

FINITE ELEMENT MODELING OF ELECTROMAGNETIC
RADIATION/SCATTERING PROBLEMS BY DOMAIN DECOMPOSITION

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

ÖZLEM ÖZGÜN

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF DOCTOR OF PHILOSOPHY
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

APRIL 2007

Approval of the Graduate School of Natural and Applied Sciences

Prof. Dr. Canan ÖZGEN
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Doctor of Philosophy.

Prof. Dr. İsmet ERKMEN
Head of Department

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Doctor of Philosophy.

Prof. Dr. Mustafa KUZUOĞLU
Supervisor

Examining Committee Members

Prof. Dr. Gönül Turhan SAYAN (METU, EEE) _____

Prof. Dr. Mustafa KUZUOĞLU (METU, EEE) _____

Asst. Prof. Dr. Vakur B. ERTÜRK (Bilkent Univ., EEE) _____

Assoc. Prof. Dr. Özlem Aydın ÇİVİ (METU, EEE) _____

Prof. Dr. Gülbin DURAL (METU, EEE) _____

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Özlem ÖZGÜN

ABSTRACT

FINITE ELEMENT MODELING OF ELECTROMAGNETIC RADIATION/SCATTERING PROBLEMS BY DOMAIN DECOMPOSITION

ÖZGÜN, Özlem

PhD, Department of Electrical and Electronics Engineering

Supervisor: Prof. Dr. Mustafa KUZUOĞLU

April 2007, 306 pages

The Finite Element Method (FEM) is a powerful numerical method to solve wave propagation problems for open-region electromagnetic radiation/scattering problems involving objects with arbitrary geometry and constitutive parameters. In high-frequency applications, the FEM requires an electrically large computational domain, implying a large number of unknowns, such that the numerical solution of the problem is not feasible even on state-of-the-art computers. An appealing way to solve a large FEM problem is to employ a Domain Decomposition Method (DDM) that allows the decomposition of a large problem into several coupled subproblems which can be solved independently, thus reducing considerably the memory storage requirements. In this thesis, two new domain decomposition algorithms (*FB-DDM* and *ILF-DDM*) are implemented for the finite element solution of electromagnetic radiation/scattering problems. For this purpose, a nodal FEM code (FEMS^{2D}) employing triangular elements and a vector FEM code (FEMS^{3D}) employing

tetrahedral edge elements have been developed for 2D and 3D problems, respectively. The unbounded domain of the radiation/scattering problem, as well as the boundaries of the subdomains in the DDMs, are truncated by the Perfectly Matched Layer (PML) absorber. The PML is implemented using two new approaches: *Locally-conformal PML* and *Multi-center PML*. These approaches are based on a locally-defined complex coordinate transformation which makes possible to handle challenging PML geometries, especially with curvature discontinuities. In order to implement these PML methods, we also introduce the concept of *complex space FEM* using elements with complex nodal coordinates. The performances of the DDMs and the PML methods are investigated numerically in several applications.

Keywords: Electromagnetic Radiation/Scattering, Finite Element Method, Domain Decomposition Method, Perfectly Matched Layer

ÖZ

ELEKTROMANYETİK IŞINIM/SAÇILIM PROBLEMLERİNİN BÖLGE AYRIŞIMLI SONLU ELEMANLAR YÖNTEMİYLE MODELLENMESİ

ÖZGÜN, Özlem

Doktora, Elektrik ve Elektronik Mühendisliği Bölümü

Tez Yöneticisi: Prof. Dr. Mustafa KUZUOĞLU

Nisan 2007, 306 sayfa

Sonlu Elemanlar Yöntemi (SEY), gelişigüzel şekil ve materyal özelliklerine sahip cisimlerden oluşan açık-bölge elektromanyetik ışınlama/ışınım problemleri için geçerli olan dalga yayılım denklemlerini çözen güçlü bir sayısal yöntemdir. Yüksek frekans uygulamalarındaki problemlerin SEY ile çözümü sırasında, elektriksel olarak büyük hesaplama bölgeleri, yani çok miktarda bilinmeyen ortaya çıkmaktadır. Bu tip problemlerin sayısal olarak çözümü en son teknolojiye sahip bilgisayarlarda bile mümkün olmamaktadır. Büyük SEY problemini çözmenin bir yolu, Bölge Ayrışım Yöntemi (BAY)'ne başvurmaktır. BAY, büyük SEY problemini herbiri bağımsız olarak çözülebilen birbirine bağlı çok sayıda alt-probleme ayrıştırmakta ve böylece bellek ihtiyacını da büyük ölçüde azaltmaktadır. Bu tezde, iki ve üç boyutlu ışınlama/ışınım problemlerinin SEY ile çözümü için iki yeni BAY geliştirilmiştir (*FB-DDM* ve *ILF-DDM*). Bu amaçla, iki boyutlu problemler için üçgen elemanlar kullanan bir SEY programı (*FEMS^{2D}*); üç boyutlu problemler için ise yüzeyleri üçgen olan dört yüzlü kenar

elemanlar kullanan bir SEY programı (FEMS^{3D}) geliştirilmiştir. Işınım/saçılım probleminin sınırsız bölgesi ve BAY'lerdeki alt-bölgelerin sınırları, soğuran bir katman olan Tamamen Eşlenmiş Katman (TEK) ile sonlandırılmaktadır. TEK, iki yeni yaklaşımla dizayn edilmektedir: *Yerel-uyumlu TEK* ve *Çok-merkezli TEK*. Bu yöntemler, hesaplanması zor ve süreksiz eğriğe sahip şekiller üzerindeki konformal TEK tasarımını mümkün kılan yerel-tanımlı bir kompleks koordinat dönüşümüne dayanmaktadır. Ayrıca, bu yöntemlerin uygulanabilmesi için kompleks düğümsel koordinatlara sahip elemanlar kullanılmasına dayanan *kompleks uzay SEY* kavramı sunulmaktadır. BAY'lerin ve TEK yöntemlerinin başarısı çok sayıda uygulamada sayısal olarak incelenmektedir.

Anahtar Kelimeler: Elektromanyetik Işınım/Saçılım, Sonlu Elemanlar Yöntemi, Bölge Ayrışım Yöntemi, Tamamen Eşlenmiş Katman

“Choose a job you love, and you will never have to work a day in your life.”
Confucius

Dedicated to
“sources of my energy” in this research

ACKNOWLEDGMENTS

I would like to express my deepest gratitude to my supervisor Prof. Dr. Mustafa Kuzuođlu for his guidance, suggestions, invaluable encouragement, belief in my ability, and everlasting friendly attitude throughout the development of this thesis. He is my inspiration for what academic life should be about.

I would like to thank Prof. Dr. Gönül Turhan Sayan and Asst. Prof. Dr. Vakur Ertürk for their evaluations and concerns during the development of this thesis. I would also like to thank all committee members for reading and commenting on this thesis.

Finally, I express my special thanks to my parents and my brothers together with their families for their sincere love and moral support. I am also indebted to my brothers for their support in printing and binding this thesis.

TABLE OF CONTENTS

PLAGIARISM.....	iii
ABSTRACT	iv
ÖZ.....	vi
ACKNOWLEDGMENTS.....	ix
TABLE OF CONTENTS	x
LIST OF TABLES	xiv
LIST OF FIGURES.....	xv
LIST OF ABBREVIATIONS	xxvii
CHAPTER	
1. INTRODUCTION.....	1
1.1. Review of Numerical Methods in Electromagnetic Modeling.....	1
1.2. The Finite Element Method (FEM).....	7
1.2.1. Fundamentals of the FEM	8
1.2.2. Historical Background of the FEM	17
1.3. Challenging Issues for the FEM in Electromagnetics.....	18
1.3.1. Mesh Truncation and Perfectly Matched Layers	18
1.3.2. High-Frequency Applications and Domain Decomposition Methods	22
1.4. Contributions of this Thesis	25

1.5. Organization of this Thesis.....	26
2. 2D FINITE ELEMENT ANALYSIS OF RADIATION/SCATTERING PROBLEM	27
2.1. Derivation of the Helmholtz Equation	27
2.1.1. Scattering Problem	28
2.1.2. Radiation Problem.....	31
2.2. Weak Variational Form of the Helmholtz Equation	33
2.3. FEM Formulation using Triangular Elements.....	34
2.4. RCS Calculation in the Scattering Problem	38
3. 3D VECTOR FINITE ELEMENT ANALYSIS OF SCATTERING PROBLEM	40
3.1. Derivation of the Vector Wave Equation	40
3.2. Weak Variational Form of the Vector Wave Equation	42
3.3. FEM Formulation using Tetrahedral Edge Elements.....	43
3.4. RCS Calculation	48
4. PERFECTLY MATCHED LAYER (PML) METHODS: “LOCALLY-CONFORMAL PML” & “MULTI-CENTER PML”	49
4.1. Locally-Conformal PML Formulation	52
4.2. Multi-Center PML Formulation	56
4.3. Complex Space FEM Formulation.....	60
4.3.1. FEM Formulation in 2D Complex Space.....	62
4.3.2. FEM Formulation in 3D Complex Space.....	65
4.4. Summary of the PML Methods	70
5. DOMAIN DECOMPOSITION METHODS (DDMs): “FB-DDM” & “ILF-DDM”	73

5.1. Forward-Backward DDM (FB-DDM)	73
5.1.1. Review of the Classical Alternating Schwarz Method.....	74
5.1.2. FB-DDM Algorithm.....	76
5.1.3. Implementation Issues	84
5.1.4. Special Problems	88
5.2. Iterative Leap-field DDM (ILF-DDM)	95
5.2.1. Review of the Huygens' Principle and Radiation Integrals .	96
5.2.2. ILF-DDM Algorithm.....	99
6. NUMERICAL EXPERIMENTS (PML METHODS)	106
6.1. Determination of the Threshold Angle Criterion in the Multi-Center PML Method	107
6.2. 2D Simulations	110
6.3. 3D Simulations	160
6.3.1. Comparison of the PML Methods	160
6.3.2. Miscellaneous Examples based on the Locally-Conformal PML Method	162
7. NUMERICAL EXPERIMENTS (FB-DDM).....	174
7.1. 2D Simulations	175
7.2. 3D Simulations	190
8. NUMERICAL EXPERIMENTS (ILF-DDM)	207
8.1. 2D Simulations	208
8.2. 3D Simulations	231
8.3. Observations	243
9. CONCLUSIONS	246

REFERENCES.....	251
APPENDICES	
A. 2D FEM SOFTWARE (FEMS ^{2D}).....	261
A.1. Mesh Generation.....	261
A.2. FEM Analysis.....	270
B. 3D FEM SOFTWARE (FEMS ^{3D}).....	271
B.1. Mesh Generation.....	271
B.2. Main Body.....	290
B.3. Matrix Solver.....	294
B.3.1. Iterative Matrix Solver.....	294
B.3.2. Direct Matrix Solver.....	301
B.4. Post-Processing.....	302
VITA.....	305

LIST OF TABLES

TABLES

1.1. Classification of numerical methods.	4
6.1. Error values for different center positions in the problem of scattering by an infinitely-long PEC cylinder of rectangular cross-section.	115
7.1. Error values for circular cylinder in the first forward iteration.	180
7.2. Error values for circular cylinder with different illuminations in the first forward iteration.	181
7.3. Error values for square cylinder in the first forward iteration.	182
A.1. Mesh Generation: List of common input parameters.	263
A.2. Mesh Generation: List of major output parameters.	263
A.3. FEM Analysis: Format of the input parameters for the analysis.	270
B.1. Mesh Generation: List of common input parameters.	273
B.2. Mesh Generation: Format of the output text file.	274
B.3. Main Body: Format of the input text file for the analysis.	291
B.4. Main Body: Format of the file containing CSR arrays.	292
B.5. Main Body: Format of the file containing the matrix equation.	293
B.6. Matrix Solver: Format of the optional input file.	296
B.7. Matrix Solver: Format of the optional file related to the Jacobi and Gauss-Seidel pre-conditioners.	299
B.8. Matrix Solver: Format of the optional file related to the SSOR and ILU(0) pre-conditioners.	299
B.9. Matrix Solver: Format of the optional file related to the ILUT and ILU(k) pre-conditioners.	300
B.10. Matrix Solver: Format of the file related to the solution vector.	301
B.11. Post-Processing: Format of the input text file for the analysis.	303
B.12. Post-Processing: Format of the output text files.	304

LIST OF FIGURES

FIGURES

1.1. Commonly used finite elements: (a) 1D, (b) 2D, (c) 3D.....	10
1.2. Examples from FEM mesh: (a) 2D domain with triangular elements, (b) 3D domain with tetrahedral elements.....	10
2.1. General model of the scattering problem: (a) Original problem, (b) Problem model in FEM.....	28
2.2. General model of the radiation problem: (a) Original problem, (b) Problem model in FEM.....	32
2.3. Isoparametric mapping in 2D FEM formulation.....	36
3.1. Isoparametric mapping in 3D FEM formulation.....	45
4.1. Some challenging PML geometries, whose problematic regions are shaded by the gray-colour.....	51
4.2. Mesh cross-sections demonstrating conformal PMLs in scattering problems: (a) Conesphere, (b) Ogive.....	51
4.3. Locally-conformal PML implementation.....	52
4.4. Multi-center PML implementation.....	56
4.5. (a) The partial cross section of a conical PML region whose internal angle is less than 90° , (b) Slightly cut PML region.	60
4.6. Mapping of triangular elements to complex triangular elements.....	62
4.7. Isoparametric mapping in 2D FEM formulation.....	63
4.8. Mapping of tetrahedral elements to complex tetrahedral elements.....	66
4.9. Isoparametric mapping in 3D FEM formulation.....	67
4.10. Classical PML implementations: (a) Cartesian, (b) Cylindrical.	71
5.1. Schwarz's original figure.	75
5.2. FB-DDM for cube with two subdomains: (a) Original problem, (b) Partitioned problem, (c) First subdomain with additional PML region.....	76
5.3. FB-DDM for sphere with two subdomains: (a) Original problem, (b) Partitioned problem, (c) First subdomain with additional PML region.....	77

5.4. FB-DDM for cube with three subdomains: (a) Partitioned problem, (b-c) First and second subdomains with additional PML regions.	79
5.5. FB-DDM for sphere with three subdomains: (a) Partitioned problem, (b-c) First and second subdomains with additional PML regions.	80
5.6. Flowchart of the FB-DDM algorithm with N subdomains.	84
5.7. Scattering from circular cylinder: (a) Partitioned problem, (b) First subdomain with additional PML region, (c) Reflected rays and the rays of shadow forming beam, (d) Diffracted rays emanating from P_1	86
5.8. Scattering from challenging cylinders: (a) T-shaped, (b) H-shaped.	88
5.9. FB-DDM for inlet: (a) Original problem, (b) Partitioned problem (Case 1), (c) Partitioned problem with inner subdomains (Case 2).	89
5.10. Flowchart of the FB-DDM algorithm for inlets (Case 2) with N_i inner subdomains and N outer subdomains.	91
5.11. FB-DDM for plate: (a) Original problem, (b) x-partitioned problem, (c) y-partitioned problem.	92
5.12. (a) ‘Puzzle-like’ partitioning of plate with 2×2 subdomains, (b-c-d) $\Omega_{11}, \Omega_{12}, \Omega_{21}$ with additional PML regions.	94
5.13. Flowchart for ‘puzzle-like’ partitioning with $N_x \times N_y$ subdomains.	94
5.14. Huygens’ Principle: (a) Original model, (b) Equivalent model.	96
5.15. ILF-DDM for dihedral: (a) Whole problem, (b-c) Two subproblems.	100
5.16. ILF-DDM for trihedral: (a) Whole problem, (b-c-d) Three subproblems.	100
5.17. ILF-DDM for multiple objects: (a) Whole problem, (b) Subproblems.	101
5.18. Flowchart of the ILF-DDM algorithm.	103
6.1. Illustration of how to determine the threshold angle criterion via a problem of scattering by an infinitely-long PEC cylinder of square cross-section.	109
6.2. Determination of the threshold angle using the error vs $\theta_{i,\max}$ plot.	110
6.3. Mesh for circular domain (radius is 2λ).	121
6.4. RCS profile for scattering from circular PEC cylinder.	121
6.5. Magnitude of current density along circular PEC cylinder.	122
6.6. Field contours for PEC circular cylinder (magnitude): (a-b) Scattered and total fields in the locally-conformal PML, (c-d) Scattered and total fields in the multi-center PML.	122
6.7. Scattered field along the x -axis for the scattering from circular PEC cylinder: (a) Magnitude, (b) Phase.	123

