

<<数值分析>>

图书基本信息

书名：<<数值分析>>

13位ISBN编号：9787111385820

10位ISBN编号：7111385829

出版时间：2012-6

出版时间：机械工业出版社

作者：（美）Timothy Sauer

页数：646

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

<<数值分析>>

内容概要

美国萨奥尔编著的《数值分析》是一本优秀的数值分析教材，书中不仅全面论述了数值分析的基本方法，还深入浅出地介绍了计算机和工程领域使用的一些高级数值方法，如压缩、前向和后向误差分析、求解方程组的迭代方法等。

每章的“实例检验”部分结合数值分析在各领域的具体应用实例，进一步探究如何更好地应用数值分析方法解决实际问题。

此外，书中含有一些算法的matlab实现代码，并且每章都配有大量难度适宜的习题和计算机问题，便于读者学习、巩固和提高。

<<数值分析>>

作者简介

萨奥尔 (Timothy Sauer) , 美国乔治梅森大学数学系教授。

1982年于加州大学伯克利分校获得数学专业博士学位, 师从著名数学家Robin Hartshorne。

他的主要研究领域为动力系统、计算数学和数学生物学。

他是《SIAM Journal on Applied Dynamical Systems》、《Journal of Difference Equations and Applications》和《Physica D》等学术期刊的编委。

<<数值分析>>

书籍目录

PREFACE

CHAPTER0 Fundamentals

- 0 . 1 Evaluating a Polynomial
- 0 . 2 Binary Numbe
- 0 . 2 . 1 Decimal to binary
- 0 . 2 . 2 Binary to decimal
- 0 . 3 Floating Point Representation of Real Numbe
- 0 . 3 . 1 Floating point fclrmats
- 0 . 3 . 2 Machine reDresentatiOn
- 0 . 3 . 3 Addition offloating point numbe
- 0 . 4 Loss of Significance
- 0 . 5 Review of Calculus

Software and Further Reading

CHAPTER 1 Solving Equatio

- 1 . 1 The Bisection Method
- 1 . 1 . 1 Bracketing a root
- 1 . 1 . 2 Howaccurate and howfast?
- 1 . 2 Fixed. Point Iteration
- 1 . 2 . 1 Fixed points of a function
- 1 . 2 . 2 Geometry of Fixed. Point Iteration
- 1 . 2 . 3 Linear convergence of Fixed. Point Iteration
- 1 . 2 . 4 Stopping criteria
- 1 . 3 Limits of Accuracy
- 1 . 3 . 1 Forward and backward error
- 1 . 3 . 2 The Wilkion polynomial
- 1 . 3 . 3 Seitivity of root. finding
- 1 . 4 Newton's Method
- 1 . 4 . 1 Quadratic convergence of Newton's Method
- 1 . 4 . 2 Linear convergence of Newton's Method
- 1 . 5 Root. Finding without Derivatives
- 1 . 5 . 1 Secant Method and variants
- 1 . 5 . 2 Brent3 Method

Reality Check1 : Kinematics ofthe Stewart platform

Software and Further Reading

CHAPTER 2 Systems of Equatio

- 2 . 1 Gaussian Elimination
- 2 . 1 . 1 Naive Gaussian elimination
- 2 . 1 . 2 Operation counts
- 2 . 2 The LU FactO rizatiOn
- 2 . 2 . 1 Matrix form of Gaussian elimination
- 2 . 2 . 2 Back substitution with the LU f2lctorization
- 2 . 2 . 3 Complexity of the LU factorization
- 2 . 3 Sources of Error
- 2 . 3 . 1 Error magnification and condition number
- 2 . 3 . 2 Swamping

