

# Contents

<b>1</b>	<b>Computers, Errors, and Algorithms</b>	<b>1</b>
1.0	Introduction	1
1.1	What Is a Numerical Method?	1
1.1A	Numerical Methods and Termination Tests	2
1.1B	Why Use a Programmable Device for Numerical Methods?	5
1.2	Do Digital Devices Make Mistakes?	6
1.2A	An Example of Digital Device Errors	6
1.2B	Errors in Using the Quadratic Formula	8
1.3	How Do Computers Store Real Numbers?	9
1.3A	Binary Numbers	9
1.3B	Storing Integers	10
1.3C	Storing Floating-Point Representations of Binary Numbers	11
1.3D	What Numbers Can Be Stored on a 23-Bit Mantissa Device?	12
1.3E	Inherent Error	13
1.3F	Bases Other Than 2	14
1.4	Errors in Fixed-Precision Arithmetic	14
1.4A	Decimal-Place Versus Significant-Digit Accuracy	14
1.4B	Fixed-Precision Arithmetic	15
1.4C	Errors in Fixed-Precision Arithmetic	16
1.4D	Internal Functions	17
1.5	Avoiding Roundoff Error	17
1.5A	Rules for Reducing Roundoff Errors	18
1.5B	Using Trigonometric Rearrangement to Avoid Loss of Significance	21
1.5C	Using Algebraic Rearrangement to Avoid Loss of Significance	22
1.5D	Using Maclaurin Series to Avoid Loss of Significance	23
1.5E	Extended Precision	24

<b>1.6</b>	<b>Errors, Accuracy, and Tests for Closeness</b>	25
1.6A	Absolute, Relative, and Percent Error	25
1.6B	Relating $\epsilon_x$ and $\rho_x$ to the Accuracy of $X$	26
1.6C	A Quantitative Discussion of Error Propagation (Optional)	26
<b>1.7</b>	<b>Iterative Algorithms</b>	28
1.7A	Flowchart for a General Iterative Algorithm	29
1.7B	Pseudoprogram for a General Iterative Algorithm	30
1.7C	Finding Fixed Points by Repeated Substitution	31
1.7D	Practical Termination Tests for Iterative Algorithms	34
1.7E	A FORTRAN Program for Repeated Substitution	35
1.7F	Linear and Quadratic Convergence	36

## 2 Numerical Methods for Finding Roots and Solving Equations 43

<b>2.0</b>	<b>Introduction</b>	43
<b>2.1</b>	<b>Solving Equations and Finding Roots</b>	44
2.1A	Converting $g(x) = h(x)$ to an Equivalent Equation $f(x) = 0$	44
2.1B	Finding Initial Guesses for Iterative Root-Finding Algorithms	46
2.1C	Troublesome Roots	46
<b>2.2</b>	<b>Methods for Finding Roots of <math>f(x)</math></b>	47
2.2A	The Slope Method Strategy	47
2.2B	The Newton–Raphson (NR) Method: $m_k = m_{\tan}$	48
2.2C	Rearranging Equations to Make the NR Method Easier to Use	51
2.2D	Finding Fixed Points of $g(x)$ as Roots of $g(x) - x$	52
2.2E	The Secant (SEC) Method: $m_k = m_{\text{sec}}$	53
2.2F	Finding Initial Guesses for the SEC Method	55
2.2G	What Can Go Wrong with the NR and SEC Methods?	56
2.2H	Bracketing Methods	57
<b>2.3</b>	<b>Practical Considerations: Convergence Rate and Accuracy</b>	60
2.3A	Some Examples Illustrating Convergence Rates of NR and SEC	61
2.3B	Accelerating Linear Convergence: Aitken’s Formula	67
2.3C	Summary: Root-Finding Methods Compared	69
2.3D	Finding Roots on a Computer	70
<b>2.4</b>	<b>Analytic Discussion of Convergence Rates (Optional)</b>	72
2.4A	Multiplicity of a Root	72
2.4B	Order of Convergence of an Iterative Method	74
2.4C	The Convergence Rate of NR	74
2.4D	The Convergence Rate of Repeated Substitution	75
<b>2.5</b>	<b>Finding Roots of Real Polynomials; Bairstow’s Method</b>	76
2.5A	Deflating Real Polynomials Using Quadratic Factors	76
2.5B	Synthetic Division by $q(x)$	77
2.5C	Bairstow’s Method	78
2.5D	How Inaccurate Coefficients Can Affect the Location of Roots	83
2.5E	Proof of the Bairstow’s Method Formulas	84