6.8. Error analysis for E_2 in the Monte Carlo Simulation of the radiation problem in circular domain: (a) Error scatter plot in the locally-conformal PML, (b) Error histogram and statistics in the locally-conformal PML, (c) Error scatter plot in the multi-center PML, (d) Error histogram and statistics in the multi-center PML.....	124
6.9. Radiated field along the x -axis for the radiation problem in circular domain (line-source is located at the origin): (a) Magnitude, (b) Phase...	125
6.10. Mesh for scattering from circular dielectric cylinder (radius is 2λ).....	126
6.11. RCS profile for scattering from circular dielectric cylinder.....	126
6.12. Field contours for scattering from circular dielectric cylinder (magnitude): (a-b) Scattered and total fields in the locally-conformal PML, (c-d) Scattered and total fields in the multi-center PML.....	127
6.13. Total field inside the circular dielectric cylinder and along the x -axis: (a) Magnitude, (b) Phase.....	128
6.14. Mesh for square domain (edge-length is 10λ).....	129
6.15. RCS profile for scattering from square PEC cylinder.....	129
6.16. Magnitude of current density along square PEC cylinder.....	130
6.17. Field contours for square PEC cylinder (magnitude): (a-b) Scattered and total fields in the locally-conformal PML, (c-d) Scattered and total fields in the multi-center PML.....	130
6.18. Error analysis for E_2 in the Monte Carlo Simulation of the radiation problem in square domain: (a) Error scatter plot in the locally-conformal PML, (b) Error histogram and statistics in the locally-conformal PML, (c) Error scatter plot in the multi-center PML, (d) Error histogram and statistics in the multi-center PML.....	131
6.19. Mesh for rectangular domain ($4\lambda \times 16\lambda$).....	132
6.20. RCS profile for scattering from rectangular PEC cylinder.....	132
6.21. Magnitude of current density along rectangular PEC cylinder.....	133
6.22. Field contours for rectangular PEC cylinder (magnitude): (a-b) Scattered and total fields in the locally-conformal PML, (c-d) Scattered and total fields in the multi-center PML.....	133
6.23. Scattered field contours (magnitude) for rectangular PEC cylinder with different centers in the multi-center PML (black ellipses indicate the regions which do not satisfy the 45° criterion): (a) Four centers, (b) Two centers (C_1 and C_4), (c) Two centers (C_2 and C_3), (d) One center at the origin.....	134
6.24. Error analysis for E_2 in the Monte Carlo Simulation of the radiation problem in rectangular domain: (a) Error scatter plot in the locally-	

conformal PML, (b) Error histogram and statistics in the locally-conformal PML, (c) Error scatter plot in the multi-center PML, (d) Error histogram and statistics in the multi-center PML.....	135
6.25. Mesh for half-circular domain (radius is 2λ).	136
6.26. RCS profile for scattering from half-circular PEC cylinder.....	136
6.27. Magnitude of current density along half-circular PEC cylinder.	137
6.28. Field contours for half-circular PEC cylinder (magnitude): (a-b) Scattered and total fields in the locally-conformal PML, (c-d) Scattered and total fields in the multi-center PML.	137
6.29. Error analysis for E_2 in the Monte Carlo Simulation of the radiation problem in half-circular domain: (a) Error scatter plot in the locally-conformal PML, (b) Error histogram and statistics in the locally-conformal PML, (c) Error scatter plot in the multi-center PML, (d) Error histogram and statistics in the multi-center PML.....	138
6.30. Mesh for elliptical domain (semi-major axis is 2λ , axial ratio is 2).	139
6.31. RCS profile for scattering from elliptical PEC cylinder.	139
6.32. Magnitude of current density along elliptical PEC cylinder.	140
6.33. Field contours for elliptical PEC cylinder (magnitude): (a-b) Scattered and total fields in the locally-conformal PML, (c-d) Scattered and total fields in the multi-center PML.....	140
6.34. Error analysis for E_2 in the Monte Carlo Simulation of the radiation problem in elliptical domain: (a) Error scatter plot in the locally-conformal PML, (b) Error histogram and statistics in the locally-conformal PML, (c) Error scatter plot in the multi-center PML, (d) Error histogram and statistics in the multi-center PML.....	141
6.35. Mesh for ogive domain (max. diameter is 2λ , nose-angle is 75°).....	142
6.36. RCS profile for scattering from ogive PEC cylinder.	142
6.37. Magnitude of current density along ogive PEC cylinder.	143
6.38. Field contours for ogive PEC cylinder (magnitude): (a-b) Scattered and total fields in the locally-conformal PML, (c-d) Scattered and total fields in the multi-center PML.....	143
6.39. Error analysis for E_2 in the Monte Carlo Simulation of the radiation problem in ogive domain: (a) Error scatter plot in the locally-conformal PML, (b) Error histogram and statistics in the locally-conformal PML, (c) Error scatter plot in the multi-center PML, (d) Error histogram and statistics in the multi-center PML.....	144
6.40. Mesh for triangular+halfcircular domain (radius 1.6λ , nose-angle 90°).....	145

6.41. RCS profile for scattering from triangular+halfcircular PEC cylinder...	145
6.42. Magnitude of current density along triangular+halfcircular PEC cylinder.	146
6.43. Field contours for triangular+halfcircular PEC cylinder (magnitude): (a-b) Scattered and total fields in the locally-conformal PML, (c-d) Scattered and total fields in the multi-center PML.	146
6.44. Error analysis for E_2 in the Monte Carlo Simulation of the radiation problem in triangular+halfcircular domain: (a) Error scatter plot in the locally-conformal PML, (b) Error histogram and statistics in the locally-conformal PML, (c) Error scatter plot in the multi-center PML, (d) Error histogram and statistics in the multi-center PML.....	147
6.45. Mesh for sharp triangular+halfcircular domain (radius 0.5λ , nose-angle 20°).	148
6.46. RCS profile for scattering from sharp triangular+halfcircular PEC cylinder.	148
6.47. Magnitude of current density along sharp triangular+halfcircular cylinder.	149
6.48. Field contours for sharp triangular+halfcircular PEC cylinder (magnitude): (a-b) Scattered and total fields in the locally-conformal PML, (c-d) Scattered and total fields in the multi-center PML.	149
6.49. Error analysis for E_2 in the Monte Carlo Simulation of the radiation problem in sharp triangular+halfcircular domain: (a) Error scatter plot in the locally-conformal PML, (b) Error histogram and statistics in the locally-conformal PML, (c) Error scatter plot in the multi-center PML, (d) Error histogram and statistics in the multi-center PML.....	150
6.50. Mesh for quadrilateral domain (vertices are shown by tick labels).	151
6.51. RCS profile for scattering from quadrilateral PEC cylinder.	151
6.52. Magnitude of current density along quadrilateral PEC cylinder.....	152
6.53. Field contours for quadrilateral PEC cylinder (magnitude): (a-b) Scattered and total fields in the locally-conformal PML, (c-d) Scattered and total fields in the multi-center PML.	152
6.54. Error analysis for E_2 in the Monte Carlo Simulation of the radiation problem in quadrilateral domain: (a) Error scatter plot in the locally-conformal PML, (b) Error histogram and statistics in the locally-conformal PML, (c) Error scatter plot in the multi-center PML, (d) Error histogram and statistics in the multi-center PML.....	153
6.55. Mesh for pentagonal domain (vertices are shown by tick labels).	154
6.56. RCS profile for scattering from pentagonal PEC cylinder.....	154

6.57. Magnitude of current density along pentagonal PEC cylinder.....	155
6.58. Field contours for pentagonal PEC cylinder (magnitude): (a-b) Scattered and total fields in the locally-conformal PML, (c-d) Scattered and total fields in the multi-center PML.	155
6.59. Error analysis for E_2 in the Monte Carlo Simulation of the radiation problem in pentagonal domain: (a) Error scatter plot in the locally- conformal PML, (b) Error histogram and statistics in the locally- conformal PML, (c) Error scatter plot in the multi-center PML, (d) Error histogram and statistics in the multi-center PML.....	156
6.60. Mesh for hexagonal domain (vertices are shown by tick labels).	157
6.61. RCS profile for scattering from hexagonal PEC cylinder.....	157
6.62. Magnitude of current density along hexagonal PEC cylinder.....	158
6.63. Field contours for hexagonal PEC cylinder (magnitude): (a-b) Scattered and total fields in the locally-conformal PML, (c-d) Scattered and total fields in the multi-center PML.	158
6.64. Error analysis for E_2 in the Monte Carlo Simulation of the radiation problem in hexagonal domain: (a) Error scatter plot in the locally- conformal PML, (b) Error histogram and statistics in the locally- conformal PML, (c) Error scatter plot in the multi-center PML, (d) Error histogram and statistics in the multi-center PML.....	159
6.65. Bistatic RCS profile of PEC sphere: (a) $\phi=0^\circ$ plane, (b) $\phi=90^\circ$ plane.....	164
6.66. Scattered farfield profile of dielectric sphere: (a) $\phi=0^\circ$ plane, (b) $\phi=90^\circ$ plane.	165
6.67. Bistatic RCS profile of PEC cube: (a) $\phi=90^\circ$ plane, (b) $\phi=0^\circ$ plane.	166
6.68. Scattered farfield profile of dielectric cube: (a) $\phi=0^\circ$ plane, (b) $\phi=90^\circ$ plane.....	167
6.69. Bistatic RCS profile of PEC prolate spheroid (semi-major axis is 0.8λ , axial ratio is 2): (a) $\phi=90^\circ$ plane, (b) $\phi=0^\circ$ plane.	168
6.70. Bistatic RCS profile of PEC conesphere (radius is 1.6λ , nose-angle 90°): (a) $\theta\theta$ polarization, (b) $\phi\phi$ polarization.....	169
6.71. Bistatic RCS profile of PEC sharp conesphere ($\theta\theta$ polarization).	170
6.72. Backscattering RCS profile of PEC finite circular cylinder ($\phi\phi$ polarization) (radius 0.3λ , height 0.6λ).	170
6.73. Monostatic RCS profile of PEC cone as a function of ka where a is the base radius of the cone (half-cone angle 20°).....	171
6.74. Monostatic RCS profile of PEC ogive as a function of the maximum diameter in wavelengths (nose angle 75°).....	171

6.75. Monostatic RCS profile of PEC cube as a function of edge length in wavelengths.	172
6.76. Bistatic RCS profile of PEC octahedron in $\phi=90^\circ$ plane.....	172
6.77. Monostatic RCS profile of PEC square plate as a function of edge length in wavelengths.	173
6.78. Backscattering RCS profile of PEC circular plate ($\theta\theta$ polarization).	173
7.1. Partitioned problem of scattering from circular cylinder (radius 15λ).	180
7.2. Bistatic RCS profile of circular PEC cylinder.....	180
7.3. Scattering by circular cylinder with different illuminations: (a) $\phi^{inc}=180^\circ$, (b) $\phi^{inc}=135^\circ$, (c) $\phi^{inc}=90^\circ$, (d) $\phi^{inc}=45^\circ$, (e) $\phi^{inc}=0^\circ$	181
7.4. Partitioned problem of scattering from square cylinder (edge-length 20λ).....	182
7.5. Bistatic RCS profile of square PEC cylinder.	182
7.6. Bistatic RCS profile of elliptical cylinder in the 1 st forward iteration (major axis is 40λ , axial ratio is 2).	183
7.7. Bistatic RCS profile of rectangular cylinder in the 1 st forward iteration ($16\lambda \times 4\lambda$).	183
7.8. Partitioned problem of scattering from T-shaped PEC cylinder.....	184
7.9. Convergence profile of T-shaped PEC cylinder.....	184
7.10. Bistatic RCS profile of T-shaped PEC cylinder.....	184
7.11. Field contours of T-shaped PEC cylinder (magnitude): (a-b) Scattered and total fields of the whole FEM, (c-d) Scattered and total fields in the 1 st forward iteration, (e-f) Scattered and total fields in the 7 th iteration, (g-h) Scattered and total fields in the 15 th iteration.	185
7.12. Partitioned problem of scattering from H-shaped PEC cylinder.....	186
7.13. Convergence profile of H-shaped PEC cylinder.	186
7.14. Bistatic RCS profile of H-shaped PEC cylinder.	186
7.15. Field contours of H-shaped PEC cylinder (magnitude): (a-b) Scattered and total fields of the whole FEM, (c-d) Scattered and total fields in the 1 st forward iteration, (e-f) Scattered and total fields in the 4 th iteration, (g-h) Scattered and total fields in the 8 th iteration.	187
7.16. Scattering from U-shaped cylinder: (a) Original problem, (b) Partitioned problem (Case 1), (c) Partitioned problem (Case 2).	188
7.17. Convergence profile in the inner part of U-shaped cylinder (Case2).....	188
7.18. Bistatic RCS profile of U-shaped PEC cylinder.	188

7.19. Field contours of U-shaped PEC cylinder (magnitude): (a-b) Scattered and total fields of the whole FEM, (c-d) Scattered and total fields in Case 1, (e-f) Scattered and total fields in Case 2.....	189
7.20. Bistatic RCS profile of sphere whose diameter is 4λ : (a) $\phi=0^\circ$ plane, (b) $\phi=90^\circ$ plane.....	194
7.21. Bistatic RCS profile of sphere whose diameter is 6λ : (a) $\phi=0^\circ$ plane, (b) $\phi=90^\circ$ plane.....	195
7.22. Bistatic RCS profile of sphere whose diameter is 8λ : (a) $\phi=0^\circ$ plane, (b) $\phi=90^\circ$ plane.....	196
7.23. Bistatic RCS profile of sphere whose diameter is 10λ : (a) $\phi=0^\circ$ plane, (b) $\phi=90^\circ$ plane.....	197
7.24. Bistatic RCS profile of cube whose edge length is 3λ : (a) $\phi=90^\circ$ plane, (b) $\phi=0^\circ$ plane.....	198
7.25. Bistatic RCS profile of cube whose edge length is 5λ : (a) $\phi=90^\circ$ plane, (b) $\phi=0^\circ$ plane.....	199
7.26. Bistatic RCS profile of conesphere (radius 1.6λ , nose angle 90°): (a) $\theta\theta$ polarization, (b) $\phi\phi$ polarization.....	200
7.27. Bistatic RCS profile of missile (diameter 1λ , length 6λ): (a) $\phi=90^\circ$ plane, (b) $\phi=0^\circ$ plane.....	201
7.28. Bistatic RCS profile of large missile (diameter 2λ , length 20λ): (a) $\phi=90^\circ$ plane, (b) $\phi=0^\circ$ plane.....	202
7.29. Bistatic RCS profile of inlet (base $1\lambda\times 1\lambda$, depth 3λ): (a) $\phi=0^\circ$ plane, (b) $\phi=90^\circ$ plane.....	203
7.30. Bistatic RCS profile of square plate whose edge length is 4λ : (a) $\phi=0^\circ$ plane, (b) $\phi=90^\circ$ plane.....	204
7.31. Bistatic RCS profile of square plate whose edge length is 10λ : (a) $\phi=0^\circ$ plane, (b) $\phi=90^\circ$ plane.....	205
7.32. Bistatic RCS profile of square plate whose edge length is 2λ for oblique incidence.....	206
7.33. Bistatic RCS profile of square plate whose edge length is 2λ for normal incidence.....	206
8.1. Original problem of example 1 (thin L-shaped cylinder, length 8λ).	212
8.2. Convergence profile of example 1.	212
8.3. Magnitude of the total current density along the boundary of the L-shaped cylinder (example 1).	212
8.4. Bistatic RCS profile of the L-shaped cylinder (example 1).	213

8.5. Field contours in example 1 (magnitude): (a-b) Scattered and total fields of the whole FEM, (c-d) Scattered and total fields in the 3 rd iteration.	213
8.6. Scattered field contours in example 1 compared to GO and UTD results (magnitude): (a) 1 st iteration of the ILF-DDM, (b) GO contribution with only 1 st order reflected field, (c) Sum of GO contribution with only 1 st order reflected field and UTD contribution with only 1 st order diffracted field, (d) 2 nd iteration of the ILF-DDM, (e) Sum of GO contributions with 1 st and 2 nd order reflected fields, and UTD contributions with 1 st and 2 nd order diffracted fields.	214
8.7. Original problem of example 2 (U-shaped cylinder).	215
8.8. Convergence profile of example 2.	215
8.9. Magnitude of the total current density along the boundary of the U-shaped cylinder (example 2).	215
8.10. Bistatic RCS profile of the U-shaped cylinder (example 2).	216
8.11. Field contours in example 2 (magnitude): (a-b) Scattered and total fields of the whole FEM, (c-d) Scattered and total fields in the 1 st iteration, (e-f) Scattered and total fields in the 10 th iteration.	216
8.12. Original problem of example 3 (short U-shaped cylinder).	217
8.13. Convergence profile of example 3.	217
8.14. Magnitude of the total current density along the boundary of the short U-shaped cylinder (example 3).	217
8.15. Bistatic RCS profile of the short U-shaped cylinder (example 3).	218
8.16. Field contours in example 3 (magnitude): (a-b) Scattered and total fields of the whole FEM, (c-d) Scattered and total fields in the 1 st iteration, (e-f) Scattered and total fields in the 4 th iteration.	218
8.17. Original problem of example 4.	219
8.18. Convergence profile of example 4.	219
8.19. Bistatic RCS profile in example 4.	219
8.20. Field contours in example 4 (magnitude): (a-b) Scattered and total fields of the whole FEM, (c-d) Scattered and total fields in the 1 st iteration, (e-f) Scattered and total fields in the 3 rd iteration.	220
8.21. Original problem of example 5.	221
8.22. Convergence profile of example 5.	221
8.23. Bistatic RCS profile in example 5.	221
8.24. Field contours in example 5 (magnitude): (a-b) Scattered and total fields of the whole FEM, (c-d) Scattered and total fields in the 1 st	

iteration, (e-f) Scattered and total fields in the 7 th iteration, (g-h) Scattered and total fields in the 15 th iteration.	222
8.25. Original problem of example 6.	223
8.26. Convergence profile of example 6.	223
8.27. Bistatic RCS profile in example 6.	223
8.28. Field contours in example 6 (magnitude): (a-b) Scattered and total fields of the whole FEM, (c-d) Scattered and total fields in the 1 st iteration, (e-f) Scattered and total fields in the 4 th iteration, (g-h) Scattered and total fields in the 8 th iteration.	224
8.29. Original problem of example 7.	225
8.30. Convergence profile of example 7.	225
8.31. Bistatic RCS profile in example 7.	225
8.32. Field contours in example 7 (magnitude): (a-b) Scattered and total fields of the whole FEM, (c-d) Scattered and total fields in the 1 st iteration, (e-f) Scattered and total fields in the 5 th iteration, (g-h) Scattered and total fields in the 10 th iteration.	226
8.33. Original problem of example 8.	227
8.34. Convergence profile of example 8.	227
8.35. Bistatic RCS profile in example 8.	227
8.36. Field contours in example 8 (magnitude): (a-b) Scattered and total fields of the whole FEM, (c-d) Scattered and total fields in the 1 st iteration, (e-f) Scattered and total fields in the 5 th iteration, (g-h) Scattered and total fields in the 10 th iteration.	228
8.37. Original problem of example 9.	229
8.38. Convergence profile of example 9.	229
8.39. Bistatic RCS profile in example 9.	229
8.40. Field contours in example 9 (magnitude): (a-b) Scattered and total fields of the whole FEM, (c-d) Scattered and total fields in the 1 st iteration, (e-f) Scattered and total fields in the 7 th iteration, (g-h) Scattered and total fields in the 15 th iteration.	230
8.41. Convergence profiles of all examples.	231
8.42. Example 1: (a) Original problem, (b) Convergence profile.	234
8.43. Bistatic RCS profile of example 1: (a) $\phi=0^\circ$ plane, (b) $\phi=90^\circ$ plane.	234
8.44. Example 2: (a) Original problem, (b) Convergence profile.	235
8.45. Bistatic RCS profile of example 2: (a) $\phi=0^\circ$ plane, (b) $\phi=90^\circ$ plane.	235

