

Contents

List of figures	xv
List of tables	xxiv
Preface	xxv
Acknowledgements	xxviii
1 Introduction to FDTD	1
1.1 The finite-difference time-domain method basic equations	2
1.2 Approximation of derivatives by finite differences	4
1.3 FDTD updating equations for three-dimensional problems	13
1.4 FDTD updating equations for two-dimensional problems	23
1.5 FDTD updating equations for one-dimensional problems	27
1.6 Exercises	32
2 Numerical stability and dispersion	33
2.1 Numerical stability	33
2.1.1 Stability in time-domain algorithm	33
2.1.2 CFL condition for the FDTD method	35
2.2 Numerical dispersion	37
2.3 Exercises	41
3 Building objects in the Yee grid	43
3.1 Definition of objects	43
3.1.1 Defining the problem space parameters	45
3.1.2 Defining the objects in the problem space	48
3.2 Material approximations	50
3.3 Subcell averaging schemes for tangential and normal components	52

3.4	Defining objects snapped to the Yee grid	55
3.4.1	Defining zero-thickness PEC objects	57
3.5	Creation of the material grid	58
3.6	Improved eight-subcell averaging	66
3.7	Exercises	66
4	Active and passive lumped elements	71
4.1	FDTD updating equations for lumped elements	71
4.1.1	Voltage source	72
4.1.2	Hard voltage source	74
4.1.3	Current source	75
4.1.4	Resistor	76
4.1.5	Capacitor	77
4.1.6	Inductor	78
4.1.7	Lumped elements distributed over a surface or within a volume	79
4.1.8	Diode	81
4.1.9	Summary	85
4.2	Definition, initialization, and simulation of lumped elements	86
4.2.1	Definition of lumped elements	86
4.2.2	Initialization of FDTD parameters and arrays	89
4.2.3	Initialization of lumped element components	90
4.2.4	Initialization of updating coefficients	97
4.2.5	Sampling electric and magnetic fields, voltages, and currents	108
4.2.6	Definition and initialization of output parameters	111
4.2.7	Running an FDTD simulation: The time-marching loop	119
4.2.8	Displaying FDTD simulation results	129
4.3	Simulation examples	132
4.3.1	A resistor excited by a sinusoidal voltage source	132
4.3.2	A diode excited by a sinusoidal voltage source	135
4.3.3	A capacitor excited by a unit-step voltage source	137
4.4	Exercises	141
5	Source waveforms and time to frequency domain transformation	143
5.1	Common source waveforms for FDTD simulations	143
5.1.1	Sinusoidal waveform	144
5.1.2	Gaussian waveform	145
5.1.3	Normalized derivative of a Gaussian waveform	148
5.1.4	Cosine-modulated Gaussian waveform	151

5.2	Definition and initialization of source waveforms for FDTD simulations	151
5.3	Transformation from time domain to frequency domain	155
5.4	Simulation examples	158
5.4.1	Recovering a time waveform from its Fourier transform	160
5.4.2	An RLC circuit excited by a cosine-modulated Gaussian waveform	162
5.5	Exercises	167
6	S-Parameters	169
6.1	Scattering parameters	169
6.2	S-Parameter calculations	170
6.3	Simulation examples	179
6.3.1	Quarter-wave transformer	179
6.4	Exercises	184
7	Perfectly matched layer absorbing boundary	185
7.1	Theory of PML	185
7.1.1	Theory of PML at the vacuum–PML interface	185
7.1.2	Theory of PML at the PML–PML interface	188
7.2	PML equations for three-dimensional problem space	191
7.3	PML loss functions	192
7.4	FDTD updating equations for PML and MATLAB® implementation	194
7.4.1	PML updating equations – two-dimensional TE_z case	194
7.4.2	PML updating equations – two-dimensional TM_z case	197
7.4.3	MATLAB® implementation of the two-dimensional FDTD method with PML	199
7.5	Simulation examples	215
7.5.1	Validation of PML performance	215
7.5.2	Electric field distribution	220
7.5.3	Electric field distribution using DFT	225
7.6	Exercises	227
8	Advanced PML formulations	229
8.1	Formulation of CPML	229
8.1.1	PML in stretched coordinates	229
8.1.2	Complex stretching variables in CFS-PML	230

8.1.3	The matching conditions at the PML–PML interface	231
8.1.4	Equations in the time domain	231
8.1.5	Discrete convolution	231
8.1.6	The recursive convolution method	232
8.2	The CPML algorithm	234
8.2.1	Updating equations for CPML	235
8.2.2	Addition of auxiliary CPML terms at respective regions	236
8.3	CPML parameter distribution	237
8.4	MATLAB® implementation of CPML in the three-dimensional FDTD method	238
8.4.1	Definition of CPML	239
8.4.2	Initialization of CPML	240
8.4.3	Application of CPML in the FDTD time-marching loop	246
8.5	Simulation examples	249
8.5.1	Microstrip low-pass filter	249
8.5.2	Microstrip branch line coupler	250
8.5.3	Characteristic impedance of a microstrip line	258
8.6	CPML in the two-dimensional FDTD method	264
8.7	MATLAB® implementation of CPML in the two-dimensional FDTD method	267
8.7.1	Definition of CPML	268
8.7.2	Initialization of CPML	268
8.7.3	Application of CPML in the FDTD time-marching loop	269
8.7.4	Validation of CPML performance	271
8.8	Auxiliary differential equation PML	273
8.8.1	Derivation of the ADE-PML formulation	273
8.8.2	MATLAB® implementation of the ADE-PML formulation	275
8.9	Exercises	275
9	Near-field to far-field transformation	279
9.1	Implementation of the surface equivalence theorem	281
9.1.1	Surface equivalence theorem	281
9.1.2	Equivalent surface currents in FDTD simulation	282
9.1.3	Antenna on infinite ground plane	285
9.2	Frequency domain near-field to far-field transformation	285
9.2.1	Time-domain to frequency-domain transformation	285
9.2.2	Vector potential approach	286
9.2.3	Polarization of radiation field	287
9.2.4	Radiation efficiency	289