<b>3</b>	<b>Solving Linear Systems Exactly Using Direct Methods</b>	<b>89</b>
<b>3.0</b>	<b>Introduction</b>	<b>89</b>
<b>3.1</b>	<b>Basic Properties of Matrices</b>	<b>90</b>
3.1A	Terminology and Notation	90
3.1B	Equality, Addition, Subtraction, and Scalar Multiplication of Matrices	91
3.1C	Properties of Matrix Addition	92
3.1D	Matrix Multiplication	92
3.1E	The Identity Matrix	94
3.1F	Properties of Matrix Multiplication	94
3.1G	The Inverse $A^{-1}$ of a Nonsingular Matrix $A$	95
3.1H	Formula for $A^{-1}$ when $n = 2$	96
<b>3.2</b>	<b>Introduction to Linear Systems; Triangular Systems</b>	<b>97</b>
3.2A	Solutions of $n \times n$ Linear Systems	97
3.2B	The Augmented Matrix for a Linear System	98
3.2C	Triangular Matrices and Systems	98
3.2D	Efficient Forward Substitution	100
3.2E	Efficient Backward Substitution	101
3.2F	Counting Arithmetic Operations	103
<b>3.3</b>	<b>Factorization Methods for Solving <math>Ax = b</math></b>	<b>103</b>
3.3A	Solving $Ax = b$ when $A = LU$	103
3.3B	Compact Forward and Backward Substitution	104
3.3C	The $LU$ -Factorization Algorithm	105
3.3D	Avoiding $l_{mm} = 0$	109
3.3E	The Triangular Decomposition Algorithm	111
3.3F	The Partial Pivoting Strategy	111
3.3G	Terminology and Notation	114
3.3H	Counting Arithmetic Operations	115
3.3I	Derivation of the $LU$ -Factorization Formulas	116
3.3J	Other Factorizations	117
<b>3.4</b>	<b>Solving Linear Systems Using Gaussian Elimination</b>	<b>118</b>
3.4A	The Elementary Operations	118
3.4B	Gaussian Elimination	120
3.4C	Comparing Gaussian Elimination to Triangular Decomposition	122
3.4D	Tridiagonal Systems	124
3.4E	The Effect of a Singular Coefficient Matrix	125
<b>3.5</b>	<b>The Determinant of an <math>n \times n</math> Matrix</b>	<b>128</b>
3.5A	Definition	128
3.5B	Basic Properties of Determinants	129
3.5C	Efficient Evaluation of $\det A$	130
3.5D	Cramer's Rule	131
3.5E	Geometric Interpretation of $\det A$	132
<b>3.6</b>	<b>Finding the Inverse of an <math>n \times n</math> Matrix</b>	<b>133</b>
3.6A	Viewing $p$ Linear Systems as $AX = B$	134

- 3.6B Efficient Solution of  $AX=B$  135
- 3.6C Finding  $A^{-1}$  137
- 3.6D Counting Arithmetic Operations 138
- 3.6E Gauss–Jordan Elimination 138