8.46. Example 3: (a) Original problem, (b) Convergence profile.	236
8.47. Bistatic RCS profile of example 3: (a) $\phi=0^\circ$ plane, (b) $\phi=90^\circ$ plane.	236
8.48. Example 4: (a) Original problem, (b) Convergence profile.	237
8.49. Bistatic RCS profile of example 4: (a) $\phi=0^\circ$ plane, (b) $\phi=90^\circ$ plane.	237
8.50. Example 5: (a) Original problem, (b) Convergence profile.	238
8.51. Bistatic RCS profile of example 5: (a) $\phi=0^\circ$ plane, (b) $\phi=90^\circ$ plane.	238
8.52. Example 6: (a) Original problem, (b) Convergence profile.	239
8.53. Bistatic RCS profile of example 6: (a) $\phi=0^\circ$ plane, (b) $\phi=90^\circ$ plane.	239
8.54. Example 7: (a) Original problem, (b) Convergence profile.	240
8.55. Bistatic RCS profile of example 7: (a) $\phi=0^\circ$ plane, (b) $\phi=90^\circ$ plane.	240
8.56. Example 8: (a) Original problem, (b) Convergence profile.	241
8.57. Bistatic RCS profile of example 8: (a) $\phi=0^\circ$ plane, (b) $\phi=90^\circ$ plane.	241
8.58. Example 9: (a) Original problem, (b) Convergence profile.	242
8.59. Bistatic RCS profile of example 9: (a) $\phi=0^\circ$ plane, (b) $\phi=90^\circ$ plane.	242
8.60. Convergence profiles of all examples.	243
A.1. Mesh Library, Circular: (a) PEC, (b) Dielectric.	264
A.2. Mesh Library, Half-circular: (a) PEC, (b) Dielectric.	264
A.3. Mesh Library, Elliptical: (a) PEC, (b) Dielectric.	265
A.4. Mesh Library, Ogive: (a) PEC, (b) Dielectric.	265
A.5. Mesh Library, Convex Polygonal: (a) PEC, (b) Dielectric.	266
A.6. Mesh Library, Concave Polygonal: (a) PEC, (b) Dielectric.	266
A.7. Mesh Library, Triangular+Halfcircular: (a) PEC, (b) Dielectric.	267
A.8. Mesh Library, Defined by $\sin(x^3) - 0.01y \leq 0$ in square domain: (a) PEC, (b) Dielectric.	267
A.9. Mesh Library, Multiple PEC scatterers.	268
A.10. Mesh partitioning of L-shaped cylinder in the ILF-DDM: (a) Original (whole) domain, (b) 2 subdomains.	268
A.11. Mesh partitioning of circular cylinder in the FB-DDM: (a) Original (whole) domain, (b) 2 subdomains, (c) 3 subdomains.	269
B.1. Mesh Library, Cube.	277
B.2. Mesh Library, Sphere.	277
B.3. Mesh Library, Ellipsoid (or spheroid).	278

B.4. Mesh Library, Circular Finite Cylinder.	278
B.5. Mesh Library, Elliptical Finite Cylinder.	279
B.6. Mesh Library, Convex Polygonal Finite Cylinder.	279
B.7. Mesh Library, Rectangular Plate.	280
B.8. Mesh Library, Circular Plate.	280
B.9. Mesh Library, Elliptical Plate.	281
B.10. Mesh Library, Polygonal Plate.	281
B.11. Mesh Library, Finite Cone with Circular-base.	282
B.12. Mesh Library, Conesphere (finite cone with spherical-base).	282
B.13. Mesh Library, Rectangular Inlet.	283
B.14. Mesh Library, Circular Inlet.	283
B.15. Mesh Library, Ogive.	284
B.16. Mesh Library, Missile.	284
B.17. Mesh Library, Pyramid with Square-base.	285
B.18. Mesh Library, Octahedron.	285
B.19. Mesh Library, Dihedral.	286
B.20. Mesh Library, Trihedral.	286
B.21. Mesh partitioning of sphere in the FB-DDM: (a) Original (whole) domain, (b) 2 subdomains, (b) 3 subdomains.	287
B.22. Mesh partitioning of cube in the FB-DDM: (a) Original (whole) domain, (b) 2 subdomains, (b) 3 subdomains.	288
B.23. Mesh partitioning of dihedral in the ILF-DDM: (a) Original (whole) domain, (b) 2 subdomains.	289
B.24. Mesh partitioning of trihedral in the ILF-DDM: (a) Original (whole) domain, (b) 3 subdomains.	289
B.25. Main Body: Example file for the input analysis file.	292
B.26. Matrix Solver: Example file for the optional input file.	298
B.27. Post Processing: Example file for the input analysis file.	304

LIST OF ABBREVIATIONS

1D	: One-dimension(al)	FETD	: Finite Element Time Domain
2D	: Two-dimension(al)	FETI	: Finite Element Tearing and Interconnecting
3D	: Three-dimension(al)	GMRES	: Generalized Minimal Residual
ABC	: Absorbing Boundary Condition	ID	: Identification Number
BC	: Boundary Condition	ILF-DDM	: Iterative Leap-field Domain Decomposition Method
BEM	: Boundary Element Method	ILU	: Incomplete LU
BiCG	: BiConjugate Gradient	LU	: Lower/Upper triangular
BiCGSTAB	: BiConjugate Gradients Stabilized	MoL	: Method of Lines
BSR	: Block Sparse Row	MoM	: Method of Moments
BVP	: Boundary Value Problem	MSR	: Modified Sparse Row
CAD	: Computer Aided Design	MS	: Mean Square
COO	: Coordinate Format	NABC	: Numerical Absorbing Boundary Condition
CG	: Conjugate Gradient	PDE	: Partial Differential Equation
CGS	: Conjugate Gradient Squared	PEC	: Perfect Electric Conductor
CPU	: Central Processing Unit	PMC	: Perfect Magnetic Conductor
CSC	: Compressed Sparse Column	PML	: Perfectly Matched Layer
CSR	: Compressed Sparse Row	PO	: Physical Optics
DDM	: Domain Decomposition Method	QMR	: Quasi-Minimal Residual
DIA	: Diagonal Format	RCS	: Radar Cross Section
EM	: Electromagnetics	RHS	: Right Hand Side
FB-DDM	: Forward-Backward Domain Decomposition Method	SSOR	: Symmetric Successive Overrelaxation
FD	: Finite Difference	TC	: Transmission Condition
FDTD	: Finite Difference Time Domain	TE	: Transverse Electric
FEM	: Finite Element Method	TLM	: Transmission Line Matrix
FEMS^{2D}	: Finite Element Software – Two Dimensional	TM	: Transverse Magnetic
FEMS^{3D}	: Finite Element Software – Three Dimensional	VOI	: Volume of Interest

CHAPTER 1

INTRODUCTION

1.1. Review of Numerical Methods in Electromagnetic Modeling

Almost every phenomenon in nature can be defined with the aid of the laws of physics, in terms of algebraic, differential or integral equations relating various quantities of interest. There are two major tasks to study physical phenomena:

- Mathematical formulation of the physical problem
- Numerical or analytical analysis of the mathematical model

The mathematical formulation of a physical problem results in mathematical expressions, often in the form of differential or integral equations relating quantities of interest in the design of the physical problem. While the derivation of the governing equations for most physical problems is not very difficult, their solution by analytical methods is generally a formidable task. In such cases, numerical methods provide alternative means of finding approximate solutions. Indeed, before computers were available, analytical solutions were obtained through procedures, like separation of variables, series expansion, conformal mapping, integral equations and variational methods. Many studies were also focused on formulating approximate and asymptotic solutions, for example in the field of diffraction problems in electromagnetics (EM). However, the invention of computers in 1940s prompted a number of researchers to develop numerical methods in solving more realistic problems which cannot be solved analytically.

An electromagnetic problem is described by means of partial differential equations (PDEs), integrodifferential equations or integral equations derived from Maxwell's equations and a set of *a priori* constraints of the problem (e.g., boundary and initial conditions, physical properties, dimensions), on an infinite-dimensional function space. Numerical methods project the solution of the continuous problem through discretization onto a finite-dimensional subspace. In other words, the original problem with infinite number of degrees of freedom is converted into one with a finite number of degrees of freedom. The set of PDEs or integral equations is converted into a system of algebraic equations, which can be solved on a computer. The parameters of interest of the problem (such as characteristic impedance matrices, transmission line parameters, or radar cross section (RCS)) are obtained from the finite number of unknowns of the discrete algebraic system of equations. Because these parameters are estimated from the solution, they are also called *a posteriori* estimates.

Nowadays, one of the major goals of the researchers in EM community is to devise CAD tools (also called electromagnetic simulators) that utilize numerical methods for the analysis, design and optimization of electromagnetic problems, as derived from Maxwell's equations, in a fast and reliable manner. The numerical methods used in a CAD tool have mainly the following steps:

- Preprocessing - The simulation domain is partitioned by curves or surfaces into a number of small grid or mesh cells. Material properties, boundary conditions and sources are also defined in this phase.
- The basic unknown parameters of the problem are the values of the unknown function at the nodes or edges of this grid or mesh cells.
- A system of equations (i.e., matrix equation system) is constructed specific to each numerical method, with the nodal or edge values as unknown parameters.

- After solving the system, the values of the unknowns are determined for each grid/mesh element, and thus, the field is actually determined in the whole domain.
- Postprocessing – Using the field values in the whole computational domain, the physical parameters of interest (S parameters, bandwidth, antenna radiation patterns, RCS, etc.) are computed.

Most common numerical methods used in the analysis of EM problems may be classified as shown in Table 1.1 [1]. The first criterion of the classification in this table is the domain in which the solution is sought (frequency or time domain). The second criterion for this classification is the type of formulation (differential equations or integral equations). This is the most traditional way of classifying numerical methods, however, a number of variations of each numerical method, and some combinations of these methods (i.e., hybrid methods) have been developed during the advancements in the numerical methods.

The numerical methods based on the integral formulation discretize certain specific integral equations involving the Green's function of the structure. These methods are best adequate for open structures. They solve for the field sources (currents and charges), unlike the PDE-based methods. The sources may be line sources (e.g., on wire antennas or scatterers), surface sources (e.g., on metallic plates), or volume sources (e.g., in homogeneous dielectric structures). The integral equation-based methods have some advantages especially for problems involving surface formulation on homogeneous bodies. In addition, the dimension of the original problem may be reduced in these methods, by means of the Fourier Transform technique which simplifies the computation of the Green's function. However, these methods also possess some certain disadvantages. First, they cannot deal with problems with highly inhomogeneous or nonlinear media, due to the lack of *a priori* knowledge of Green's functions.

Second, the analytical load of these methods is high in contrast to PDE-based formulation. Finally, although these methods have fewer unknowns as compared to the PDE-based methods, they give rise to systems of equations with dense coefficient matrices whose numerical evaluations are difficult and time consuming, due to the non-local character of the integral equations.

Table 1.1. Classification of numerical methods.

Frequency Domain		Time Domain	
Differential equations	Integral equations	Differential equations	Integral equations
<ul style="list-style-type: none"> • Finite Difference Method (FD) • Method of Lines (MoL) • Variational Methods <ul style="list-style-type: none"> - Rayleigh-Ritz - Weighted residuals <ul style="list-style-type: none"> * Collocation * Subdomain * Galerkin * Least Squares <p style="text-align: center;">⇓</p> <p>FINITE ELEMENT METHOD (FEM)</p>	<ul style="list-style-type: none"> • Method of Moments (MoM) <ul style="list-style-type: none"> * Collocation * Subdomain * Galerkin * Least Squares • Boundary Element Method (BEM) 	<ul style="list-style-type: none"> • Finite Difference Time Domain Method (FDTD) • Method of Lines (MoL) • Transmission Line Matrix Method (TLM) • Finite Element Time Domain Method (FETD) 	<ul style="list-style-type: none"> • Method of Moments (MoM)

The *Method of Moments (MoM)* is the most well-known integral equation-based numerical method, and has been introduced by Harrington [2] for the solution of electromagnetic boundary value problems (BVPs). In this method, the unknowns (i.e., the sources) are expanded by a sum of certain basis functions multiplied by unknown coefficients. Then, the residual of the integral equation is weighted using a suitable inner product and a set of weighting functions. This procedure

constructs a linear system of equations, which can be solved in a usual way. Depending on the type of weighting functions, several methods may be developed in the context of the MoM, such as collocation, subdomain, Galerkin and least squares.

The *Boundary Element Method (BEM)* [3] is based on both MoM and FEM, such that, a finite element discretization of the boundary is performed using the Green's function operator.

The numerical methods based on the PDE formulation discretize Maxwell's equations over the entire computational domain. These methods are especially suitable for closed structures which may be inhomogeneous and anisotropic. However, they can handle open structures using some specific additional techniques, such as absorbing boundary conditions, perfectly matched layers, etc. Moreover, sparse matrix structure yielding in these methods has the advantage of reducing the storage, memory requirements and CPU time.

The *Finite Difference Method (FD)* has been used in numerical analysis almost from the beginning of differential calculus, and was first applied to solve an EM problem by Ames [4]. The FD discretizes both the problem domain and differential operators by substituting the finite difference expressions of the derivatives of unknown fields at the points of the space discretization (viz., at the vertices of rectangular or cubic grids). The FD may not be suitable for problems whose geometries are arbitrarily complex, thus, the problem may need a large number of discretization points increasing the dimension of the system of equations.

The time domain variant of the FD is the *Finite Difference Time Domain Method (FDTD)*, and has been first introduced by Yee [5]. This method is based on the direct discretization of the time domain Maxwell's equations by considering the

material properties and boundary conditions. In this method, the differential operators with respect to space and time variables are discretized by using finite difference expressions, and the unknown function is calculated iteratively by marching-in-time. The computational domain is usually modeled in Cartesian coordinates, thus the curved boundaries cannot be modeled successfully due to staircase effects.

The *Method of Lines (MoL)* is a variation of the FD method, and has been introduced by Liskovets [6]. The MoL reduces the dimension of the problem into one-dimension by analytical solution. Therefore, this method can analyze only the geometries which are homogeneous in one direction.

The *Transmission Line Matrix Method (TLM)* [7] is a PDE-based numerical method, which is only specific to time domain. This method discretizes the Huygens' Principle, and solves the original electromagnetic problem by making an analogy with a distributed parameter equivalent circuit.

Variational Methods [8] involve finding the extremum of an integral depending on an unknown function and its derivatives. These methods can be divided into two categories: direct and indirect methods.

- In the direct method (also called the *Rayleigh-Ritz Method*), first the functional or integral is derived from the PDEs. The unknowns of the problem are approximated by a linear combination of suitably defined basis functions multiplied by some unknown coefficients. These coefficients are obtained by substituting the approximation of the unknowns into the functional, and minimizing it with respect to these coefficients.
- The indirect methods are commonly called as the *Method of Weighted Residuals*. The unknowns of the problem are approximated, similar to the Rayleigh-Ritz Method, by a sum of some basis functions. Then, the

residue of the PDE is weighted by a set of suitably defined weighting or test functions, and is equated to zero. Various methods have been developed depending on the type of weighting functions, and can be summarized as follows:

- *Collocation or point matching*: The weighting functions are Dirac delta functions, thus, the method forces the residue to become zero at a number of preselected points in the domain.
- *Subdomain*: The weighting function is zero in whole domain except in a given subdomain, where it may be equal to a constant, linear function or sinusoidal function.
- *Galerkin*: The weighting functions are the same as basis functions.
- *Least squares*: The weighting functions are the minimum points of the functional derived by weighting the residual by itself.
- *Petrov-Galerkin*: The weighting functions are any other functions.

1.2. The Finite Element Method (FEM)

The FEM is a variational method developed for the approximate solution of BVPs governed by PDEs, and is one of the most important developments in the history of computational methods. Although the FEM has its origin in the field of structural engineering, this technique has evolved to a widely utilized computational approach for many fields of science and engineering (such as electromagnetics, physics, acoustics, seismology and geophysics) within only a few decades, due to its two main advantages:

- Flexibility to handle any type of geometry and material inhomogeneity without a need to alter the formulation or the computer code (geometrical fidelity).
- Low memory and CPU time requirements due to its highly sparse and banded matrix structure.

1.2.1. Fundamentals of the FEM

The most distinctive feature of the FEM is the division of a given domain into a set of simple subdomains, called *finite elements*. The original domain (i.e., the FEM mesh) is then considered as an assembly of these elements connected at a finite number of preselected points, called *nodes*. The basic principle of the method is to replace the entire continuous domain with some number of finite elements in which the function is represented as a linear combination of some simple basis functions (also called shape, approximation or interpolation functions) with unknown coefficients. Hence, the solution of the entire system is approximated by a number of finite unknown coefficients. Then, the local system of algebraic relations is derived in each element by applying the Rayleigh-Ritz Method or most commonly the Weighted Residual Method (with Galerkin procedure). Finally, the local system of equations is assembled to obtain the global system of equations to solve for the unknown function. The main steps in the generation and the solution of the FEM procedure can be summarized as follows:

- Define the computational domain of the problem.
- Select the mesh truncation method in the case of open-region problems.
- Select the types of finite elements.
- Discretize the domain with finite elements and create the mesh (preprocessing).
- Select the basis or shape functions (usually linear functions or quadratic polynomials).
- Generate the element matrices by applying the Rayleigh-Ritz Method or the Weighted Residual Method to the differential equation over each element. If possible, apply an integration by parts to the corresponding integral equation to obtain the *weak variational form* of the integral.
- Assemble the local element matrices to form the sparse global system of equations and impose the boundary conditions.

- Choose the type of solver, and solve the matrix system.
- Postprocess the computed data to extract the parameters of interest.

