

Contents

Preface

xiii

I	Preliminaries: Mathematical Modeling, Errors, Hardware, and Software	1
1	Errors and Arithmetic	5
1.1	Sources of Error	5
1.2	Computational Science and Scientific Computing	7
1.3	Computer Arithmetic	8
1.4	How Errors Propagate	14
1.5	Mini Case Study: Avoiding Catastrophic Cancellation	15
1.6	How Errors Are Measured	17
1.7	Conditioning and Stability	20
2	Sensitivity Analysis: When a Little Means a Lot	23
2.1	Sensitivity Is Measured by Derivatives.	23
2.2	Condition Numbers Give Bounds on Sensitivity.	24
2.3	Monte Carlo Experiments Can Estimate Sensitivity.	27
2.4	Confidence Intervals Give Insight into Sensitivity	28
3	Computer Memory and Arithmetic: A Look Under the Hood	31
3.1	A Motivating Example	31
3.2	Memory Management	32
3.3	Determining Hardware Parameters	34
3.4	Speed of Computer Arithmetic	36
4	Design of Computer Programs: Writing Your Legacy	39
4.1	Documentation	39
4.2	Software Design	41
4.3	Validation and Debugging	42
4.4	Efficiency	43

II	Dense Matrix Computations	45
5	Matrix Factorizations	49
5.1	Basic Tools for Matrix Manipulation: The BLAS	50
5.2	The LU and Cholesky Decompositions	52
5.3	The QR Decomposition	57
5.3.1	QR Decomposition by Givens Rotations	58
5.3.2	QR by Gram–Schmidt Orthogonalization	60
5.3.3	Computing and Using the QR Decomposition	62
5.3.4	Mini Case Study: Least Squares Data Fitting	65
5.4	The Rank-Revealing QR Decomposition (RR-QR)	67
5.5	Eigendecomposition	68
5.5.1	Computing the Eigendecomposition	68
5.5.2	Mini Case Study: Stability Analysis of a Linear Control System	71
5.5.3	Other Uses for Eigendecompositions	72
5.6	The Singular Value Decomposition (SVD)	73
5.6.1	Computing and Using the SVD	73
5.6.2	Mini Case Study: Solving Ill-Conditioned and Rank-Deficient Least Squares Problems	74
5.7	Some Matrix Tasks to Avoid	76
5.8	Summary	78
6	Case Study: Image Deblurring: I Can See Clearly Now <i>(coauthored by James G. Nagy)</i>	81
7	Case Study: Updating and DOWndating Matrix Factorizations: A Change in Plans	87
8	Case Study: The Direction-of-Arrival Problem	97
III	Optimization and Data Fitting	105
9	Numerical Methods for Unconstrained Optimization	109
9.1	Fundamentals for Unconstrained Optimization	109
9.1.1	How Do We Recognize a Solution?	110
9.1.2	Geometric Conditions for Optimality	112
9.1.3	The Basic Minimization Algorithm	113
9.2	The Model Method: Newton	114
9.2.1	How Well Does Newton’s Method Work?	116
9.2.2	Making Newton’s Method Safe: Modified Newton Methods	117
9.3	Descent Directions and Backtracking Linesearches	119
9.4	Trust Regions	121
9.5	Alternatives to Newton’s Method	122
9.5.1	Methods that Require Only First Derivatives	123
9.5.2	Low-Storage First-Derivative Methods	126
9.5.3	Methods that Require No Derivatives	129
9.6	Summary	131