## 4 Solving Linear and Nonlinear Systems Using Fixed Precision Arithmetic 147

- 4.0 Introduction 147
- 4.1 Minimizing the Effects of Roundoff 147
  - 4.1A Example 147
  - 4.1B Rationale for Effective Pivoting Strategies 149
  - 4.1C Pivoting in the Presence of Roundoff Errors 150
  - 4.1D Partial Extended Precision 152
- 4.2 Recognizing an Ill-Conditioned Coefficient Matrix 153
  - 4.2A Ill-Conditioned Matrices 153
  - 4.2B The Pivot Condition Number  $C_p(A)$  154
  - 4.2C The Condition Number  $\text{cond } A$  (Optional) 156
- 4.3 Assessing and Improving the Accuracy of a Solution 158
  - 4.3A The Residual 159
  - 4.3B Iterative Improvement 160
- 4.4 Solving  $AX = B$  on a Computer 162
  - 4.4A Suggestions for Using a Canned Program 162
  - 4.4B Performing the Triangular Decomposition Algorithm on a Computer 162
  - 4.4C Full Pivoting (FP) 164
- 4.5 Iterative Methods for Solving Linear Systems 167
  - 4.5A The Jacobi and Gauss–Seidel Methods 167
  - 4.5B Strategies for Improving the Likelihood of Convergence 171
  - 4.5C Accelerating Convergence of Gauss–Seidel Iteration 172
  - 4.5D Relaxation 175
- 4.6 Solving Nonlinear Systems of Equations 175
  - 4.6A Obtaining Graphical Estimates of Solutions of  $2 \times 2$  Systems 176
  - 4.6B The Newton–Raphson Method for Solving Nonlinear Systems 176
  - 4.6C Practical Considerations when Using NRSYS 181

## 5 Curve Fitting and Function Approximation 187

- 5.0 Introduction 187
- 5.1 Fitting Curves to Discrete Data 188
  - 5.1A Discussion of the Problem 188
  - 5.1B The Least Square Error Criterion 189
  - 5.1C Fitting a Straight Line 190
  - 5.1D A FORTRAN Subroutine for Fitting Straight Lines 193

<b>5.2</b>	<b>Fitting Two-Parameter Curves to Monotone, Convex Data</b>	193
5.2A	The Normal Equations for a Two-Parameter Guess Function	193
5.2B	Linearizing Monotone, Convex Data	195
5.2C	Computer Programs for Fitting Monotone, Convex Data	199
<b>5.3</b>	<b>Fitting Curves That Are Linear in <math>n</math> Parameters; Polynomial Fitting</b>	200
5.3A	The Normal Equations for $g(x) = \sum_{j=1}^n \gamma_j \phi_j(x)$	201
5.3B	The Polynomial Wiggle Problem	203
5.3C	Deficiencies of $E(\hat{g})$ as an Indicator of Good Fit	204
5.3D	Using the Determination Index to Assess the Fit of $\hat{g}(x)$	205
5.3E	The Importance of Sketching the Data	207
5.3F	Ill Conditioning in Polynomial Curve Fitting	209
<b>5.4</b>	<b>Polynomial and Rational Function Approximation of <math>f(x)</math></b>	212
5.4A	Taylor Polynomial Approximation	212
5.4B	Rational Function Approximation for $x \approx 0$	213
5.4C	Chebyshev Economization	217
5.4D	Least Square Approximation of $f(x)$ on $I = [a, b]$	221