The discretization of the entire computational domain by finite elements (i.e., mesh generation) is perhaps the most important and difficult phase of the FEM procedure, because the computer storage requirements, computation time and the accuracy of the results are highly affected by how the domain is discretized. The shape of elements should provide an easy definition of each subdomain, and also the discretization of complex geometries. Basic element shapes are triangles and tetrahedra in two- and three-dimensions (2D and 3D), respectively. These are also called simplexes, and any polygon or polyhedron can be expressed as a union of simplexes. For one-dimensional (1D) problems, the elements are often short line segments (see Figure 1.1(a)). For 2D problems, the elements are usually triangles or quadrilaterals (see Figure 1.1(b)). For 3D problems, tetrahedra and hexahedra are the most commonly used types of elements (see Figure 1.1(c)). It is worth mentioning that triangles and tetrahedra are best suited to model curved surfaces and irregular regions. Two examples are given in Figure 1.2 to illustrate the finite element discretization of a 2D and 3D domain by triangles and tetrahedral elements, respectively. When a mesh of simple elements is used to form a piecewise linear approximation of a function, the accuracy of the approximation directly depends on the sizes and shapes of the elements. If the elements are small enough (typically less than $1/10$ of a wavelength per side), the field inside the element can be reliably approximated by some linear or higher order expansion. Furthermore, the shape of each element should be as close to its ideal shape as possible. In other words, the “quality” of elements should be as good as possible in order to increase the accuracy and speed of applications. This is because there is a mathematical connection between the mesh geometry, interpolation (or discretization) errors and the global matrix conditioning [9]. “Low quality” elements may increase the condition number of the matrix (poor conditioning), and consequently, may

cause a degradation in the accuracy of the results. Although the accuracy or speed of applications can be compromised by just a few low-quality elements, mesh generation algorithms should have at least a single “quality measure” criterion to evaluate the individual elements they create. There are some efforts in meshing literature to develop some quality measures ([9]-[10]). Another important point regarding the mesh generation is that for accurate results and good rate of convergence, the FEM requires a *refined mesh* (or a large number of elements) in regions where the field does not have a smooth behaviour (i.e., corners, discontinuous boundary conditions, abrupt changes in the physical characteristics of the different media and coupling effects). Self-adaptive procedures, which are capable of deciding which parts of the domain cause the greatest error in the approximation and of improving the approximation in these parts, may be utilized in order to achieve well-adapted meshes [11].

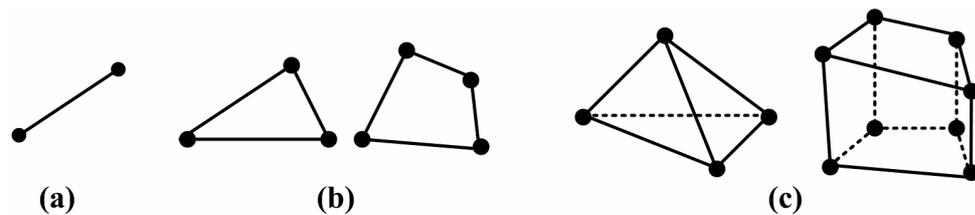


Figure 1.1. Commonly used finite elements: (a) 1D, (b) 2D, (c) 3D.

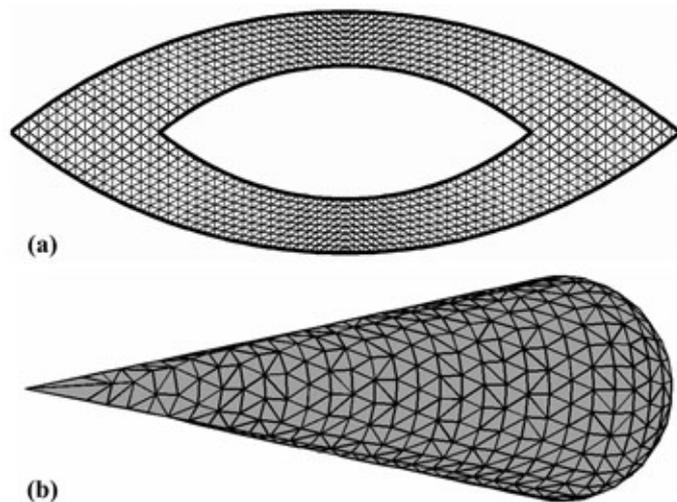


Figure 1.2. Examples from FEM mesh: (a) 2D domain with triangular elements, (b) 3D domain with tetrahedral elements.

In the FEM, the problem governed by a scalar PDE (such as the scalar Helmholtz equation) is usually formulated in terms of the values of the unknown function at nodes associated with the elements. Thus, these elements are usually called *nodal-elements*. However, if the problem involves the solution of the vector equations (like the vector wave equation), then it is usually formulated in terms of the values of the unknown function at edges associated with the elements. Therefore, these elements are referred to *edge-elements* (or *vector elements*). Edge-elements were first described by Whitney [12], and then constructed on the tetrahedron and rectangular brick by Nedelec [13]. Edge-elements are especially very useful to eliminate the spurious modes (i.e., non-physical solutions), and to handle the field singularities associated with, for instance, corners of the geometry. They only enforce the tangential continuity of the vector field, and allow the normal component of the vector to be discontinuous from one element to the next. Due to this feature, they are well suited to the EM problems, because; (i) the physical continuity requirements of the electric and magnetic fields are satisfied, (ii) the interfacial boundary conditions are automatically obtained by the natural boundary conditions, (iii) Dirichlet boundary conditions are easily imposed along curved boundaries [14].

The algebraic system of equations for the unknown coefficients of the expansion is constructed by enforcing the differential equation in a weighted sense over each element, and by applying the Rayleigh-Ritz Method or the Weighted Residual Method with Galerkin procedure. We derive here the most general form of the system of equations via Galerkin's method in order to better grasp the philosophy behind the FEM formulation [15]. A typical boundary value problem (BVP) can be described by a governing differential equation in a domain Ω as follows

$$Lu = f \tag{1.1}$$

together with the boundary conditions on the boundary Γ enclosing the domain. In (1.1), L is a differential operator, f is the excitation or forcing function, and u

is the unknown function (either scalar or vectorial). In electromagnetics, the form of governing differential equation ranges from simple equations like Poisson equation to more complicated vector wave equations.

We then derive an approximation for the unknown solution in an element (i.e., element e) as follows

$$\tilde{u}^e = \sum_{j=1}^n N_j^e u_j^e = (\bar{N}^e)^T \bar{u}^e \quad (1.2)$$

where $\bar{N}^e = [N_1^e, N_2^e, \dots, N_N^e]^T$ and $\bar{u}^e = [u_1^e, u_2^e, \dots, u_N^e]^T$. In addition, n is the number of nodes (or edges, if vector elements are used) in the element, u_j^e is the value of u at node j (or edge j) of the element, and N_j^e is the shape function for node j (or edge j). The functions N_j^e are nonzero only within the element e , and vanish outside this element.

Using (1.1), the weighted residual for the e^{th} element is derived as

$$R_i^e = \int_{\Omega^e} (L\tilde{u}^e - f) N_i^e d\Omega \quad i=1, 2, 3, \dots, n \quad (1.3)$$

Using (1.2), (1.3) becomes

$$R_i^e = \int_{\Omega^e} N_i^e L(\bar{N}^e)^T d\Omega \bar{u}^e - \int_{\Omega^e} f N_i^e d\Omega \quad i=1, 2, 3, \dots, n \quad (1.4)$$

The weak variational form of the integral equation in (1.4) may be derived at this moment by applying an integration by parts to this integral, if possible. The *weak form* is a weighted-integral statement of a differential equation in which the differentiation is distributed among the dependent variable (i.e., unknown function) and the weight function, and includes the natural boundary conditions of the problem [16]. A desirable characteristic of the weak formulation is that it requires weaker (i.e., less) continuity of the dependent variable, and usually results in a symmetric set of algebraic equations in the coefficients.

Eqn. (1.4) can be written in matrix form as

$$\bar{R}^e = [A^e] \bar{u}^e - \bar{b}^e \quad (1.5)$$

where $\bar{R}^e = [R_1^e, R_2^e, \dots, R_N^e]^T$, $[A^e]$ is a $n \times n$ local element matrix, and \bar{b}^e is a $n \times 1$ column vector. The entries $[A^e]$ and \bar{b}^e are given by

$$A_{ij}^e = \int_{\Omega^e} N_i^e L N_j^e d\Omega \quad (1.6)$$

and

$$b_i^e = \int_{\Omega^e} f N_i^e d\Omega \quad (1.7)$$

Then, the elemental equations in (1.6) and (1.7) should be summed over all elements to form the global system of equations, through a process called *assembly*. We note that the weighted residual R_i is associated with the node i (or edge i , if vector elements are used), and is a summation over the elements directly connected to the node i (or edge i). Thus, (1.5) can be expanded using the local and global relations, and summed over each element as follows

$$\bar{R} = \sum_{e=1}^M \bar{R}^e = \sum_{e=1}^M ([A^e] \bar{u}^e - \bar{b}^e) \quad (1.8)$$

where $\bar{R} = [R_1, R_2, \dots, R_N]^T$, M is the number of elements connected to each node (or edge), and N is the total number of unknowns or nodes (or edges). By setting (1.8) to equal to zero, the system of equations can be obtained by

$$\sum_{e=1}^M ([A^e] \bar{u}^e - \bar{b}^e) = 0 \quad (1.9)$$

which can be written as

$$[A] \bar{u} = \bar{b} \quad (1.10)$$

where $[A]$ is a $N \times N$ symmetric matrix, \bar{u} is a $N \times 1$ unknown vector whose entries are the unknown coefficients, and \bar{b} is a $N \times 1$ known vector.

Before solving the system of equations in (1.10), the boundary conditions (BCs) should be imposed. The type of BCs ranges from the simple Dirichlet and

Neumann conditions, to more complicated impedance and radiation conditions. In general, there are two kinds of BCs which are encountered frequently: One is the Dirichlet BC, and the other is the Neumann BC. The Dirichlet BC prescribes the field value at the boundary, and is an essential BC that must be imposed explicitly. However, the Neumann BC requires the normal derivative of the field value to vanish at the boundary, and is called a natural BC since it is satisfied implicitly and automatically in the solution phase.

One of the most important features of the FEM is that the global matrix $[A]$ in (1.10) is banded and highly sparse (i.e., containing a sufficient number of zero entries to make special processing worthwhile) because the unknowns of an element are connected only to those of contiguous elements. These characteristics make the FEM more attractive in terms of low storage, memory and CPU time requirements because only the nonzero entries need to be stored by employing some storage schemes. Typically, the FEM matrix systems have $O(N)$ storage requirements, which means that the memory needed for the solution of the matrix system is proportional to the number of unknowns N . There are various storage schemes for sparse matrices, such as Compressed Sparse Row (CSR) format, Compressed Sparse Column (CSC) format, Coordinate (COO) format, Diagonal (DIA) format, Block Sparse Row (BSR) format, and etc. [17]. One of the most common schemes in use is the CSR format. In this scheme, all the nonzero entries of the sparse matrix are stored row by row in a one-dimensional real array AA . In addition, their column indices are contained in an integer array JA as stored in the array AA , and the addresses in AA and JA of the beginning of each row are contained in a pointer integer array IA . A similar storage scheme which stores the row indices instead of the column indices is called the CSC format. The CSC format is useful when the matrix is to be accessed along the rows and not the columns (e.g., in the multiplication of the transpose of a matrix with a vector). The CSR/CSC schemes are very efficient for addition, multiplication, permutation and transposition of sparse matrices.

The final issue related to the matrix structure is that, as mentioned previously, the FEM formulation yields a banded matrix whose bandwidth is determined by the maximum difference between the global numbers of two nodes (or edges, if vector elements are used) in an element. Therefore, if a banded matrix solution technique is employed to solve the matrix equation system, the storage and CPU time can be further reduced by utilizing a proper numbering scheme to minimize the bandwidth. However, if the bandwidth minimization is unnecessary, the numbering scheme can be arbitrary.

Once the matrix equation system in (1.10) has been constructed, its solution proceeds with the application of some solution techniques. Methods used to solve matrix equations can be subdivided into two main groups: direct methods and iterative methods. Direct methods are basically based on Gaussian elimination. Among various direct methods, decomposition methods (like LU or Cholesky factorization) and Frontal / Multifrontal methods [18] are more suitable for the FEM systems. However, direct methods, especially those based on factorization schemes, may not be conveniently applied to the solution of large systems (i.e., systems with a large number of unknowns) even on state-of-the-art parallel machines, because when a sparse matrix is factorized, the upper or lower triangular factors may not reflect the sparsity pattern of the respective upper or lower triangle of the original sparse matrix. This phenomenon is called as *fill-in*, and refers to the additional nonzero elements generated during factorization. Since the matrix fill-in may cause serious storage penalties if left uncontrolled, it is necessary to implement iterative solvers whose memory requirements are a small fraction of the storage demand of the matrix. Iterative methods preserve the sparsity pattern of the matrix, and more attractively involve the matrix only in terms of matrix-vector products. Most commonly used iterative solution techniques are: Conjugate Gradient (CG) algorithm [19] and its variants (such as BiConjugate Gradient (BiCG) [20], Conjugate Gradient Squared (CGS) [21] and BiConjugate Gradients Stabilized (BiCGSTAB) [22]),

Quasi-Minimal Residual (QMR) algorithm [23], and Generalized Minimal Residual (GMRES) algorithm [24]. Basically, all iterative algorithms approach the solution through a sequence of approximations until reaching a user-specified termination criterion or a predefined maximum number of iterations. The number of iterations required for convergence is not generally known in advance, but it depends on the accuracy desired, and on the structure of the matrix (i.e., its sparsity pattern, conditioning and eigenvalue spectrum). In general, FEM matrices with a large number of unknowns are relatively ill-conditioned, and consequently, iterative solvers may require many iterations to reach a certain specified accuracy. The most common technique to accelerate the convergence is to precondition the matrix (called as *preconditioning*), which refers to improving the condition number of the matrix by premultiplying it by another matrix (i.e., $[P]^{-1}[A]\bar{u} = [P]^{-1}\bar{b}$). The matrix $[P]$ should be chosen such that (i) it should mimic the properties of $[A]$ to make $[P]^{-1}[A]$ close to the identity matrix, and (ii) it should be easily computable and invertible in $O(N)$ operations. Most widely used preconditioners include: Jacobi (or diagonal), block diagonal, Gauss Seidel, Symmetric Successive Overrelaxation (SSOR), and Incomplete-LU (ILU) techniques [25]. The simplest method is the Jacobi preconditioning, where the matrix $[P]$ is a diagonal matrix whose entries are the diagonal entries of the matrix $[A]$. The ILU preconditioner is more effective than all the others, and based on the LU factorization by ignoring all the fill-in entries. Higher values of fill-in can improve the convergence, but there is a trade-off between storage and the number of iterations for convergence.

After solving the system of equations for the unknown functions, some desired parameters (such as impedance, radiation and scattering patterns) may be calculated and displayed in a more interpretable and illustrative way using some tools, such as plots or color pictures. This final phase, which is named as *postprocessing*, may be separated from other phases of the method.

1.2.2. Historical Background of the FEM

It is not easy to trace the invention of the FEM due to different interpretations in the definition of what constitutes a finite element method. The origin of the FEM may be attributed to Hrenikoff (1941) [26], if the FEM is considered as a numerical approach where a global approximation of a PDE is obtained from a sequence of local approximations over subdomains. If the FEM is viewed as a numerical method of piecewise-continuous polynomial approximation over subdomains, the invention of the FEM may also be credited to Courant (1943) [27], who used piecewise linear approximation for a Dirichlet problem where the domain is divided into triangular elements. Although certain key features of the FEM are addressed in these works, the FEM is much more than these approximations. Any FEM consists of the representation of the domain by elements, the assembly of elements, the imposition of boundary conditions, etc. Hence, the formal presentation of the FEM is attributed to Turner et al. (1956) [28] and Argyris & Kelsey (1960) [29]. The name “finite element method” was first used by Clough (1960) [30]. After the inception of the FEM, the literature on the FEM has grown tremendously in many areas of science and engineering.

The utilization of the FEM in the area of electromagnetics, and especially in microwave and millimeter wave engineering, began in the late 1960s. The first paper dealing with an electromagnetic problem was published by Ahmed [31], who analyzed the modes propagating along homogeneous metallic waveguides. Then, Silvester [32] and Arlett et al. [33] published similar papers to analyze waveguides. The FEM has also been successfully applied to open-region problems in scattering, microwave circuits and antennas, after the developments of some concepts, such as absorbing boundary conditions, perfectly matched absorbers, infinite elements, etc. Currently, the FEM plays a fundamental role in the analysis of electromagnetic problems, and is one of the most widely used CAD tools for microwave components, antennas and scattering.

1.3. Challenging Issues for the FEM in Electromagnetics

Due to the nature of the FEM, the researchers in EM community must cope with two main issues in order to be able to solve some electromagnetic boundary value problems, especially radiation/scattering problems.

- The first issue is that since the computational domain must be bounded in the FEM, when solving open-region radiation/scattering problems, the infinite region exterior to the radiator/scatterer must be truncated by means of an artificial boundary or layer to render the problem manageable.
- The second issue is that the FEM has some drawbacks in high frequency applications where the electrical size of the computational domain increases enormously, because these problems possess a large number of unknowns and thus require large amount of memory storage which may not be available in today's technology. In this case, Domain Decomposition Methods (DDMs), where the 'large' problem is partitioned into several coupled subproblems which can be solved independently, can be utilized to resolve this difficulty.

These two challenging issues, which are also the main concerns of this thesis, are examined briefly in the following two subsections in terms of the practical and historical aspects.

1.3.1. Mesh Truncation and Perfectly Matched Layers

In open-region radiation/scattering problems where the physical domain extends to infinity, the solution of Maxwell's equations can be ensured by imposing the Sommerfeld radiation condition, which simulates the behavior of the electromagnetic field at the farfield. Since this asymptotic behavior is naturally built into the Green's functions which are the kernels of the integral operators,

there is no need to impose this condition separately in the integral equation-based numerical methods. However, in finite methods (like FEM or FDTD), the unbounded physical domain must be truncated by an artificial boundary or layer to achieve a bounded computational domain. Consequently, a boundary condition must be imposed on this artificial boundary in order to couple the computational domain to the free-space region extending to infinity. Such a boundary condition should be designed in such a way that the outgoing waves which arrive at the boundary must be absorbed with negligible reflection. In other words, this boundary condition should make the boundary appear as transparent to the fields, such that the fields in free-space should behave as if the domain extends to infinity.