<<数值分析>>

2 . 4 The PA=LU Factorization

2 . 4 . 1 Partial pivoting

2 . 4 . 2 Permutation matrices

2 . 4 . 3 PA=LU factorization

Reality Check 2: The Euler. Bernoulli Beam

2 . 5 Iterative Methods

2 . 5 . 1 Jacobi Method

2 . 5 . 2 Gauss—Seidel Method and SOR

2 . 5 . 3 Convergence of iterative methods

2 . 5 . 4 Sparse matrix computation

2 . 6 Methods for symmetric positive. definite matrices

2 . 6 . 1 Symmetric positive. definite matrices

2 . 6 . 2 Cholesky factorization

2 . 6 . 3 Conjugate Gradient Method

2 . 6 . 4 Preconditioning

2 . 7 Nonlinear Systems of Equations

2 . 7 . 1 Multivariate Newton's Method

2 . 7 . 2 Broyden's Method

Software and Further Reading

CHAPTER 3 Interpolation

3 . 1 Data and Interpolating Functions

3 . 1 . 1 Lagrange interpolation

3 . 1 . 2 Newton's divided differences

3 . 1 . 3 How many degree d polynomials pass through n points?

3 . 1 . 4 Code for interpolation

3 . 1 . 5 Representing functions by approximating polynomials

3 . 2 Interpolation Error

3 . 2 . 1 Interpolation error formula

3 . 2 . 2 Proof of Newton form and error formula

3 . 2 . 3 Runge phenomenon

3 . 3 Chebyshev Interpolation

3 . 3 . 1 Chebyshev's theorem

3 . 3 . 2 Chebyshev polynomials

3 . 3 . 3 Change of interval

3 . 4 Cubic Splines

3 . 4 . 1 Properties of splines

3 . 4 . 2 Endpoint conditions

3 . 5 Bézier Curves

Reality Check 3 : Fonts from Bézier curves

Software and Further Reading

CHAPTER 4 Least Squares

4 . 1 Least Squares and the Normal Equations

4 . 1 . 1 Inconsistent systems of equations

4 . 1 . 2 Fitting models to data

4 . 1 . 3 Conditioning of least squares

4 . 2 A Survey of Models

<<数值分析>>

- 4 . 2 . 1 Periodic data
 - 4 . 2 . 2 Data linearization
 - 4 . 3 QR Factorization
 - 4 . 3 . 1 Gram. Schmidt Orthoanizationalization and least squares
 - 4 . 3 . 2 Modified Gram. Schmidt orthogonalization
 - 4 . 3 . 3 Householder reflecto
 - 4 . 4 Generalized Minimum Residual(GMRES)Method
 - 4 . 4 . 1 Krylov methods
 - 4 . 4 . 2 PrecOnditiOned GMRES
 - 4 . 5 Nonlinear Least Squares
 - 4 . 5 . 1 Gauss. Newton Method
 - 4 . 5 . 2 Models with nonlinear paramete
 - 4 . 5 . 3 The Levenberg. Marquardt Method .
- Reality Check4 : GPS , Conditioning , and Nonlinear Least Squares
Software and Further Reading

CHAPTER 5 Numerical Differentiation and Integration

- 5 . 1 Numerical Differentiation
- 5 . 1 . 1 Finite difference formulas
- 5 . 1 . 2 Rounding error
- 5 . 1 . 3 Extrapolation
- 5 . 1 . 4 Symbolic differentiation and integration
- 5 . 2 Newton. Cotes Formulas for Numerical Integration
- 5 . 2 . 1 Trapezoid Rule
- 5 . 2 . 2 Simpson's Rule
- 5 . 2 . 3 Composite Newton. Cotes formulas
- 5 . 2 . 4 Open Newton. Cotes Methods
- 5 . 3 Romberg Integration
- 5 . 4 Adaptive Quadrature
- 5 . 5 Gaussian Quadrature

Reality Check5 : Motion Control in Computer. Aided Modeling
Software and Further Reading

CHAPTER 6 Ordinary Differential Equatio

- 6 . 1 Initial Value Problems
- 6 . 1 . 1 Euler's Method
- 6 . 1 . 2 Existence, uniqueness . and continuity for solutio
- 6 . 1 . 3 Fit. order linear equatio
- 6 . 2 Analysis of IVP Solve
- 6 . 2 . 1 Local and global truncation error
- 6 . 2 . 2 The explicit Trapezoid Method
- 6 . 2 . 3 Taylor Methods
- 6 . 3 Systems of Ordinary Difl . erential Equatio
- 6 . 3 . 1 Higher Order equatio
- 6 . 3 . 2 Computer simulation : the pendulum
- 6 . 3 . 3 Computer simulation : orbital mechanics
- 6 . 4 Runge. Kutta Methods and Applicatio
- 6 . 4 . 1 The Runge. Kutta family