## 6 Interpolation 229

<b>6.0</b>	<b>Introduction</b>	229
<b>6.1</b>	<b>The Unique Interpolating Polynomial for <math>P_k, P_{k+1}, \dots, P_{k+m}</math></b>	230
6.1A	Example (Quadratic Interpolation)	231
6.1B	The Polynomial Interpolation Problem	231
6.1C	Existence of $p_{k,k+m}(x)$ ; Lagrange's Form	232
6.1D	Uniqueness of $p_{k,k+m}(x)$	234
<b>6.2</b>	<b>Divided Differences and the Recursive Form of <math>p_{k,k+m}(x)</math></b>	236
6.2A	Divided Differences and Divided Difference Tables	236
6.2B	Recursive Formulas for $p_{k,k+m}(x)$	239
6.2C	Locating Leading Coefficients by Inspection	243
6.2D	Using a DD Table to Recognize Polynomial Data	243
6.2E	Difference Tables for Equispaced Nodes	245
6.2F	Proof That $\Delta^m y_k$ Is the Leading Coefficient of $p_{k,k+m}(x)$	248
<b>6.3</b>	<b>Practical Strategies for Polynomial Interpolation</b>	249
6.3A	Strategy for Efficient Polynomial Interpolation	249
6.3B	Accuracy of Polynomial Interpolation	252
6.3C	Error Propagation on a DD Table	252
6.3D	The Error of Polynomial Interpolation	253
6.3E	Uniform Approximation Using Chebyshev Nodes	254
6.3F	Derivation of the Formula for $E_{k,k+m}(z)$	256
<b>6.4</b>	<b>Interpolation Using Piecewise Cubic Splines</b>	257
6.4A	Piecewise Linear Interpolation	258
6.4B	Piecewise Cubic Splines	258
6.4C	Finding $q_0(x), \dots, q_{n-1}(x)$	259

- 6.4D Finding  $\sigma_0, \dots, \sigma_n$  263
- 6.4E Selecting Endpoint Strategies 265
- 6.4F Summary: Algorithm for Piecewise Cubic Spline Interpolation 265
- 6.4G Polynomial Versus Cubic Spline Interpolation 267

## 7 Numerical Methods for Differentiation and Integration273

- 7.0 Introduction 273
- 7.1 Formulas for  $f'(x)$ ; Richardson's formula 274
  - 7.1A "Big  $O$ " Notation for Describing Truncation Errors 274
  - 7.1B Using  $\Delta f(x)/h$  to Approximate  $f'(x)$  277
  - 7.1C Roundoff Versus Truncation Error: The Stepsize Dilemma 279
  - 7.1D Richardson's Improvement Formula 280
  - 7.1E Using Richardson's Formula to Remedy the Stepsize Dilemma 282
  - 7.1F The  $O(h^2)$  Central Difference Approximation  $f'(x) \approx \delta f(x)/2h$  283
  - 7.1G Desirable Stepsizes for Richardson's Formula 284
- 7.2 Approximation Formulas for  $k$ th Derivatives 285
  - 7.2A The  $O(h)$  Approximation  $f^{(k)}(x) \approx \Delta^k f(x)/h^k$  285
  - 7.2B Higher-Order Formulas for  $f^{(k)}(x)$  287
- 7.3 Introduction to Numerical Integration (Quadrature) 290
  - 7.3A Using Interpolating Polynomials to Obtain Quadrature Formulas 291
  - 7.3B Formulas for Equispaced Points 294
  - 7.3C Truncation Error of Quadrature Formulas 299
- 7.4 Composite Rules and Romberg Integration 301
  - 7.4A Composite Trapezoidal and Simpson Rules:  $T[h]$  and  $S[h]$  301
  - 7.4B Truncation Error of  $T[h]$  and  $S[h]$  304
  - 7.4C Obtaining  $S[h]$  from  $T[h]$  and  $T[2h]$  305
  - 7.4D Romberg Integration 306
- 7.5 Gauss Quadrature 309
  - 7.5A Gauss Quadrature on  $[-1, 1]$  309
  - 7.5B Gauss Quadrature on  $[a, b]$  311
  - 7.5C Choosing a Method for Estimating a Proper Integral 313
- 7.6 Discontinuities and Improper Integrals 314
  - 7.6A Removable Discontinuities on  $[a, b]$  314
  - 7.6B Integrals with Infinite Discontinuities on  $[a, b]$  316
  - 7.6C Integrals over  $[a, \infty)$ ,  $(-\infty, b]$ , and  $(-\infty, \infty)$  318
- 7.7 Multiple Integration 320
  - 7.7A General Numerical Procedure for Approximating
 
$$I = \int_a^b \int_{F_1(x)}^{F_2(x)} f(x, y) dy dx$$
 321
  - 7.7B Getting  $I = \int_a^b \int_{F_1(x)}^{F_2(x)} f(x, y) dy dx$  Accurately on a Computer 324