In literature, there are different mesh truncation approaches, each of which has its own advantages as well as disadvantages. In this context, Absorbing Boundary Conditions (ABCs) have been developed as partial differential operators, which basically annihilate the outgoing waves. The expressions of ABCs depend on both the geometry of the artificial boundary, and the partial differential equation governing the problem. The major disadvantage of the ABCs is that the artificial boundary must be located sufficiently far away from the radiator/scatterer. This results in an increase in the size of the FEM mesh, which is undesirable. For EM problems, ABCs have been defined for planar [34], cylindrical [35], and spherical [36] boundaries to solve the wave equations.

Apart from analytically defined ABCs, Numerical Absorbing Boundary Conditions (NABCs) have been derived in terms of a linear system of equations which relate the values of the function at the boundary points to those at the neighboring points [37]-[38]. It has been shown that suitably-defined NABCs tend to be analytical ABCs on locally planar, cylindrical or spherical boundaries, as the mesh size tends to zero [39]-[40].

Another approach for mesh truncation is to combine the finite elements with elements of infinite extent (i.e., infinite elements). The infinite elements were first introduced by Bettess [41] for Laplace's equation in exterior domains. This method has also been used in electromagnetic wave scattering problems [42]. However, this approach has difficulties in terms of both the definition of basis functions having decay behavior, and the numerical integration that must be performed over the infinite element.

A very popular approach to the problem of mesh truncation is the introduction of a reflectionless absorbing layer, which is called the Perfectly Matched Layer (PML), at the outer boundary. The PML approach is basically the truncation of the computational domain by an artificial layer which absorbs outgoing plane waves irrespective of their frequency and angle of incidence, without any reflection. The most attractive feature of the PML is the ability to minimize the white space (i.e., free-space), especially in radiation and/or scattering problems, due to its close proximity and conformity to the surface of the radiator/scatterer.

The PML concept has been first introduced by Berenger [43] for electromagnetic waves in the context of the FDTD method. Berenger's PML is based on a split-field formulation of Maxwell's equations in Cartesian coordinates, and yields non-Maxwellian fields within the PML domain. Following the introduction of the PML in the FDTD method, the PML concept has been extensively used in FDTD applications [44]-[47]. A major step, which may be considered as a touchstone to start the implementation of the PML in the FEM simulations, is achieved by Sacks et al. [48], who realized the Maxwellian PML in Cartesian coordinates as an anisotropic layer with appropriately defined permittivity and permeability tensors. The anisotropic PML concept, originally introduced in Cartesian coordinates, has been extended to cylindrical and spherical coordinates [49], and has been used in the design of conformal PMLs using a local curvilinear coordinate system [50], in the context of the FEM. The anisotropic

PML absorber has also been utilized for the truncation of the FDTD lattices [51]. Then, the anisotropic PML has been investigated analytically and numerically in literature for the FEM mesh truncation [52]-[57], and the FDTD mesh truncation [58]-[60].

Another formulation, yielding a PML action, has been introduced by Chew and Weedon [61] for the electromagnetic fields in the FDTD method. This non-Maxwellian approach is implemented via the concept of complex coordinate stretching, which is essentially the analytical continuation of the field variables to complex space. It has also been shown that the anisotropic PML can be realized by applying a complex coordinate transformation in cylindrical, spherical [62], and curvilinear coordinates [63]. This is achieved through the mapping of the non-Maxwellian fields obtained during the complex coordinate transformation to a set of Maxwellian fields in an anisotropic medium representing the PML.

All of the above-mentioned PML implementations in FEM simulations are based on the design of a Maxwellian PML as an anisotropic medium having suitably defined constitutive parameters. In this thesis, however, we introduce and implement two new PML approaches: “*Locally-conformal PML*” method and “*Multi-center PML*” method [105]-[109], without using artificial anisotropic materials. These methods, which are based on a locally-defined complex coordinate transformation, are easily implementable, non-Maxwellian PML techniques for mesh truncation in FEM applications. In these methods, the analytic continuation of the frequency-domain Maxwell’s equations to complex space, due to the complex coordinate transformation, yields a PML region which absorbs outgoing waves without any reflection. Both methods are designed in complex space by keeping the same constitutive parameters everywhere and replacing the real coordinates with their complex counterparts calculated via the complex coordinate transformation. Since these approaches are based on a locally-defined complex coordinate transformation which has no explicit

dependence on the differential geometric characteristics of the PML-free space interface, it is possible to handle challenging PML geometries with interfaces having arbitrary curvature, especially those with curvature discontinuities. These new techniques are described in detail in Chapter 4, and are investigated numerically in Chapter 6.

1.3.2. High-Frequency Applications and Domain Decomposition Methods

It is well-known that achieving efficient and accurate simulations of EM problems employing electrically-large and/or geometrically complicated objects (such as the calculation of the RCS of large targets) is still a very challenging problem in some application areas, for example in the analysis and/or design of radar and communication systems. In high-frequency applications, the FEM requires an electrically-large computational domain, implying a large number of unknowns, such that the numerical solution of the problem is not feasible even on state-of-the-art computers. An appealing way to handle such a problem by circumventing this bottleneck is to employ a Domain Decomposition Method (DDM), which is based on the divide-and-conquer philosophy. A general DDM allows the decomposition of a large problem into several coupled subproblems (or subdomains) which can be solved independently and easily, thus reducing considerably the memory storage requirements. The subdomains are solved iteratively (or non-iteratively in some cases) by appropriately communicating with the others. Then, the solution of the original (whole) problem is reached by assembling the solution of all subdomains. Furthermore, a DDM is intrinsically suitable for the parallel computation of the subdomains, providing a further decrease in the overall computation time.

The perspective of the domain decomposition method itself is very broad as a research subject. Although it has been very popular in applied mathematics and

computational mechanics, it has been utilized in electromagnetic boundary value problems (BVPs) during the last few years. There are several domain decomposition strategies devised for problems governed by partial differential equations. The oldest class of the domain decomposition methods is a group of approaches known as Schwarz methods [64], developed for solving BVPs by decomposing the original spatial domain into smaller subdomains. The classical alternating Schwarz method solves several overlapping subproblems iteratively with Dirichlet boundary conditions (BCs) in each subdomain. The algorithm starts with an arbitrary initial guess for BCs (usually zero BCs), and solves the subdomains iteratively by modifying the BCs in each iteration until a certain equilibrium condition is satisfied in the solution. Overlapping Schwarz methods have also some variants, such as additive and multiplicative Schwarz methods. The number of studies based on the overlapping Schwarz methods is very limited in the EM literature. Up to now, these methods have been usually implemented in microwave interconnect structures in the static regime [65]-[66]. The original Schwarz method has been further extended to an iterative algorithm employing non-overlapping subdomains (overlap only at an interface) with transmission conditions (TCs) across the subdomain interfaces. The non-overlapping Schwarz method has been implemented in computational electromagnetics in the modeling of certain problems [67]-[70]. However, the suitable choice of TCs in the non-overlapping Schwarz method is very crucial and challenging to achieve stability and convergence in the solution.

Another domain decomposition method based on the non-overlapping subdomains is the Finite Element Tearing and Interconnecting (FETI) technique, which is an iterative substructuring algorithm based on the Lagrange multipliers [71]-[73]. In this method, the solution of each subdomain is constrained through the use of Lagrange multipliers by enforcing the continuity of the tangential fields across each boundary interface. Then, a reduced system of equations representing the Lagrange multipliers is derived and solved iteratively.

Another non-overlapping domain decomposition method is the Cement Technique, which solves the finite element problems with non-conforming/non-matching triangulations across an interface [74]-[75]. In this method, cement elements can be constructed at the interface of different types of elements, and the appropriate transmission conditions across the different triangulations are weakly enforced through the Lagrange multipliers.

Finally, there have been some efforts in the EM literature to analyze the problem of scattering from multiple objects using equivalent currents according to the Huygens' Principle. These studies employ the MoM-PO hybrid method [76] and the FDTD method [77] by dividing the original problem containing the multiple objects into smaller subproblems.

In this thesis, we introduce two alternative DDM methodologies: “*Forward-Backward DDM (FB-DDM)*”, and “*Iterative Leap-field DDM (ILF-DDM)*” [110]-[112]. Each method has some important advantages with respect to its way of implementation depending on the geometrical structure of the problem. The most distinguished feature of these fast and efficient methods is the utilization of the locally-conformal PML method along the boundaries of the subdomains. Basically, the FB-DDM is an improved version of the classical alternating Schwarz method with overlapping subdomains. This method is non-iterative in some cases involving smooth geometries, or it usually converges in a few iterations in other cases involving challenging geometries. The ILF-DDM is based on the dual employment of the FEM and the Huygens' (or equivalence) Principle iteratively. This method can be applied to *not only* cases involving multiple objects *but also* problems containing a single challenging object with the aid of the locally-conformal PML approach. These new DDM techniques are described in detail in Chapter 5, and are investigated numerically in Chapters 7 and 8.

1.4. Contributions of this Thesis

The main contribution of this thesis is to develop domain decomposition algorithms in the solution of electromagnetic problems via FEM. Efficient implementation of these algorithms requires properly and efficiently designed FEM software, which requires an appropriate mesh truncation scheme. Therefore, the contributions of this thesis, which can be arranged like the links of a chain, can be summarized as follows:

1. We have developed two domain decomposition algorithms for the finite element solution of radiation/scattering problems:

- a. Forward-Backward DDM (FB-DDM)
- b. Iterative Leap-field DDM (ILF-DDM)

We have also developed mesh partitioning schemes for above algorithms.

2. In order to implement the domain decomposition algorithms, we have developed two FEM softwares including mesh generation algorithms itself:

- a. Nodal FEM software (FEMS^{2D}) employing triangular node-based isoparametric elements to solve the scalar Helmholtz equation in 2D radiation/scattering problems.
- b. Vector FEM software (FEMS^{3D}) employing tetrahedral edge elements to solve the vector wave equation in 3D radiation/scattering problems.

3. In the finite element solution of radiation/scattering problems having unbounded domains, we have developed two mesh truncation schemes based on the concept of Perfectly Matched Layer (PML):

- a. Locally-conformal PML
- b. Multi-center PML

1.5. Organization of this Thesis

The thesis is organized as follows:

- Chapter 2 formulates the 2D radiation/scattering problems using the FEM procedure.
- Chapter 3 formulates the 3D scattering problem using the FEM procedure.
- Chapter 4 presents the two PML approaches: “Locally-conformal PML” and “Multi-center PML”.
- Chapter 5 presents the two domain decomposition methodologies: “FB-DDM” and “ILF-DDM”.
- Chapter 6 reports the results of some numerical experiments to demonstrate the performance of the PML techniques.
- Chapter 7 reports the results of some numerical experiments to demonstrate the performance of the FB-DDM algorithm.
- Chapter 8 reports the results of some numerical experiments to demonstrate the performance of the ILF-DDM algorithm.
- Chapter 9 draws conclusions in the context of the contributions of this thesis.
- Finally, Appendices A and B illustrate how to use the softwares FEMS^{2D} and FEMS^{3D}, respectively.

CHAPTER 2

2D FINITE ELEMENT ANALYSIS OF RADIATION/SCATTERING PROBLEM

In this chapter, we formulate the 2D radiation/scattering problems using the FEM procedure. The architecture of the FEM software (FEMS^{2D}) which is developed to implement these problems is presented in detail in Appendix A.

2.1. Derivation of the Helmholtz Equation

In the following two sub-sections, we briefly formulate the wave equations governing the electromagnetic radiation and scattering problems in 2D. In the scattering problem, we consider the TM_z electromagnetic scattering problem involving a single infinitely-long cylindrical PEC or dielectric scatterer of arbitrary cross-section. In the radiation problem, we deal with the problem of radiation by a single line-source inside the infinitely-long cylindrical region of arbitrary cross-section. In fact, we implement this simple radiation problem (which may also be viewed as a problem of constructing the Green's function) for the purpose of validating our new PML approaches which are introduced in Chapter 4. Thus, we also derive this radiation problem here for the sake of completeness.

Throughout the thesis, we assume that the suppressed time dependence is of the form $\exp(j\omega t)$.

2.1.1. Scattering Problem

The original picture of the scattering problem and the general setting of the problem in FEM simulation are illustrated in Figure 2.1(a) and Figure 2.1(b), respectively. Since the domain of the original problem in Figure 2.1(a) extends to infinity, we truncate the physical domain by a PML region (Ω_{PML}) which is conformal to the domain including the free-space region (Ω_{FS}) and the region occupied by the scatterer (Ω_{S}), as shown in Figure 2.1(b). The computational domain (Ω_{C}) is the union of the regions Ω_{PML} and Ω_{FS} for a PEC scatterer, and is the union of the regions Ω_{PML} , Ω_{FS} and Ω_{S} for a dielectric scatterer.

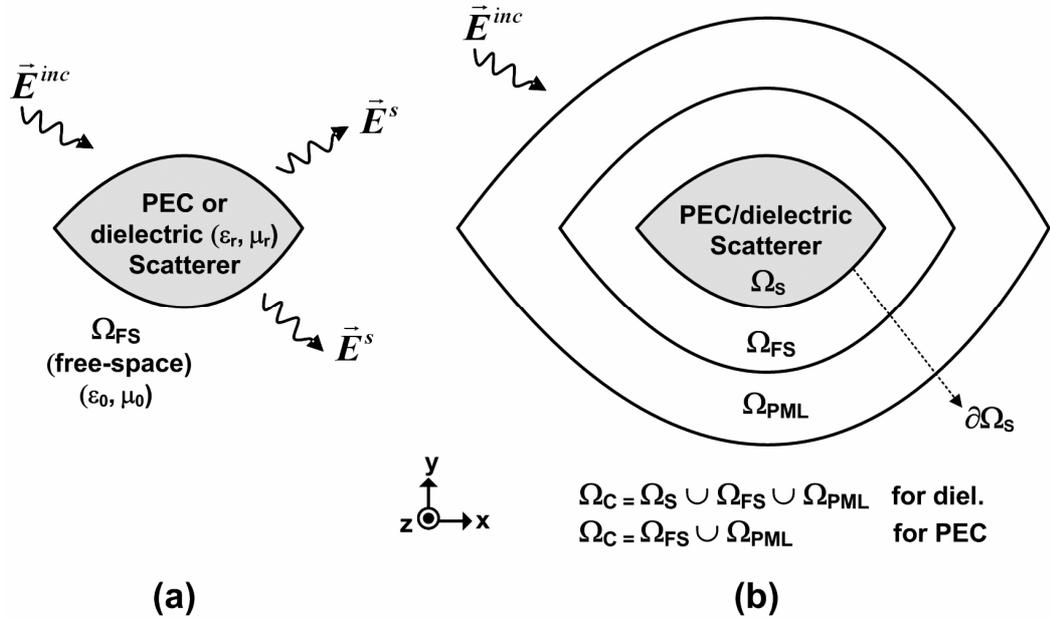


Figure 2.1. General model of the scattering problem: (a) Original problem, (b) Problem model in FEM.

In the scattering problem, an infinitely-long cylindrical PEC/dielectric scatterer of arbitrary shape and material properties is illuminated by an incident wave that does not vary parallel to the axis of the cylinder. We assume that the geometry and boundary conditions of the cylinder do not vary along the z -axis in Cartesian coordinates. In this case, any field can be decomposed into transverse magnetic

(TM) and transverse electric (TE) parts with respect to the z -variable. In the TM_z case, only the z -component of the electric field ($\vec{E} = \hat{a}_z E_z$) and x - y -components of the magnetic field ($\vec{H} = \hat{a}_x H_x + \hat{a}_y H_y$) exist. However, in the TE_z case, which is the dual of TM_z , only the z -component of the magnetic field and x - y -components of the electric field exist. Here, we assume that the cylindrical scatterer is illuminated by a TM_z plane wave.

The field in the presence of the scatterer satisfies the well-known source-free Maxwell's equations given by

$$\nabla \times \vec{E} = -j\omega\mu_0\mu_r\vec{H} \quad (2.1.a)$$

$$\nabla \times \vec{H} = j\omega\varepsilon_0\varepsilon_r\vec{E} \quad (2.1.b)$$

where ε_0 and μ_0 are the permittivity and permeability of free-space, respectively. In addition, ε_r and μ_r are the relative permittivity and permeability of the scatterer, respectively, and they are assumed to be unity inside the free-space region. By combining (2.1.a) and (2.1.b), and eliminating the magnetic field, the scalar Helmholtz equation for the z -component of the electric field (E_z) can be written as

$$\nabla \cdot \left(\frac{1}{\mu_r} \nabla E_z \right) + k^2 \varepsilon_r E_z = 0 \quad (2.2)$$

where $k = \omega\sqrt{\mu_0\varepsilon_0}$ is the free-space wave number.

The field in the presence of the scatterer can be splitted into two parts: one is the incident field ($\vec{E}^{inc} = \hat{a}_z E_z^{inc}$) produced in the absence of the scatterer, and the other is the scattered field ($\vec{E}^s = \hat{a}_z E_z^s$) produced by the equivalent induced currents, which radiate into free-space, on the scatterer or on a surface enclosing the scatterer. In other words,

$$\vec{E} = \vec{E}^{inc} + \vec{E}^s \quad (2.3)$$

For PEC scatterers, we arrive at our boundary value problem (BVP) governed by the homogeneous Helmholtz equation for the scattered field with a Dirichlet type boundary condition (BC) as follows

$$\nabla^2 E_z^s + k^2 E_z^s = 0 \quad \text{in } \Omega_{\text{FS}} \quad (2.4.a)$$

$$\text{with BC: } E_z^s = -E_z^{\text{inc}} \quad \text{on } \partial\Omega_S \quad (2.4.b)$$

Indeed, in (2.4.b), we use the fact that the total field vanishes on the boundary of the PEC scatterer ($\partial\Omega_S$).