<<数值分析>>

6 . 4 . 2 Computer simulation : the Hodgkin. Huxley neuron

6 . 4 . 3 Computer simulation : the Lorenz equatio

RealityCheck 6The Tacoma Narrows Bridge

6 . 5 Variable Step. Size Methods

6 . 5 . 1 Embedded Runge. Kutta pai

6 . 5 . 2 Order 4 / 5 methods

6 . 6 Implicit Methods and Stiff Equatio

6 . 7 Multistep Methods

6 . 7 . 1 Generating multistep methods

6 . 7 . 2 Explicit multistep methods

6 . 7 . 3 Implicit multistep methods

Software and Further Reading

CHAPTER 7 Boundary Value Problems

7 . 1 Shooting Method

7 . 1 . 1 Solutio of boundary value problems

7 . 1 . 2 Shooting Method implementation

Reality Check7:Buckling of a Circular Ring

7 . 2 Finite Difference Methods

7 . 2 . 1 Linear boundary value problems

7 . 2 . 2 Nonlinear boundary value problems

7 . 3 Collocation and the Finite Element Method

7 . 3 . 1 Collocation

7 . 3 . 2 Finite elements and the Galerkin Method

Software and Further Reading

CHAPTER 8 Partial Differential Equatio

8 . 1 Parabolic Equatio

8 . 1 . 1 Forward Difference Method

8 . 1 . 2 Stability analysis of Forward Difierence Method

8 . 1 . 3 Backward Di fference Method

8 . 1 . 4 Crank. Nicolson Method

8 . 2 Hyperbolk : Equatio

8 . 2 . 1 The wave equation

8 . 2 . 2 The CFL condition

8 . 3 Elliptic Equatio

8 . 3 . 1 Finite Difference Method for elliptic equatio

RealityCheck8 : Heat distribution on a cooling fin

8 . 3 . 2 Finite Element Method for elliptic equatio

8 . 4 Nonlinear partial differential equatio

8 . 4 . 1 Implicit Newton solver

8 . 4 . 2 Nonlinear equatio in two space dimeio

Software and Further Reading

CHAPTER 9 Random Numbe and Applicatio

9 . 1 Random Numbe

9 . 1 . 1 Pseudo. random numbe

9 . 1 . 2 Exponential and normal random numbe

9 . 2 Monte Carlo Simulation

9 . 2 . 1 Power laws for Monte Carlo estimation

<<数值分析>>

- 9 . 2 . 2 Quasi. random numbe
- 9 . 3 Discrete and Continuous Brownian Motion
- 9 . 3 . 1 Random walks
- 9 . 3 . 2 Continuous Brownian motion
- 9 . 4 Stochastic DifFerential Equatio
- 9 . 4 . 1 Adding noise to differential equatio
- 9 . 4 . 2 Numerical methods for SDEs

Reality Check 9 : The Black. Scholes Formula Software and Further Reading

CHAPTER 10 Trigonometric Interpolation and the FFT

- 10 . 1 The Fourier Trafo ml
- 10 . 1 . 1 Complex arithmetic
- 10 . 1 . 2 Discrete Fourier Traform
- 10 . 1 . 3 The Fast Fourier Traform
- 10 . 2 Trigonometric Interpolation
- 10 . 2 . 1 The DFT Interpolation Theorem
- 10 . 2 . 2 E fficient evaluation of trigonometric functio
- 10 . 3 The FFT and Signal Processing
- 10 . 3 . 1 Orthogonality and interpolation
- 10 . 3 . 2 Least squares fitting with trigonometric functio
- 10 . 3 . 3 Sound , noise , and filtering

Reality Check 10 : The Wiener Filter

Software and Further Reading

CHAPTER 11 Compression

- 11 . 1 The Discrete Cosine Traform
- 11 . 1 . 1 One. dimeional DCT
- 11 . 1 . 2 The DCT and least squares approximation
- 11 . 2 Two. Dimeional DCT and Image Compression
- 11 . 2 . 1 Two. dimeional DCT
- 11 . 2 . 2 Image compression
- 11 . 23 Quantization
- 11 . 3 HufFman Coding
- 11 . 3 . 1 Information theory and coding
- 11 . 3 . 2 Huffman coding for the JPEG format
- 11. 14 Modified DCT and Audio Compression
- 11 . 4 . 1 Modified Discrete Cosine Traform
- 11 . 4 . 2 Bit quantization