9.3	MATLAB® implementation of near-field to far-field transformation	289
9.3.1	Definition of NF–FF parameters	289
9.3.2	Initialization of NF–FF parameters	290
9.3.3	NF–FF DFT during time-marching loop	293
9.3.4	Postprocessing for far-field calculation	297
9.4	Simulation examples	309
9.4.1	Inverted-F antenna	309
9.4.2	Strip-fed rectangular dielectric resonator antenna	315
9.5	Exercises	320
10	Thin-wire modeling	323
10.1	Thin-wire formulation	323
10.2	MATLAB® implementation of the thin-wire formulation	327
10.3	Simulation examples	330
10.3.1	Thin-wire dipole antenna	330
10.4	An improved thin-wire model	335
10.5	MATLAB® implementation of the improved thin-wire formulation	339
10.6	Simulation example	339
10.7	Exercises	341
11	Scattered field formulation	345
11.1	Scattered field basic equations	345
11.2	The scattered field updating equations	346
11.3	Expressions for the incident plane waves	350
11.4	MATLAB® implementation of the scattered field formulation	354
11.4.1	Definition of the incident plane wave	354
11.4.2	Initialization of the incident fields	355
11.4.3	Initialization of the updating coefficients	358
11.4.4	Calculation of the scattered fields	359
11.4.5	Postprocessing and simulation results	361
11.5	Simulation examples	365
11.5.1	Scattering from a dielectric sphere	365
11.5.2	Scattering from a dielectric cube	370
11.5.3	Reflection and transmission coefficients of a dielectric slab	376
11.6	Exercises	380

12 Total field/scattered field formulation	381
12.1 Introduction	381
12.2 MATLAB® implementation of the TF/SF formulation	386
12.2.1 Definition and initialization of incident fields	386
12.2.2 Updating incident fields	389
12.2.3 Updating fields on both sides of the TF/SF boundary	390
12.3 Simulation examples	393
12.3.1 Fields in an empty problem space	394
12.3.2 Scattering from a dielectric sphere	395
12.4 Exercises	396
13 Dispersive material modeling	397
13.1 Modeling dispersive media using ADE technique	398
13.1.1 Modeling Debye medium using ADE technique	398
13.1.2 Modeling Lorentz medium using ADE technique	400
13.1.3 Modeling Drude medium using ADE technique	401
13.2 MATLAB® implementation of ADE algorithm for Lorentz medium	402
13.2.1 Definition of Lorentz material parameters	402
13.2.2 Material grid construction for Lorentz objects	403
13.2.3 Initialization of updating coefficients	406
13.2.4 Field updates in time-marching loop	408
13.3 Simulation examples	410
13.3.1 Scattering from a dispersive sphere	410
13.4 Exercises	412
14 Analysis of periodic structures	413
14.1 Periodic boundary conditions	413
14.2 Constant horizontal wavenumber method	417
14.3 Source excitation	422
14.4 Reflection and transmission coefficients	424
14.4.1 TE mode reflection and transmission coefficients	425
14.4.2 TM mode reflection and transmission coefficients	427
14.4.3 TEM mode reflection and transmission coefficients	428
14.5 MATLAB® implementation of PBC FDTD algorithm	429
14.5.1 Definition of a PBC simulation	429

14.5.2	Initialization of PBC	431
14.5.3	PBC updates in time-marching loop	434
14.6	Simulation examples	442
14.6.1	Reflection and transmission coefficients of a dielectric slab	442
14.6.2	Reflection and transmission coefficients of a dipole FSS	443
14.6.3	Reflection and transmission coefficients of a Jerusalem-cross FSS	444
15	Nonuniform grid	447
15.1	Introduction	447
15.2	Transition between fine and coarse grid subregions	447
15.3	FDTD updating equations for the nonuniform grids	452
15.4	Active and passive lumped elements	454
15.5	Defining objects snapped to the electric field grid	457
15.6	MATLAB [®] implementation of nonuniform grids	458
15.6.1	Definition of subregions	459
15.6.2	Initialization of subregions	460
15.6.3	Initialization of updating coefficients	464
15.6.4	Initialization of time step duration	466
15.7	Simulation examples	466
15.7.1	Microstrip patch antenna	466
15.7.2	Three-pole microstrip low-pass filter	467
16	Graphics processing unit acceleration of finite-difference time-domain method	471
16.1	GPU programming using CUDA	472
16.1.1	Host and device	472
16.1.2	Thread hierarchy	474
16.1.3	Memory hierarchy	476
16.1.4	Performance optimization in CUDA	477
16.1.5	Achieving parallelism	477
16.2	CUDA implementation of two-dimensional FDTD	477
16.2.1	Coalesced global memory access	479
16.2.2	Thread to cell mapping	481
16.2.3	Use of shared memory	486
16.2.4	Optimization of number of threads	487
16.3	Performance of two-dimensional FDTD on CUDA	487

APPENDIX A	One-dimensional FDTD code	491
APPENDIX B	Convolutional perfectly-matched layer regions and associated field updates for a three-dimensional domain	495
APPENDIX C	MATLAB® code for plotting far-field patterns	505
APPENDIX D	MATLAB® GUI for project template	509
	References	511
	About the authors	519
	Index	523