Similarly, for dielectric scatterers, we derive the Helmholtz equation for the scattered field as follows

$$\nabla \cdot \left(\frac{1}{\mu_r} \nabla E_z^s \right) + k^2 \varepsilon_r E_z^s = k^2 (1 - \varepsilon_r) E_z^{\text{inc}} \quad (2.5)$$

It is worthwhile to mention that the right-hand-side of (2.5) exist only in the region where $\varepsilon_r \neq 1$ (only in the region of the scatterer itself), thus, (2.5) reduces to (2.4.a) in free-space region.

Although the incident field can be arbitrary, we are interested in the case of an excitation created by some source in the far-zone. Hence, we consider the incident field to be a uniform plane wave, which can be written as

$$\vec{E}^{\text{inc}} = \hat{a}_z \exp \left[jk \left(x \cos \varphi^{\text{inc}} + y \sin \varphi^{\text{inc}} \right) \right] \quad (2.6)$$

where φ^{inc} is the angle of incidence with respect to the x -axis in Cartesian coordinates.

In the scattering problem, as well as the radiation problem, the propagation of the fields away from the scatterer is guaranteed by imposing the Sommerfeld radiation condition, which is imposed at the far-zone of the scatterer and expressed in the 2D TM_z problem by

$$\lim_{r \rightarrow \infty} \sqrt{r} \left(\frac{\partial E_z^s}{\partial r} + jkE_z^s \right) = 0 \quad (2.7)$$

where r is the radial distance from the scatterer. Since the physical domain extends to infinity in the original radiation/scattering problem and we need to satisfy the Sommerfeld radiation condition, we utilize the PML approach to truncate the physical domain such that the fields in the free-space behave as though the domain extends to infinity. The construction and the formulation of the PML are fully investigated in Chapter 4, thus they are skipped here. But as a brief remark, in the PML region Ω_{PML} , we solve (2.4.a) in complex space simply by interchanging the real coordinates in Ω_{PML} by their complex counterparts, and preserving the original form of the homogeneous Helmholtz equation.

2.1.2. Radiation Problem

In the radiation problem, we consider the problem of radiation of a single line-source inside an infinitely-long cylindrical region of arbitrary shape. Similar to the scattering problem, we consider the TM_z case where only the z -component of the electric field and x - y -components of the magnetic field exist. The original unbounded model and the general model of the problem in FEM simulation are shown in Figure 2.2(a) and Figure 2.2(b), respectively. The electric line-source is assumed to be positioned at an arbitrary point \vec{r}' inside Ω_{LS} . The computational domain (Ω_{C}) is the union of the regions Ω_{PML} and Ω_{FS} .

The line-source is assumed that its length extends to infinity along the z -direction, and its electric current density is represented by

$$\vec{J} = \hat{a}_z I \delta(\vec{r} - \vec{r}') \quad (2.8)$$

where I is a constant (amp/m).

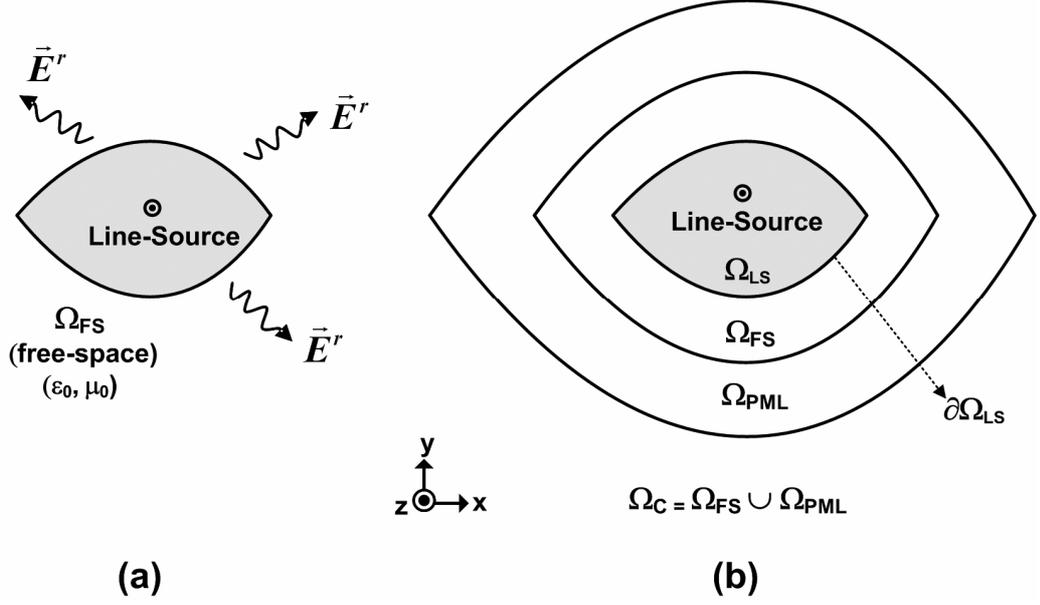


Figure 2.2. General model of the radiation problem: (a) Original problem, (b) Problem model in FEM.

Starting from the Maxwell's equations which are given by

$$\nabla \times \vec{E}^r = -j\omega\mu_0\vec{H}^r \quad (2.9.a)$$

$$\nabla \times \vec{H}^r = j\omega\varepsilon_0\vec{E}^r + \vec{J} \quad (2.9.b)$$

the scalar Helmholtz equation for the z -component of the radiated electric field (E_z^r) can be written as

$$\nabla^2 E_z^r + k^2 E_z^r = j\omega\mu_0 I \delta(\vec{r} - \vec{r}') \quad (2.10)$$

where k is the free-space wave number.

Then, we can convert (2.10) into the homogeneous Helmholtz equation with a Dirichlet type BC as follows

$$\nabla^2 E_z^r + k^2 E_z^r = 0 \quad \text{in } \Omega_{\text{FS}} \quad (2.11.a)$$

$$\text{with BC: } E_z^r = E_z^{\text{analytic}} \quad \text{on } \partial\Omega_{\text{LS}} \quad (2.11.b)$$

where E_z^{analytic} is the analytical solution of (2.10) which is given by [82]

$$E_z^{\text{analytic}}(\vec{r}) = -\frac{k\eta}{4} I H_0^{(2)}(k|\vec{r} - \vec{r}'|) \quad (2.12)$$

where $H_0^{(2)}$ is the Hankel function of the second kind of zeroth order, and η is the intrinsic impedance of the medium.

2.2. Weak Variational Form of the Helmholtz Equation

In order to formulate the finite element equations, we first derive the weak variational form of the Helmholtz equations derived in Section 2.1, using the weighted residual method.

For the scattering problem (PEC case), in order to derive the weak variational form of (2.4.a), we first take the inner product of this equation by a scalar weight function w , as follows

$$\int_{\Omega_{\text{FS}}} (\nabla^2 E_z^s) w d\Omega + k^2 \int_{\Omega_{\text{FS}}} E_z^s w d\Omega = 0 \quad (2.13)$$

Using the following identity (for any scalar function ϕ and vector function \vec{A})

$$\nabla \cdot (\phi \vec{A}) = \phi \nabla \cdot \vec{A} + \nabla \phi \cdot \vec{A} \quad (2.14)$$

and the following divergence theorem

$$\int_{\Omega} \nabla \cdot \vec{A} d\Omega = \int_{\partial\Omega} \vec{A} \cdot d\vec{s} \quad (2.15)$$

the weak variational form of (2.4.a) can be expressed as

$$\int_{\Omega_{\text{FS}}} \nabla E_z^s \cdot \nabla w d\Omega - k^2 \int_{\Omega_{\text{FS}}} E_z^s w d\Omega = 0 \quad (2.16)$$

Similarly, for the scattering problem (dielectric case), the weak variational form of (2.5) can be computed as

$$\int_{\Omega_{\text{FS}} \cup \Omega_{\text{S}}} \frac{1}{\mu_r} \nabla E_z^s \cdot \nabla w d\Omega - k^2 \int_{\Omega_{\text{FS}} \cup \Omega_{\text{S}}} \epsilon_r E_z^s w d\Omega = \int_{\Omega_{\text{S}}} k^2 (\epsilon_r - 1) E_z^{\text{inc}} w d\Omega \quad (2.17)$$

Finally, for the radiation problem, the weak variational form of (2.11.a) can be calculated in a similar manner as

$$\int_{\Omega_{FS}} \nabla E_z^r \cdot \nabla w d\Omega - k^2 \int_{\Omega_{FS}} E_z^r w d\Omega = 0 \quad (2.18)$$

2.3. FEM Formulation using Triangular Elements

In FEM, we solve the weak form of the Helmholtz equation derived in Section 2.2 by discretizing the computational domain using node-based triangular elements. First, we denote the unknown function (E_z^s in the scattering problem or E_z^r in the radiation problem) as u . Within each element, the unknown function is approximated as

$$u^e(\vec{r}) = \sum_{i=1}^3 N_i(\vec{r}) u_i^e \quad (2.19)$$

where u_i^e is the unknown function at the i^{th} node, and $N_i(\vec{r})$ is the scalar shape function (or approximation function) for the i^{th} node.

Afterwards, we substitute (2.19) into (2.16), (2.17) and (2.18), and we use the Galerkin approach where the weight functions are chosen to be equal to the shape functions (i.e., $w = N_j(\vec{r})$). Then, for the scattering problem (PEC case) and for the radiation problem, the weak variational forms in (2.16) and (2.18) become

$$\begin{aligned} & \int_{\Omega_{FS}^e} \nabla \left(\sum_{i=1}^3 N_i(\vec{r}) u_i^e \right) \cdot \nabla N_j(\vec{r}) d\Omega - \\ & k^2 \int_{\Omega_{FS}^e} \left(\sum_{i=1}^3 N_i(\vec{r}) u_i^e \right) N_j d\Omega = 0 \quad (j=1, 2, 3) \end{aligned} \quad (2.20)$$

Similarly, for the scattering problem (dielectric case), the weak variational form in (2.17) becomes

$$\begin{aligned}
& \int_{\Omega_{FS}^e \cup \Omega_S^e} \frac{1}{\mu_r^e} \nabla \left(\sum_{i=1}^3 N_i(\vec{r}) \mathbf{u}_i^e \right) \cdot \nabla N_j(\vec{r}) d\Omega \\
& - k^2 \int_{\Omega_{FS}^e \cup \Omega_S^e} \varepsilon_r^e \left(\sum_{i=1}^3 N_i(\vec{r}) \mathbf{u}_i^e \right) N_j(\vec{r}) d\Omega = \int_{\Omega_S^e} k^2 (\varepsilon_r^e - 1) E_z^{inc} N_j(\vec{r}) d\Omega \quad (2.21)
\end{aligned}$$

(j=1, 2, 3)

From (2.20), we construct a 3×3 local matrix whose ij-th entry is given by

$$a_{ij}^e = \int_{\Omega_{FS}^e} \nabla N_i(\vec{r}) \cdot \nabla N_j(\vec{r}) d\Omega - k^2 \int_{\Omega_{FS}^e} N_i(\vec{r}) N_j(\vec{r}) d\Omega \quad (2.22)$$

and we construct a 3×1 local right-hand-side (RHS) vector whose entries are all zero (i.e., $b_j^e = 0$, j=1, 2, 3).

Similarly, from (2.21), we construct a 3×3 matrix whose ij-th entry is given by

$$a_{ij}^e = \int_{\Omega_{FS}^e \cup \Omega_S^e} \frac{1}{\mu_r^e} \nabla N_i(\vec{r}) \cdot \nabla N_j(\vec{r}) d\Omega - k^2 \int_{\Omega_{FS}^e \cup \Omega_S^e} \varepsilon_r^e N_i(\vec{r}) N_j(\vec{r}) d\Omega \quad (2.23)$$

and we construct a 3×1 local RHS vector whose j-th entry is given by

$$b_j^e = \int_{\Omega_S^e} k^2 (\varepsilon_r^e - 1) E_z^{inc} N_j(\vec{r}) d\Omega \quad (2.24)$$

The integrations in (2.22), (2.23) and (2.24) are not performed directly in terms of the global coordinates, but the element is mapped to a master element in local coordinates using the “isoparametric mapping” (see Figure 2.3). In this mapping, both the global coordinates and the unknown function are expressed in terms of the same shape functions.

In each local master element, the scalar shape functions are defined as ([15])

$$N_1 = 1 - \varepsilon - \eta \quad (2.25.a)$$

$$N_2 = \varepsilon \quad (2.25.b)$$

$$N_3 = \eta \quad (2.25.c)$$

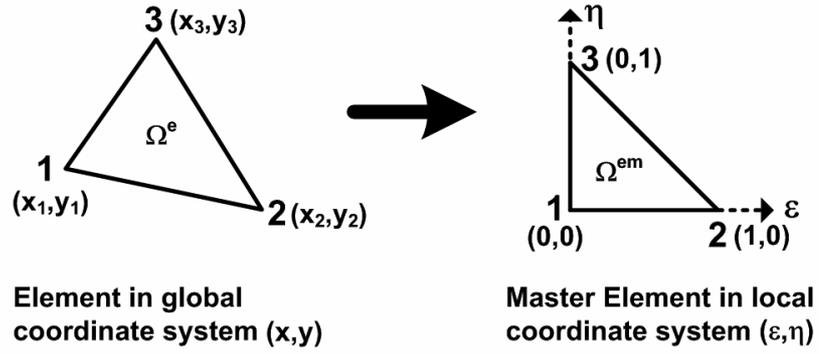


Figure 2.3. Isoparametric mapping in 2D FEM formulation.

Using the isoparametric mapping, the coordinate variable variations are expressed in terms of the scalar shape functions and the global node coordinates (in Cartesian coordinates) as follows

$$x = \sum_{i=1}^3 x_i N_i(\varepsilon, \eta) \quad (2.26.a)$$

$$y = \sum_{i=1}^3 y_i N_i(\varepsilon, \eta) \quad (2.26.b)$$

The unknown function in (2.19) is also expressed as follows

$$u^e(\varepsilon, \eta) = \sum_{i=1}^3 N_i(\varepsilon, \eta) u_i^e \quad (2.27)$$

Using the expressions in (2.26), the Jacobian matrix is calculated as

$$J_{FEM} = \begin{bmatrix} \partial x / \partial \varepsilon & \partial y / \partial \varepsilon \\ \partial x / \partial \eta & \partial y / \partial \eta \end{bmatrix} = \begin{bmatrix} x_2 - x_1 & y_2 - y_1 \\ x_3 - x_1 & y_3 - y_1 \end{bmatrix} \quad (2.28)$$

Then, using (2.25)-(2.28), the a_{ij}^e expression in (2.22) becomes

$$a_{ij}^e = \int_{\Omega_{FS}^{em}} \nabla N_i(\varepsilon, \eta) \cdot \nabla N_j(\varepsilon, \eta) \det(J_{FEM}) d\varepsilon d\eta - k^2 \int_{\Omega_{FS}^{em}} N_i(\varepsilon, \eta) N_j(\varepsilon, \eta) \det(J_{FEM}) d\varepsilon d\eta \quad (2.29)$$

Similarly, a_{ij}^e in (2.23) and b_j^e in (2.24) become, respectively, as

$$a_{ij}^e = \int_{\Omega_{FS}^{em} \cup \Omega_S^{em}} \frac{1}{\mu_r^e} \nabla N_i(\varepsilon, \eta) \cdot \nabla N_j(\varepsilon, \eta) \det(\mathbf{J}_{FEM}) d\varepsilon d\eta - k^2 \int_{\Omega_{FS}^{em} \cup \Omega_S^{em}} \varepsilon_r^e N_i(\varepsilon, \eta) N_j(\varepsilon, \eta) \det(\mathbf{J}_{FEM}) d\varepsilon d\eta \quad (2.30)$$

and

$$b_j^e = \int_{\Omega_S^{em}} k^2 (\varepsilon_r^e - 1) E_z^{inc} N_j(\varepsilon, \eta) \det(\mathbf{J}_{FEM}) d\varepsilon d\eta \quad (2.31)$$

In order to evaluate the expressions in (2.29) and (2.30), we need to calculate $\nabla N_i(\varepsilon, \eta)$ expression, which simply depends on $\nabla \varepsilon$ and $\nabla \eta$ terms. That is,

$$\nabla N_1(\varepsilon, \eta) = 1 - \nabla \varepsilon - \nabla \eta \quad (2.32.a)$$

$$\nabla N_2(\varepsilon, \eta) = \nabla \varepsilon \quad (2.32.b)$$

$$\nabla N_3(\varepsilon, \eta) = \nabla \eta \quad (2.32.c)$$

The components of these terms ($\nabla \varepsilon$ and $\nabla \eta$) are entirely determined by the inverse of the Jacobian matrix in (2.28), whose entries are given in terms of the global node coordinates, as follows

$$\begin{aligned} \frac{\partial \varepsilon}{\partial x} &= [\mathbf{J}_{FEM}^{-1}]_{1,1}, & \frac{\partial \varepsilon}{\partial y} &= [\mathbf{J}_{FEM}^{-1}]_{2,1} \\ \frac{\partial \eta}{\partial x} &= [\mathbf{J}_{FEM}^{-1}]_{1,2}, & \frac{\partial \eta}{\partial y} &= [\mathbf{J}_{FEM}^{-1}]_{2,2} \end{aligned} \quad (2.33)$$

where $[\mathbf{J}_{FEM}^{-1}]_{ij}$ is the ij -th entry of the inverse Jacobian matrix.

The integrals in (2.29)-(2.31) are evaluated using the Gaussian Quadrature integration scheme. Then, these elemental equations are assembled over all elements to form the global system of equations, by using the nodal connectivity knowledge of the elements. Before solving the system of equations, BCs in (2.4.b) and (2.11.b) are imposed. Then, the system of equations is solved for the unknown function by means of some direct or iterative solvers.

2.4. RCS Calculation in the Scattering Problem

In the scattering problem, the radar cross section (RCS), which is a description of how a scatterer reflects an incident electromagnetic wave in a given direction, is calculated in 2D as follows

$$\sigma_{2D} = \lim_{r \rightarrow \infty} 2\pi r \frac{|\vec{E}_{far}^s|^2}{|\vec{E}^{inc}|^2} \quad (2.34)$$

where \vec{E}_{far}^s and \vec{E}^{inc} are the far-zone scattered field observed in a given direction, and the incident field, respectively. In addition, r is the farfield (Rayleigh) distance given by

$$r \geq \frac{2D^2}{\lambda} \quad (2.35)$$

where D is the largest dimension of the scatterer.

