

Contents

Preface	xiii
I The Basic Framework for Stationary Problems	1
1 Some model PDEs	3
1.1 Laplace's equation; elliptic BVPs	3
1.1.1 Physical experiments modeled by Laplace's equation	5
1.2 Other elliptic BVPs	8
1.2.1 The equations of isotropic elasticity	8
1.2.2 General linear elasticity	10
1.3 Exercises for Chapter 1	11
2 The weak form of a BVP	15
2.1 Review of vector calculus	15
2.1.1 The divergence theorem	15
2.1.2 Green's identity	17
2.1.3 Other forms of the divergence theorem and Green's identity	18
2.2 The weak form of a BVP	20
2.2.1 Minimization of energy	21
2.2.2 Relaxing the PDE	23
2.2.3 A few details about Sobolev spaces	27
2.3 The weak form for other boundary conditions and PDEs	29
2.3.1 Neumann conditions and the weak form	29
2.3.2 Mixed boundary conditions	31
2.3.3 Inhomogeneous boundary conditions	31
2.3.4 Other elliptic BVPs	33
2.4 Existence and uniqueness theory for the weak form of a BVP	35
2.4.1 Vector spaces and inner products	35
2.4.2 Hilbert spaces	39
2.4.3 Linear functionals	41
2.4.4 The Riesz representation theorem	42
2.4.5 Variational problems and the Riesz representation theorem	42

2.5	Examples of ellipticity	45
2.5.1	The model problem	45
2.5.2	The equations of isotropic elasticity	48
2.6	Variational formulation of nonsymmetric problems	51
2.7	Exercises for Chapter 2	53
3	The Galerkin method	57
3.1	The projection theorem	57
3.2	The Galerkin method for a variational problem	59
3.2.1	Another interpretation of the Galerkin method	62
3.2.2	The Galerkin method for a nonsymmetric problem	63
3.3	Exercises for Chapter 3	63
4	Piecewise polynomials and the finite element method	67
4.1	Piecewise linear functions defined on a triangular mesh	67
4.1.1	Using piecewise linear functions in Galerkin's method	70
4.1.2	The sparsity of the stiffness matrix	74
4.2	Quadratic Lagrange triangles	77
4.2.1	Continuous piecewise quadratic functions	77
4.2.2	The finite element method with quadratic Lagrange triangles	78
4.3	Cubic Lagrange triangles	80
4.3.1	Continuous piecewise cubic functions	80
4.3.2	The finite element method with cubic Lagrange triangles	83
4.4	Lagrange triangles of arbitrary degree	84
4.4.1	Hierarchical bases for finite element spaces	85
4.5	Other finite elements: Rectangles and quadrilaterals	86
4.5.1	Rectangular elements	86
4.5.2	General quadrilaterals	87
4.6	Using a reference triangle in finite element calculations	91
4.7	Isoparametric finite element methods	93
4.7.1	Isoparametric quadratic triangles	96
4.7.2	Isoparametric triangles of higher degree	100
4.8	Exercises for Chapter 4	101
5	Convergence of the finite element method	105
5.1	Approximating smooth functions by continuous piecewise linear functions	105
5.1.1	The standard refinement of a triangulation	106
5.1.2	Nondegenerate families of triangulations	106
5.1.3	Approximation by piecewise linear functions	107
5.2	Approximation by higher-order piecewise polynomials	108
5.3	Convergence in the energy norm	110
5.4	Convergence in the L^2 -norm	115
5.5	Variational crimes	118
5.5.1	Numerical integration	118

5.5.2	Outline of the analysis of the effect of quadrature	120
5.5.3	Isoparametric finite elements	121
5.6	Exercises for Chapter 5	122
 II Data Structures and Implementation		125
 6 The mesh data structure		127
6.1	Programming the finite element method	127
6.1.1	Assembling the stiffness matrix	127
6.1.2	Computing the load vector	131
6.2	The mesh data structure	134
6.2.1	The list of nodes	134
6.2.2	The list of edges	135
6.2.3	The list of elements	135
6.2.4	The list of free boundary edges	137
6.2.5	Other fields in the mesh data structure	137
6.3	The MATLAB implementation	138
6.3.1	Generating a mesh by refinement	139
6.3.2	Generating a mesh from a triangle-node list	140
6.3.3	Assessing the quality of a triangulation	142
6.3.4	Viewing a mesh	144
6.3.5	Handling a domain with a curved boundary	147
6.3.6	Viewing a piecewise linear function	148
6.3.7	MATLAB functions	150
6.3.8	A summary of the notation	151
6.4	Exercises for Chapter 6	152
 7 Programming the finite element method: Linear Lagrange triangles		155
7.1	Quadrature	155
7.1.1	Gaussian quadrature	155
7.1.2	Evaluating the standard basis functions on a triangle	162
7.1.3	Quadrature over a square	165
7.2	Assembling the stiffness matrix	166
7.3	Computing the load vector	168
7.3.1	Inhomogeneous Dirichlet conditions	169
7.3.2	Inhomogeneous Neumann conditions	170
7.4	Examples	171
7.4.1	Homogeneous boundary conditions	173
7.4.2	Inhomogeneous boundary conditions	174
7.4.3	A more realistic example	179
7.5	The MATLAB implementation	182
7.5.1	MATLAB functions	182
7.6	Exercises for Chapter 7	183