10	Numerical Methods for Constrained Optimization	135
10.1	Fundamentals for Constrained Optimization	135
10.1.1	Optimality Conditions for Linear Constraints	136
10.1.2	Optimality Conditions for the General Case	138
10.2	Solving Problems with Bound Constraints	139
10.3	Solving Problems with Linear Equality Constraints: Feasible Directions	140
10.4	Barrier and Penalty Methods for General Constraints	141
10.5	Interior-Point Methods	144
10.6	Summary	147
11	Case Study: Classified Information: The Data Clustering Problem <i>(coauthored by Nargess Memarsadeghi)</i>	149
12	Case Study: Achieving a Common Viewpoint: Yaw, Pitch, and Roll <i>(coauthored by David A. Schug)</i>	157
13	Case Study: Fitting Exponentials: An Interest in Rates	163
14	Case Study: Blind Deconvolution: Errors, Errors Everywhere	169
15	Case Study: Blind Deconvolution: A Matter of Norm	175
IV	Monte Carlo Computations	183
16	Monte Carlo Principles	187
16.1	Random Numbers and Their Generation	188
16.2	Properties of Probability Distributions	190
16.3	The World Is Normal	191
16.4	Pseudorandom Numbers and Their Generation	192
16.5	Mini Case Study: Testing Random Numbers	193
17	Case Study: Monte Carlo Minimization and Counting: One, Two, Too Many <i>(coauthored by Isabel Beichl and Francis Sullivan)</i>	195
18	Case Study: Multidimensional Integration: Partition and Conquer	203
19	Case Study: Models of Infection: Person to Person	213
V	Ordinary Differential Equations	221
20	Solution of Ordinary Differential Equations	225
20.1	Initial Value Problems for Ordinary Differential Equations	226
20.1.1	Standard Form	226
20.1.2	Solution Families and Stability	228

20.2	Methods for Solving IVPs for ODEs	232
20.2.1	Euler's Method, Stability, and Error	232
20.2.2	Predictor-Corrector Methods	237
20.2.3	The Adams Family	239
20.2.4	Some Ingredients in Building a Practical ODE Solver	240
20.2.5	Solving Stiff Problems	243
20.2.6	An Alternative to Adams Formulas: Runge–Kutta	243
20.3	Hamiltonian Systems	245
20.4	Differential-Algebraic Equations	247
20.4.1	Some Basics	248
20.4.2	Numerical Methods for DAEs	249
20.5	Boundary Value Problems for ODEs	250
20.5.1	Shooting Methods	253
20.5.2	Finite Difference Methods	254
20.6	Summary	256
21	Case Study: More Models of Infection: It's Epidemic	259
22	Case Study: Robot Control: Swinging Like a Pendulum <i>(coauthored by Yalin E. Sagduyu)</i>	265
23	Case Study: Finite Differences and Finite Elements Getting to Know You	273
VI	Nonlinear Equations and Continuation Methods	281
24	Nonlinear Systems	285
24.1	The Problem	285
24.2	Nonlinear Least Squares Problems	287
24.3	Newton-like Methods	288
24.3.1	Newton's Method for Nonlinear Equations	288
24.3.2	Alternatives to Newton's Method	289
24.4	Continuation Methods	291
24.4.1	The Theory behind Continuation Methods	293
24.4.2	Following the Solution Path	294
25	Case Study: Variable-Geometry Trusses	297
26	Case Study: Beetles, Cannibalism, and Chaos	301
VII	Sparse Matrix Computations, with Application to Partial Differential Equations	307
27	Solving Sparse Linear Systems Taking the Direct Approach	311
27.1	Storing and Factoring Sparse Matrices	311
27.2	What Matrix Patterns Preserve Sparsity?	313
27.3	Representing Sparsity Structure	314

27.4	Some Reordering Strategies for Sparse Symmetric Matrices	314
27.5	Reordering Strategies for Nonsymmetric Matrices	321
28	Iterative Methods for Linear Systems	323
28.1	Stationary Iterative Methods (SIMs)	324
28.2	From SIMs to Krylov Subspace Methods	326
28.3	Preconditioning CG	328
28.4	Krylov Methods for Symmetric Indefinite Matrices and for Normal Equations	330
28.5	Krylov Methods for Nonsymmetric Matrices	331
28.6	Computing Eigendecompositions and SVDs with Krylov Methods . . .	333
29	Case Study: Elastoplastic Torsion: Twist and Stress	335
30	Case Study: Fast Solvers and Sylvester Equations Both Sides Now	341
31	Case Study: Eigenvalues: Valuable Principles	347
32	Multigrid Methods: Managing Massive Meshes	353
	Bibliography	361
	Index	373