If the incident and observation directions are the same, σ_{2D} is called the “monostatic” or “backscatter” RCS, otherwise, it is referred as the “bistatic” RCS.

The far-zone scattered field is obtained by the Huygens’ Principle (see Section 5.2.1), which is utilized by employing a closed surface S enclosing the scatterer in the free-space region. From the knowledge of the field values on S , the equivalent currents are calculated by

$$\vec{J} = \hat{n} \times \vec{H} \quad (2.36.a)$$

$$\vec{M} = \vec{E} \times \hat{n} \quad (2.36.b)$$

where \vec{E} and \vec{H} are the total fields on S , and \hat{n} is the unit normal vector directed outward on S . Then, the far-zone scattered field is calculated using these equivalent currents as follows

$$\vec{E}_{far}^s(\vec{r}) = jk \frac{e^{-jkr}}{4\pi r} \iint_S \left[\vec{r} \times \vec{M}(\vec{r}') + \eta_0 \vec{r} \times (\vec{r} \times \vec{J}(\vec{r}')) \right] e^{-jk(\vec{r}' \cdot \vec{r})} dS' \quad (2.37)$$

where \vec{r} and \vec{r}' are the observation and source points respectively, and η_0 is the free space impedance.

In the 2D case, the RCS (sometimes called as echo width), is usually normalized to λ (wavelength) or m (meter). The unit for σ_{2D} / λ is dBw, and for σ_{2D} / m is dBm.

CHAPTER 3

3D VECTOR FINITE ELEMENT ANALYSIS OF SCATTERING PROBLEM

In this chapter, we generalize the analysis in 2D to 3D, and formulate the 3D scattering problem using the FEM procedure. The architecture of the FEM software (FEMS^{3D}) which is developed to implement this problem is presented in detail in Appendix B.

3.1. Derivation of the Vector Wave Equation

In this section, we briefly formulate the vector wave equation governing the electromagnetic scattering problem in 3D. The general model, which is previously given in Figure 2.1, is also applicable here, but with a generalization to 3D. In this case, we consider the electromagnetic scattering problem involving a single PEC or dielectric scatterer of arbitrary shape and material properties. The scatterer is illuminated by a plane wave with arbitrary polarization and direction of incidence.

Similar to the 2D analysis, we start with the well-known source-free Maxwell's equations for the field in the presence of the scatterer as follows

$$\nabla \times \vec{E} = -j\omega\mu_0\mu_r\vec{H} \quad (3.1.a)$$

$$\nabla \times \vec{H} = j\omega\varepsilon_0\varepsilon_r\vec{E} \quad (3.1.b)$$

where ε_0 and μ_0 are the permittivity and permeability of free-space, respectively. Moreover, ε_r and μ_r are the relative permittivity and permeability of the scatterer, respectively, and they are set equal to unity inside the free-space region. From the combinations of the expressions in (3.1), the vector wave equation for the electric field can be written as

$$\nabla \times \frac{1}{\mu_r} \nabla \times \vec{E} - k^2 \varepsilon_r \vec{E} = 0 \quad (3.2)$$

where $k = \omega \sqrt{\mu_0 \varepsilon_0}$ is the free-space wave number.

The field in the presence of the scatterer can be expressed as

$$\vec{E} = \vec{E}^{inc} + \vec{E}^s \quad (3.3)$$

where \vec{E}^{inc} is the incident field produced in the absence of the scatterer, and \vec{E}^s is the scattered field produced by the equivalent induced currents on the scatterer or on a surface enclosing the scatterer.

For PEC scatterers, we derive the BVP governed by the vector wave equation for the scattered field as follows

$$\nabla \times \nabla \times \vec{E}^s - k^2 \vec{E}^s = 0 \quad \text{in } \Omega_{FS} \quad (3.4.a)$$

$$\text{with BC: } \hat{n} \times \vec{E}^s = -\hat{n} \times \vec{E}^{inc} \quad \text{on } \partial\Omega_S \quad (3.4.b)$$

Similarly, (3.4.b) is due the fact that the total field vanishes on the boundary of the PEC scatterer ($\partial\Omega_S$).

For dielectric scatterers, we derive the vector wave equation for the scattered field as follows

$$\nabla \times \frac{1}{\mu_r} \nabla \times \vec{E}^s - k^2 \varepsilon_r \vec{E}^s = k^2 (\varepsilon_r - 1) \vec{E}^{inc} \quad (3.5)$$

Similarly, (3.5) reduces to (3.4.a) in free-space region.

We consider the incident field to be a uniform plane wave, which can be written as

$$\begin{aligned}\vec{E}^{inc}(\vec{r}) &= \hat{a}_p \exp[-jk\hat{a}_k \cdot \vec{r}] \\ &= \hat{a}_p \exp\left[jk\left(x \sin \theta^{inc} \cos \varphi^{inc} + y \sin \theta^{inc} \sin \varphi^{inc} + z \cos \theta^{inc}\right)\right]\end{aligned}\quad (3.6)$$

where \hat{a}_k is the unit vector along the direction of incidence, \hat{a}_p is the unit vector denoting the polarization, and $(\theta^{inc}, \varphi^{inc})$ represent the direction of incidence.

As mentioned in Chapter 2, the propagation of the fields away from the scatterer is ensured by imposing the Sommerfeld radiation condition at the far-zone of the scatterer, which is expressed in a 3D problem by

$$\lim_{r \rightarrow \infty} r(\nabla \times \vec{E} + jk\hat{a}_r \times \vec{E}) = 0 \quad (3.7)$$

where r is the radial distance from the scatterer. In order to simulate the Sommerfeld radiation condition, we employ the PML approach to truncate the physical domain, as formulated in Chapter 4. In the PML region Ω_{PML} , we solve (3.4.a) in complex space by replacing the real coordinates in Ω_{PML} by their complex counterparts, and preserving the original form of the vector wave equation.

3.2. Weak Variational Form of the Vector Wave Equation

We calculate the weak variational form of the vector wave equation derived in Section 3.1, using the weighted residual method.

For the scattering problem (PEC case), in order to derive the weak variational form of (3.4.a), we first take the inner product of this equation by a vector weight function \vec{W} , as follows

$$\int_{\Omega_{\text{FS}}} (\nabla \times \nabla \times \vec{E}^s) \cdot \vec{W} d\Omega - k^2 \int_{\Omega_{\text{FS}}} \vec{E}^s \cdot \vec{W} d\Omega = 0 \quad (3.8)$$

Using the following identity (for any vector functions \vec{A} and \vec{B})

$$(\nabla \times \vec{A}) \cdot \vec{B} = \vec{A} \cdot (\nabla \times \vec{B}) + \nabla \cdot (\vec{A} \times \vec{B}) \quad (3.9)$$

and the following divergence theorem

$$\int_{\Omega} \nabla \cdot (\vec{A} \times \vec{B}) d\Omega = \int_{\partial\Omega} (\vec{A} \times \vec{B}) \cdot d\vec{s} \quad (3.10)$$

the weak variational form of (3.4.a) can be expressed as

$$\int_{\Omega_{FS}} (\nabla \times \vec{E}^s) \cdot (\nabla \times \vec{W}) d\Omega - k^2 \int_{\Omega_{FS}} \vec{E}^s \cdot \vec{W} d\Omega = 0 \quad (3.11)$$

Similarly, for the scattering problem (dielectric case), the weak variational form of (3.5) can be computed as

$$\int_{\Omega_{FS} \cup \Omega_S} \left(\frac{1}{\mu_r} \nabla \times \vec{E}^s \right) \cdot (\nabla \times \vec{W}) d\Omega - k^2 \int_{\Omega_{FS} \cup \Omega_S} \varepsilon_r \vec{E}^s \cdot \vec{W} d\Omega = \int_{\Omega_S} k^2 (\varepsilon_r - 1) \vec{E}^{inc} \cdot \vec{W} d\Omega \quad (3.12)$$

3.3. FEM Formulation using Tetrahedral Edge Elements

In FEM, we solve the weak form of the vector wave equation derived in Section 3.2 by discretizing the computational domain using tetrahedral edge elements.

Within each element, the unknown function (i.e., \vec{E}^s) is approximated as

$$\vec{E}^{s,e}(\vec{r}) = \sum_{i=1}^6 \vec{N}_i(\vec{r}) E_i^e \quad (3.13)$$

where E_i^e is the unknown tangential field along the i^{th} edge, and $\vec{N}_i(\vec{r})$ is the vector shape function for the i^{th} edge.

Then, we substitute (3.13) into (3.11) and (3.12), and we use the Galerkin approach where the weight functions are chosen to be equal to the shape

functions (i.e., $\vec{W} = \vec{N}_i(\vec{r})$). After this step, for the scattering problem (PEC case), the weak variational form in (3.11) becomes

$$\begin{aligned} & \int_{\Omega_{FS}^e} \left[\nabla \times \left(\sum_{i=1}^6 \vec{N}_i(\vec{r}) \mathbf{E}_i^e \right) \right] \cdot (\nabla \times \vec{N}_j(\vec{r})) d\Omega - \\ & k^2 \int_{\Omega_{FS}^e} \left(\sum_{i=1}^6 \vec{N}_i(\vec{r}) \mathbf{E}_i^e \right) \cdot \vec{N}_j(\vec{r}) d\Omega = 0 \quad (j=1, 2, \dots, 6) \end{aligned} \quad (3.14)$$

Similarly, for the scattering problem (dielectric case), the weak variational form in (3.12) becomes

$$\begin{aligned} & \int_{\Omega_{FS}^e \cup \Omega_S^e} \left[\frac{1}{\mu_r} \nabla \times \left(\sum_{i=1}^6 \vec{N}_i(\vec{r}) \mathbf{E}_i^e \right) \right] \cdot (\nabla \times \vec{N}_j(\vec{r})) d\Omega - \\ & k^2 \int_{\Omega_{FS}^e \cup \Omega_S^e} \varepsilon_r \left(\sum_{i=1}^6 \vec{N}_i(\vec{r}) \mathbf{E}_i^e \right) \cdot \vec{N}_j(\vec{r}) d\Omega = \int_{\Omega_S^e} k^2 (\varepsilon_r - 1) \vec{E}^{inc} \cdot \vec{N}_j(\vec{r}) d\Omega \end{aligned} \quad (3.15)$$

(j=1, 2, ..., 6)

From (3.14), we construct a 6×6 local matrix whose ij-th entry is given by

$$a_{ij}^e = \int_{\Omega_{FS}^e} (\nabla \times \vec{N}_i(\vec{r})) \cdot (\nabla \times \vec{N}_j(\vec{r})) d\Omega - k^2 \int_{\Omega_{FS}^e} \vec{N}_i(\vec{r}) \cdot \vec{N}_j(\vec{r}) d\Omega \quad (3.16)$$

and we construct a 6×1 local right-hand-side (RHS) vector whose entries are all zero (i.e., $b_j^e = 0$, $j=1, 2, \dots, 6$).

Similarly, from (3.15), we construct a 6×6 local matrix whose ij-th entry is given by

$$a_{ij}^e = \int_{\Omega_{FS}^e \cup \Omega_S^e} \left[\frac{1}{\mu_r} \nabla \times \vec{N}_i(\vec{r}) \right] \cdot (\nabla \times \vec{N}_j(\vec{r})) d\Omega - k^2 \int_{\Omega_{FS}^e \cup \Omega_S^e} \varepsilon_r \vec{N}_i(\vec{r}) \cdot \vec{N}_j(\vec{r}) d\Omega \quad (3.17)$$

and we construct a 6×1 local RHS vector whose j-th entry is given by

$$b_j^e = \int_{\Omega_S^e} k^2 (\varepsilon_r - 1) \vec{E}^{inc} \cdot \vec{N}_j(\vec{r}) d\Omega \quad (3.18)$$

The integrations in (3.16), (3.17) and (3.18) are not performed directly in terms of the global coordinates, but the element is mapped to a master element in local coordinates using the “isoparametric mapping” (see Figure 3.1). In this mapping, both the global coordinates and the unknown function are expressed in terms of the same shape functions.

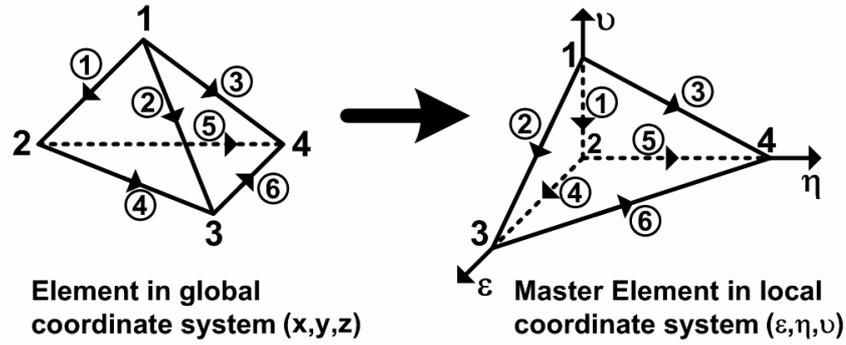


Figure 3.1. Isoparametric mapping in 3D FEM formulation.

In each local master element, the vector shape functions are expressed as ([15])

$$\vec{N}_i(\varepsilon, \eta, \nu) = (N_{i_1} \nabla N_{i_2} - N_{i_2} \nabla N_{i_1}) l_i \quad (3.19)$$

where l_i is the length of the i^{th} edge, i_1 and i_2 are the nodes associated with the i^{th} edge, N_{i_1} and N_{i_2} are the scalar shape functions for the nodes i_1 and i_2 , respectively. The edge numbers and the associated nodes i_1 and i_2 are defined in Figure 3.1 by the direction of arrows over each edge. That is, for the i^{th} edge, i_1 is the node at the beginning of the arrow, and i_2 is the node at the end of the arrow. For instance, for the 1st edge, i_1 is 1 and i_2 is 2.

In each local master element, the scalar shape functions are defined as ([15])

$$N_1 = 1 - \varepsilon - \eta - \nu \quad (3.20.a)$$

$$N_2 = \varepsilon \quad (3.20.b)$$

$$N_3 = \eta \quad (3.20.c)$$

$$N_4 = \nu \quad (3.20.d)$$

Using the isoparametric mapping, the coordinate variable variations are expressed in terms of the scalar shape functions and the global node coordinates (in Cartesian coordinates) as follows:

$$x = \sum_{i=1}^4 x_i N_i(\varepsilon, \eta, \nu) \quad (3.21.a)$$

$$y = \sum_{i=1}^4 y_i N_i(\varepsilon, \eta, \nu) \quad (3.21.b)$$

$$z = \sum_{i=1}^4 z_i N_i(\varepsilon, \eta, \nu) \quad (3.21.c)$$

The unknown field is also expressed as follows

$$\vec{E}^{s,e}(\varepsilon, \eta, \nu) = \sum_{i=1}^6 \vec{N}_i(\varepsilon, \eta, \nu) E_i^{s,e} \quad (3.22)$$

Using the expressions in (3.21), the Jacobian matrix is calculated as

$$J_{\text{FEM}} = \begin{bmatrix} \partial x / \partial \varepsilon & \partial y / \partial \varepsilon & \partial z / \partial \varepsilon \\ \partial x / \partial \eta & \partial y / \partial \eta & \partial z / \partial \eta \\ \partial x / \partial \nu & \partial y / \partial \nu & \partial z / \partial \nu \end{bmatrix} = \begin{bmatrix} x_2 - x_1 & y_2 - y_1 & z_2 - z_1 \\ x_3 - x_1 & y_3 - y_1 & z_3 - z_1 \\ x_4 - x_1 & y_4 - y_1 & z_4 - z_1 \end{bmatrix} \quad (3.23)$$

Then, using (3.19)-(3.23), the a_{ij}^e expression in (3.16) becomes

$$a_{ij}^e = \int_{\Omega_{FS}^{em}} (\nabla \times \vec{N}_i(\varepsilon, \eta, \nu)) \cdot (\nabla \times \vec{N}_j(\varepsilon, \eta, \nu)) \det(J_{\text{FEM}}) d\varepsilon d\eta d\nu - k^2 \int_{\Omega_{FS}^{em}} \vec{N}_i(\varepsilon, \eta, \nu) \cdot \vec{N}_j(\varepsilon, \eta, \nu) \det(J_{\text{FEM}}) d\varepsilon d\eta d\nu \quad (3.24)$$

Similarly, a_{ij}^e expression in (3.17) and b_j^e expression in (3.18) become, respectively, as

$$a_{ij}^e = \int_{\Omega_{FS}^{em} \cup \Omega_S^{em}} \left[\frac{1}{\mu_r} \nabla \times \vec{N}_i(\varepsilon, \eta, \nu) \right] \cdot (\nabla \times \vec{N}_j(\varepsilon, \eta, \nu)) \det(J_{\text{FEM}}) d\varepsilon d\eta d\nu - k^2 \int_{\Omega_{FS}^{em} \cup \Omega_S^{em}} \varepsilon_r \vec{N}_i(\varepsilon, \eta, \nu) \cdot \vec{N}_j(\varepsilon, \eta, \nu) \det(J_{\text{FEM}}) d\varepsilon d\eta d\nu \quad (3.25)$$

and

$$b_j^e = \int_{\Omega_3^{em}} k^2 (\varepsilon_r^e - 1) \bar{E}^{inc} \cdot \bar{N}_j(\varepsilon, \eta, \nu) \det(\mathbf{J}_{FEM}) d\varepsilon d\eta d\nu \quad (3.26)$$