8	Lagrange triangles of arbitrary degree	187
8.1	Quadrature for higher-order elements	187
8.2	Assembling the stiffness matrix and load vector	192
8.3	Implementing the isoparametric method	195
8.3.1	Placement of nodes in the isoparametric method	199
8.4	Examples	200
8.5	The MATLAB implementation	203
8.5.1	version2	203
8.5.2	version3	205
8.6	Exercises for Chapter 8	206
9	The finite element method for general BVPs	209
9.1	Scalar BVPs	209
9.1.1	An example	212
9.2	Isotropic elasticity	213
9.3	Mesh locking	218
9.4	The MATLAB implementation	220
9.5	Exercises for Chapter 9	221
III	Solving the Finite Element Equations	223
10	Direct solution of sparse linear systems	225
10.1	The Cholesky factorization for positive definite matrices	225
10.1.1	The Cholesky factorization for dense matrices	226
10.1.2	The Cholesky factorization for banded matrices	228
10.2	Factoring general sparse matrices	229
10.3	Exercises for Chapter 10	233
11	Iterative methods: Conjugate gradients	235
11.1	The CG method	235
11.1.1	The CG algorithm	239
11.1.2	Convergence of the CG algorithm	243
11.2	Hierarchical bases for finite element spaces	244
11.2.1	Hierarchical bases for linear Lagrange triangles	244
11.2.2	Relationship between the stiffness matrices in nodal and hierarchical bases	249
11.3	The hierarchical basis CG method	250
11.4	The preconditioned CG method	252
11.4.1	Alternate derivation of PCG	253
11.4.2	Preconditioners	254
11.5	The pure Neumann problem	256
11.6	The MATLAB implementation	262
11.6.1	MATLAB functions	262
11.7	Exercises for Chapter 11	263

12	The classical stationary iterations	267
12.1	Stationary iterations	267
12.1.1	Matrix norms	268
12.1.2	Convergence of stationary iterations	270
12.2	The classical iterations	270
12.2.1	Jacobi iteration	271
12.2.2	Gauss–Seidel iteration	272
12.2.3	SOR iteration	273
12.2.4	Symmetric SOR	274
12.2.5	CG with SSOR preconditioning	275
12.3	The MATLAB implementation	276
12.3.1	MATLAB functions	276
12.4	Exercises for Chapter 12	276
13	The multigrid method	279
13.1	Stationary iterations as smoothers	279
13.1.1	The stiffness matrix for the model problem	279
13.1.2	Fourier modes and the spectral decomposition of K	281
13.1.3	Jacobi iteration	284
13.1.4	Weighted Jacobi iteration	287
13.2	The coarse grid correction algorithm	291
13.2.1	Projecting the equation onto a coarser mesh	292
13.2.2	The projected equation and the Galerkin idea	294
13.2.3	The two-grid multigrid algorithm	295
13.3	The multigrid V-cycle	296
13.3.1	W-cycles and μ -cycles	300
13.4	Full multigrid	300
13.4.1	Discretization, algebraic, and total errors	303
13.5	The MATLAB implementation	304
13.5.1	MATLAB functions	304
13.6	Exercises for Chapter 13	305
IV	Adaptive Methods	307
14	Adaptive mesh generation	309
14.1	Algorithms for local mesh refinement	311
14.1.1	Algorithms based on the standard refinement	311
14.1.2	Algorithms based on bisection	312
14.2	Selecting triangles for local refinement	315
14.3	A complete adaptive algorithm	317
14.4	The MATLAB implementation	322
14.4.1	MATLAB functions	324
14.5	Exercises for Chapter 14	325

15	Error estimators and indicators	329
15.1	An explicit error indicator based on estimating the curvature of the solution	330
15.2	An explicit error indicator based on the residual	334
15.3	The element residual error estimator	340
15.4	Some final examples	345
15.4.1	A discontinuous coefficient	346
15.4.2	A reentrant corner	346
15.4.3	Transition from Dirichlet to Neumann conditions	348
15.5	The MATLAB implementation	349
15.5.1	MATLAB functions	349
15.6	Exercises for Chapter 15	349
	Bibliography	353
	Index	357