## 8 Numerical Methods for Ordinary Differential Equations 333

<b>8.0</b>	<b>Introduction</b>	333
<b>8.1</b>	<b>Introduction to Solving the Initial Value Problem (IVP)</b>	334
8.1A	Existence and Uniqueness of Solutions	334
8.1B	Euler's Method	337
8.1C	The Order of a Numerical Method	340
<b>8.2</b>	<b>Self-Starting Methods: Taylor and Runge–Kutta</b>	341
8.2A	Taylor's Method	342
8.2B	Second-Order Runge–Kutta Methods	344
8.2C	Higher-Order Runge–Kutta Methods	346
8.2D	Partial Extended Precision	350
<b>8.3</b>	<b>Multistep Methods (Predictor–Corrector Strategies)</b>	352
8.3A	The Adams Predictor–Corrector Method (APC4)	354
8.3B	Stepsize Control of Predictor–Corrector Methods	357
8.3C	Stability	359
8.3D	Comparing RK and PC Methods	361
<b>8.4</b>	<b>First-Order Systems and <math>n</math>th-Order IVPs</b>	362
8.4A	Notation and Terminology	362
8.4B	Vector Formulation of Numerical Methods for Solving (IVP) $_n$	363
8.4C	Solving an $n$ th-Order IVP	365
8.4D	Solving (IVP) $_n$ on a Computer	367
8.4E	Stiff Differential Equations	368
8.4F	Linearity	372
<b>8.5</b>	<b>Boundary Value Problems</b>	373
8.5A	The Shooting Method	373
8.5B	The Importance of Linearity for the Shooting Method	376
8.5C	The Finite Difference Method	379
8.5D	The Finite Difference Method when (BVP) $_2$ Is Linear	380
8.5E	Comparison of the Shooting Method and Finite Difference Method	383

## 9 Eigenvalues 391

<b>9.0</b>	<b>Introduction</b>	391
<b>9.1</b>	<b>Basic Properties of Eigenvalues and Eigenvectors</b>	392
9.1A	The Characteristic Polynomial $p_A(\lambda)$	392
9.1B	Similar Matrices and Diagonalizability	394
9.1C	Using Eigenvectors to Uncouple Linear IVPs	396
<b>9.2</b>	<b>The Power Method</b>	400
9.2A	The Power Method for Finding Dominant Eigenvalues	400
9.2B	Convergence Considerations	402
9.2C	The Inverse Power Method for Finding Smallest Eigenvalues	403

9.2D	Shifting Eigenvalues	404	
9.2E	Practical Considerations when Using the Power Method		405
9.3	Methods for Finding All Eigenpairs of a Matrix		406
9.3A	Symmetric Matrices	406	
9.3B	Orthogonal Matrices	406	
9.3C	Jacobi's Method for Symmetric Matrices	409	
9.3D	Practical Considerations for Jacobi's Method	413	
9.3E	Factorization Methods	415	
9.4	Characteristic Values and Solutions of Homogeneous BVPs		416
9.4A	Buckling of Axially Loaded Beams	417	
9.4B	Numerical Procedure for Estimating Characteristic Values	418	
9.4C	Improving Accuracy of Estimates of $\lambda$	421	
9.4D	The Sturm–Liouville Equation	421	
9.5	Using Eigenvalues to Uncover the Structure of $A$		424
9.5A	The Principal Axis Theorem	424	
9.5B	Describing $\ A\ $ and $\text{cond } A$ when $A$ Is Symmetric	426	
9.5C	Positive Definite Matrices	426	
9.5D	Relating $\text{cond } A$ to Eigenvalues when $A$ Is Not Symmetric	427	
9.5E	The Generalized Eigenvalue Problem	428	

## APPENDIXES

<b>I</b>	Using Pseudoprograms to Describe Algorithms	435
<b>II</b>	Review of the Basic Results of Calculus	443
	Bibliography	451
	Answers to Selected Exercises	453
	Index	465