the standard numerical integration techniques and their error analysis, we show in Chapter 3 how knowledge of the form of numerical errors can be used to construct more accurate numerical results (Richardson extrapolation) and to construct adaptive schemes that obtain the solution to the accuracy specified by the user. Usually, at this point in my lectures, I seize the opportunity, offered by these examples, to stress the value of a detailed knowledge of numerical error and its pay-offs even for the most application-oriented students. Knowledge is quickly transferred to power in constructing novel numerical methods.

Chapter 4 is on numerical solution of ordinary differential equations (ODEs) – the heart of this first course in numerical analysis. A number of new concepts such as stability and stiffness are introduced. The reader begins to experience new tools in his arsenal for solving relatively complex problems that would have been impossible to do analytically. Because so many interesting applications are cast in ordinary differential equations, this chapter is particularly interesting for engineers. Different classes of numerical methods are introduced and analyzed even though there are several well-known powerful numerical ODE solver packages available to solve any practical ODE without having to know their inner workings. The reason for this extensive coverage of a virtually solved problems is that the same algorithms are used for solution of partial differential equations when canned programs for general PDEs are not available and the user is forced to write his or her own programs. Thus, it is essential to learn about the properties of numerical methods for ODEs in order to develop good programs for PDEs.

Chapter 5 discusses the numerical solution of partial differential equations and relies heavily on the analysis of initial value problems introduced for ODEs. In fact by using the modified wavenumber analysis, we can cast into ODEs the discretized initial value problems in PDEs, and the knowledge of ODE properties becomes very useful and no longer of just academic value. Once again the knowledge of numerical errors is used to solve a difficult problem of dealing with large matrices in multi-dimensional PDEs by the approximate factorization technique. Dealing with large matrices is also a focus of numerical techniques for elliptic partial differential equations, which are dealt with by introducing the foundations of iterative solvers.

Demand for high accuracy is increasing as computational engineering matures. Today's engineers and physicists are less interested in qualitative features of numerical solutions and more concerned with numerical accuracy. A branch of numerical analysis deals with spectral methods, which offer highly accurate numerical methods for solution of partial differential equations. Chapter 6 covers aspects of Fourier analysis and introduces transform methods for partial differential equations.

My early work in numerical analysis was influenced greatly by discussions with Joel Ferziger and subsequently by the works of Harvard Lomax at NASA–Ames. Thanks are due to all my teaching assistants who helped me develop the course upon which this book is based; in particular, I thank Jon Freund and Arthur Kravchenko who provided valuable assistance in preparation of this book. I am especially grateful to Albert Honein for his substantial

help in preparing this book in its final form and for his many contributions as my teaching assistant in several courses in computational mathematics at Stanford.

Parviz Moin
Stanford, California
July 2000