Reality Check 11 : A Simple Audio Codec

Software and Further Reading

CHAPTER 12 Eigenvalues and Singular Values

- 12 . 1 Power Iteration Methods
- 12 . 1 . 1 Power Iteration
- 12 . 1 . 2 Convergence of Power Iteration
- 12 . 1 . 3 Invee Power Iteration
- 12 . 1 . 4 Rayleigh Quotient Iteration
- 12 . 2 QR Algorithm
- 12 . 2 . 1 Simultaneous iteration

<<数值分析>>

12 . 2 . 2 Real Schur form and the QR algorithm

12 . 2 . 3 Upper Hessenberg form

Reality Check 12 : How Search Engines Rate Page Quality

12 . 3 Singular Value Decomposition

12 . 3 . 1 Finding the SVD in general

12 . 3 . 2 Special case : symmetric matrices

12 . 4 Application of the SVD

12 . 4 . 1 Properties of the SVD

12 . 4 . 2 Dimension reduction

12 . 4 . 3 Compression

12 . 4 . 4 Calculating the SVD

Software and Further Reading

CHAPTER 13 Optimization

13 . 1 Unconstrained Optimization without Derivatives

13 . 1 . 1 Golden Section Search

13 . 1 . 2 Successive parabolic interpolation

13 . 1 . 3 Nelder-Mead search

13 . 2 Unconstrained Optimization with Derivatives

13 . 2 . 1 Newton's Method

13 . 2 . 2 Steepest Descent

13 . 2 . 3 Conjugate Gradient Search

Reality Check 13 : Molecular Conformation and Numerical Optimization

Software and Further Reading

Appendix A

A . 1 Matrix Fundamentals

A . 2 Block Multiplication

A . 3 Eigenvalues and Eigenvectors

A . 4 Symmetric Matrices

A . 5 Vector Calculus

Appendix B

B . 1 Starting MATLAB

B . 2 Graphics

B . 3 Programming in MATLAB

B . 4 Flow Control

B . 5 Functions

B . 6 Matrix Operations

B . 7 Animation and Movies

ANSWERS TO SELECTED EXERCISES

BIBLIOGRAPHY

INDEX

章节摘录

版权页：插图： We must truncate the number in some way, and in so doing we necessarily make a small error. One method, called chopping, is to simply throw away the bits that fall off the end—that is, those beyond the 52nd bit to the right of the decimal point. This protocol is simple, but it is biased in that it always moves the result toward zero. The alternative method is rounding. In base 10, numbers are customarily rounded up if the next digit is 5 or higher, and rounded down otherwise. In binary, this corresponds to rounding up if the bit is 1. Specifically, the important bit in the double precision format is the 53rd bit to the right of the radix point, the first one lying outside of the box. The default rounding technique, implemented by the IEEE standard, is to add 1 to bit 52 (round up) if bit 53 is 1, and to do nothing (round down) to bit 52 if bit 53 is 0, with one exception: If the bits following bit 52 are 10000..., exactly halfway between up and down, we round up or round down according to which choice makes the final bit 52 equal to 0. (Here we are dealing with the mantissa only, since the sign does not play a role.) Why is there the strange exceptional case? Except for this case, the rule means rounding to the normalized floating point number closest to the original number—hence its name, the Rounding to Nearest Rule. The error made in rounding will be equally likely to be up or down. Therefore, the exceptional case, the case where there are two equally distant floating point numbers to round to, should be decided in a way that doesn't prefer up or down systematically. This is to try to avoid the possibility of an unwanted slow drift in long calculations due simply to a biased rounding. The choice to make the final bit 52 equal to 0 in the case of a tie is somewhat arbitrary, but at least it does not display a preference up or down. Problem 8 sheds some light on why the arbitrary choice of 0 is made in case of a tie.

<<数值分析>>

编辑推荐

<<数值分析>>

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>