In order to evaluate the expressions in (3.24)-(3.26), we need to calculate both $\bar{N}_i(\varepsilon, \eta, \nu)$ and $\nabla \times \bar{N}_i(\varepsilon, \eta, \nu)$ expressions, which simply depend on $\nabla \varepsilon$, $\nabla \eta$ and $\nabla \nu$ terms. More explicitly, $\bar{N}_i(\varepsilon, \eta, \nu)$ is expressed as

$$\bar{N}_1(\varepsilon, \eta, \nu) = l_1 [\nabla \varepsilon (1 - \eta - \nu) + \varepsilon \nabla \eta + \varepsilon \nabla \nu] \quad (3.27.a)$$

$$\bar{N}_2(\varepsilon, \eta, \nu) = l_2 [\nabla \eta (1 - \varepsilon - \nu) + \eta \nabla \varepsilon + \eta \nabla \nu] \quad (3.27.b)$$

$$\bar{N}_3(\varepsilon, \eta, \nu) = l_3 [\nabla \nu (1 - \eta - \varepsilon) + \nu \nabla \eta + \nu \nabla \varepsilon] \quad (3.27.c)$$

$$\bar{N}_4(\varepsilon, \eta, \nu) = l_4 [\varepsilon \nabla \eta - \eta \nabla \varepsilon] \quad (3.27.d)$$

$$\bar{N}_5(\varepsilon, \eta, \nu) = l_5 [\nu \nabla \varepsilon - \varepsilon \nabla \nu] \quad (3.27.e)$$

$$\bar{N}_6(\varepsilon, \eta, \nu) = l_6 [\eta \nabla \nu - \nu \nabla \eta] \quad (3.27.f)$$

and $\nabla \times \bar{N}_i(\varepsilon, \eta, \nu)$ is expressed as

$$\nabla \times \bar{N}_1(\varepsilon, \eta, \nu) = 2l_1 \nabla N_1 \times \nabla N_2 = 2l_1 [\nabla \varepsilon \times \nabla \eta + \nabla \varepsilon \times \nabla \nu] \quad (3.28.a)$$

$$\nabla \times \bar{N}_2(\varepsilon, \eta, \nu) = 2l_2 \nabla N_1 \times \nabla N_3 = 2l_2 [\nabla \eta \times \nabla \varepsilon + \nabla \eta \times \nabla \nu] \quad (3.28.b)$$

$$\nabla \times \bar{N}_3(\varepsilon, \eta, \nu) = 2l_3 \nabla N_1 \times \nabla N_4 = 2l_3 [\nabla \nu \times \nabla \eta + \nabla \nu \times \nabla \varepsilon] \quad (3.28.c)$$

$$\nabla \times \bar{N}_4(\varepsilon, \eta, \nu) = 2l_4 \nabla N_2 \times \nabla N_3 = 2l_4 [\nabla \varepsilon \times \nabla \eta] \quad (3.28.d)$$

$$\nabla \times \bar{N}_5(\varepsilon, \eta, \nu) = 2l_5 \nabla N_2 \times \nabla N_4 = 2l_5 [\nabla \varepsilon \times \nabla \nu] \quad (3.28.e)$$

$$\nabla \times \bar{N}_6(\varepsilon, \eta, \nu) = 2l_6 \nabla N_3 \times \nabla N_4 = 2l_6 [\nabla \eta \times \nabla \nu] \quad (3.28.f)$$

The components of the terms ($\nabla \varepsilon$, $\nabla \eta$ and $\nabla \nu$) are entirely determined by the inverse of the Jacobian matrix in (3.23), whose entries are given in terms of the global node coordinates, as follows

$$\begin{aligned}
\frac{\partial \varepsilon}{\partial x} &= [\mathbf{J}_{\text{FEM}}^{-1}]_{1,1}, & \frac{\partial \varepsilon}{\partial y} &= [\mathbf{J}_{\text{FEM}}^{-1}]_{2,1}, & \frac{\partial \varepsilon}{\partial z} &= [\mathbf{J}_{\text{FEM}}^{-1}]_{3,1} \\
\frac{\partial \eta}{\partial x} &= [\mathbf{J}_{\text{FEM}}^{-1}]_{1,2}, & \frac{\partial \eta}{\partial y} &= [\mathbf{J}_{\text{FEM}}^{-1}]_{2,2}, & \frac{\partial \eta}{\partial z} &= [\mathbf{J}_{\text{FEM}}^{-1}]_{3,2} \\
\frac{\partial \nu}{\partial x} &= [\mathbf{J}_{\text{FEM}}^{-1}]_{1,3}, & \frac{\partial \nu}{\partial y} &= [\mathbf{J}_{\text{FEM}}^{-1}]_{2,3}, & \frac{\partial \nu}{\partial z} &= [\mathbf{J}_{\text{FEM}}^{-1}]_{3,3}
\end{aligned} \tag{3.29}$$

where $[\mathbf{J}_{\text{FEM}}^{-1}]_{ij}$ is the ij -th entry of the inverse Jacobian matrix.

The integrals in the elemental equations in (3.24)-(3.26) are evaluated using the Gaussian Quadrature. Then, these elemental equations are assembled over all elements to form the global system of equations, by using the edge connectivity knowledge of the elements. After the BC in (3.4.b) is imposed in the case of PEC scatterer, the system of equations is solved for the unknown function using some iterative solvers.

3.4. RCS Calculation

In the 3D scattering problem, RCS is calculated as follows

$$\sigma_{3D} = \lim_{r \rightarrow \infty} 4\pi r^2 \frac{|\vec{E}_{far}^s|^2}{|\vec{E}^{inc}|^2} \tag{3.30}$$

where \vec{E}_{far}^s and \vec{E}^{inc} are the far-zone scattered field observed in a given direction and the incident field, respectively.

Similar to the 2D case, the far-zone scattered field is obtained by the Huygens' Principle, and is calculated by the same expressions in (2.36) and (2.37) in Chapter 2.

In the 3D case, the RCS is usually normalized to λ^2 or m^2 . The unit for σ_{3D}/λ^2 is dBsw, and for σ_{3D}/m^2 is dBsm.

CHAPTER 4

PERFECTLY MATCHED LAYER (PML) METHODS: “LOCALLY-CONFORMAL PML” & “MULTI-CENTER PML”

In this chapter, we present two PML approaches, which we name as the “*Locally-conformal PML*” method and the “*Multi-center PML*” method [105]-[109]. These approaches are easily implementable, non-Maxwellian PML techniques in terms of the locally-defined complex coordinate stretching approach without employing artificial anisotropic materials, for mesh truncation in FEM applications. In these methods, the PML action is achieved by the analytic continuation of the frequency-domain Maxwell’s equations to complex space, due to the complex coordinate transformation. In order to implement these methods in complex space, we keep the same constitutive parameters everywhere and replace the real coordinates with their complex counterparts calculated via the complex coordinate transformation. In these approaches, Maxwell’s equations are modified accordingly inside the complex space, and the weak variational form of the wave equation is derived in terms of the complex coordinates. Then, the weak variational form of the wave equation is discretized using the *complex elements* (i.e., elements with complex nodal coordinates). In other words, the elements in real coordinate system are mapped to the complex elements through the complex coordinate stretching. Although the resultant fields inside the PML region are non-Maxwellian, this does not cause any drawback for the PML implementation in the FEM because we demonstrate that the equations related to the FEM formulation in PML region depend directly on

the node coordinates, thus, the replacement of the node coordinates with the complex coordinates is sufficient to achieve the design of these PML methods.

These new techniques have some vital practical advantages in terms of the easy design of PMLs having challenging geometries. Although the anisotropic PML concept has been used in the theoretical design of conformal PMLs using a local coordinate system ([50] and [63]), this approach has some difficulties in terms of the computational effort, especially for the implementation of the PML at the intersection regions or in the case of abrupt changes in curvature. The anisotropic PML implementation may be easy for some smooth geometries (such as a spherical or cylindrical shell), but it may be very difficult to design an anisotropic PML for geometries in Figure 4.1, due to the presence of some problematic regions shown in the figure. The PML geometries in Figure 4.1(a) and Figure 4.1(c) may be interpreted as the partial cross sections of a conformal PML region designed to enclose a conesphere object in a scattering problem (see Figure 4.2(a)). Similarly, the geometry in Figure 4.1(b) is a cross-sectional sketch of a PML domain conformal to an ogive object (see Figure 4.2(b)). We demonstrate that such challenging PML implementations can be easily obtained via these new PML approaches. The main advantage of the approaches is the flexibility to design a PML domain which encloses an arbitrarily-shaped convex spatial domain, possibly with abrupt changes and even discontinuities in the curvature of the free space-PML interface. Such conformal PML domains are very crucial especially in radiation and/or scattering problems, where the minimization of the white space is essential in order to save on the computational supply (such as memory and processing power). The implementations of these methods are very easy since the parametric representation of the complex coordinate transformation is obtained by using only a single parameter and a single unit vector, which are easily derived from the node coordinates in an existing FEM mesh using some computational search techniques, and the usage of a local coordinate system is avoided.

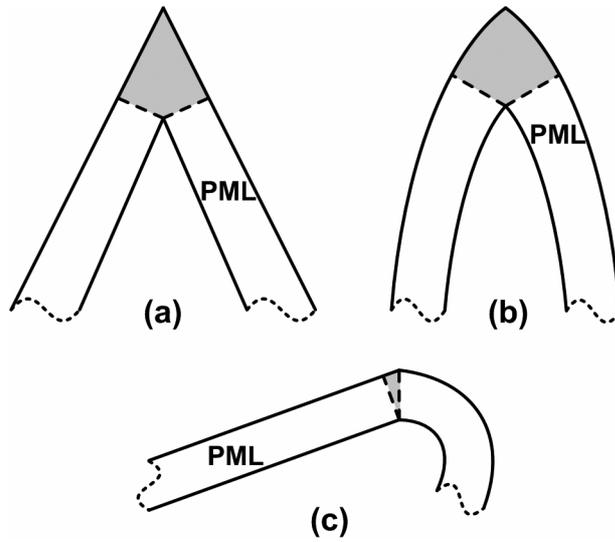


Figure 4.1. Some challenging PML geometries, whose problematic regions are shaded by the gray-colour.

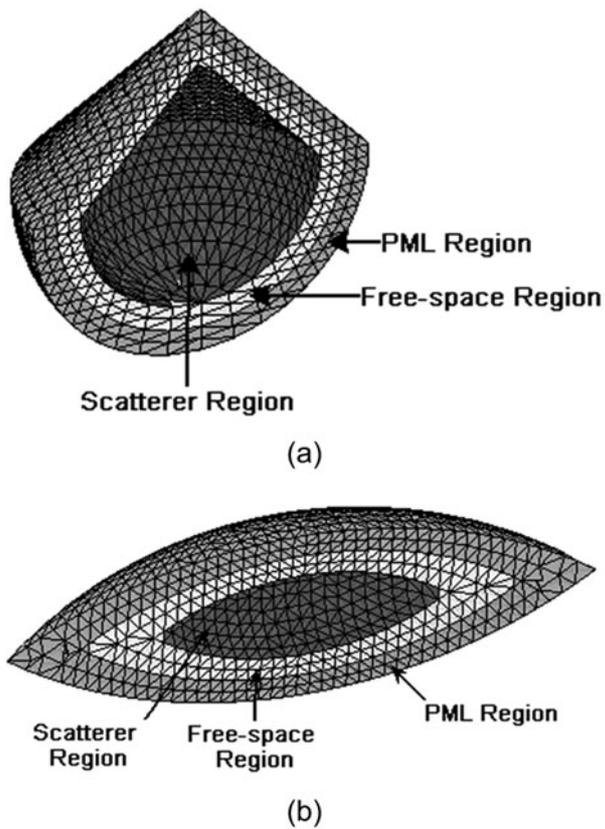


Figure 4.2. Mesh cross-sections demonstrating conformal PMLs in scattering problems: (a) Conesphere, (b) Ogive.

4.1. Locally-Conformal PML Formulation

The locally-conformal PML method is based on the complex coordinate stretching introduced by Chew & Weedon [61]. The initial phase of the approach is the construction of the spatial region occupied by the PML such that the white space within the computational domain is minimized. If the problem involves a single convex object (i.e., the spatial domain occupied by the object is a convex subset of \mathbb{R}^3 in 3D or \mathbb{R}^2 in 2D), we can place the PML region at a suitable distance, which is typically a fraction of the wavelength λ , from the surface of the object. For an object which is not convex or for a group of objects, we can choose a PML region which is conformal to the convex hull (i.e., the smallest convex set that encloses the object(s)). We illustrate the method in Figure 4.3, which represents a partial cross-section of an arbitrary PML region (Ω_{PML}). We do not show the object(s) in Ω , which actually contains the free-space region and the region occupied by object(s), for the purpose of simplification. The PML region is enclosed within the boundaries $\partial\Omega$ and $\partial\Omega_{\text{PML}}$. The whole computational domain (Ω_{C}) is the union of the regions Ω_{PML} and Ω . We choose four typical points in Ω_{PML} (P_a, P_b, P_c and P_d), for the sake of illustration.

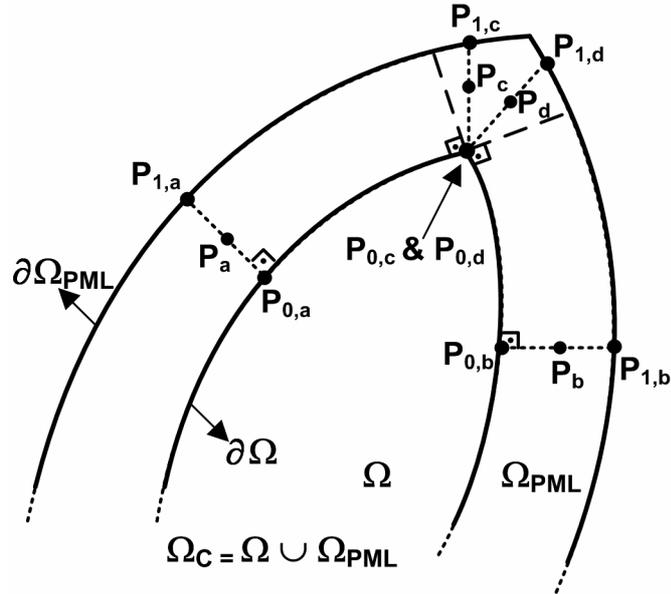


Figure 4.3. Locally-conformal PML implementation.

In order to obtain the PML action, the region Ω_{PML} , which is originally a subset of \mathbb{R}^3 in 3D (or \mathbb{R}^2 in 2D), is extended to $\Gamma \subset \mathbb{C}^3$ in 3D (or $\Gamma \subset \mathbb{C}^2$ in 2D), which is a manifold in \mathbb{C}^3 in 3D (or \mathbb{C}^2 in 2D). During this procedure, locally defined coordinate systems are totally avoided, and each point P in Ω_{PML} is mapped to \tilde{P} in Γ by using the parameter

$$\xi = \|\vec{r} - \vec{r}_0\| \quad (4.1)$$

and the unit vector

$$\hat{n}(P, P_0) = \frac{\vec{r} - \vec{r}_0}{\|\vec{r} - \vec{r}_0\|} \quad (4.2)$$

where \vec{r} and \vec{r}_0 are the position vectors of the points P and P_0 , respectively. The point P_0 (i.e., $P_{0,a}$, $P_{0,b}$, $P_{0,c}$ or $P_{0,d}$ in Figure 4.3) is located on $\partial\Omega$, and $\vec{r}_0 \in \partial\Omega$ is the solution of

$$\min_{\vec{r}_0 \in \partial\Omega} \|\vec{r} - \vec{r}_0\| \quad (4.3)$$

Such a \vec{r}_0 exists and is uniquely defined because $\partial\Omega$ is the boundary of the convex set Ω . The unit vector in (4.2) is actually the unit vector which is normal to $\partial\Omega$, except for the PML points inside the subdomain (i.e., problematic region) which is shown between the dashed lines in Figure 4.3. That is, for all points inside the problematic region (such as P_c and P_d), the solution of the minimization problem in (4.3) yields the same value for \vec{r}_0 , as shown in Figure 4.3. For other points (e.g., P_a and P_b), \vec{r}_0 can be evaluated via (4.3) by using some simple search techniques in the existing FEM mesh coordinates.

Now, consider an outgoing wave in the neighborhood of an arbitrary point P_0 . This wave can be locally represented as a superposition of plane waves (i.e., local plane wave spectrum) and a typical representative is given below as

$$\vec{E}(\vec{r}) = \hat{a}_p \exp[-jk\hat{a}_k \cdot \vec{r}] \quad (4.4)$$

where k is the free-space wave number, \hat{a}_k is the unit vector representing the direction of incidence, and \hat{a}_p is the unit vector denoting the polarization. We note that \hat{a}_k is not arbitrary, and it must satisfy the constraint $\hat{a}_k \cdot \hat{n}(P, P_0) > 0$ to be in conformity with the assumption that the original field is outgoing. We also note that this constraint excludes waves which are tangential to $\partial\Omega$ at P_0 .

In order to design the PML efficiently, the three conditions must be satisfied:

- (i) the plane wave in (4.4) must be transmitted into Ω_{PML} without any reflection,
- (ii) the transmitted wave must decay monotonically within Ω_{PML} ,
- (iii) the magnitude of the transmitted wave must be negligible on $\partial\Omega_{\text{PML}}$.

On the basis of these conditions, we postulate that, in Ω_{PML} , (4.4) becomes

$$\hat{E}(\vec{r}) = \vec{E}(\vec{r}) \cdot g(\vec{r}) \quad (4.5)$$

where $g(\vec{r})$ is a function satisfying the following two conditions:

- (i) $g(\vec{r})|_{\text{on } \partial\Omega} = 1$ (assures the continuity of $\vec{E}(\vec{r})$ at the interface $\partial\Omega$ to get rid of numerical reflections)
- (ii) $g(\vec{r})$ is a monotonically decreasing function of ξ .

Thus, we assume that $g(\vec{r})$ is given by

$$g(\vec{r}) = \exp[-f(\xi) \hat{a}_k \cdot \hat{n}(P, P_0)] \quad (4.6)$$

where $f(\xi)$ is a function satisfying the following two conditions:

- (i) $f(0) = 0$
- (ii) $f(\xi)$ is a monotonically increasing function of ξ .

Thus, we assume that $f(\xi)$ is given by

$$f(\xi) = \frac{\alpha \xi^m}{m \|\vec{r}_1 - \vec{r}_0\|^{m-1}} \quad (4.7)$$

where α is a positive parameter, m is a positive integer (typically 2 or 3) related to the decay rate of the field strength inside Ω_{PML} , and \vec{r}_1 is the position vector of the point P_1 (i.e., $P_{1,a}$, $P_{1,b}$, $P_{1,c}$ or $P_{1,d}$ in Figure 4.3), which is actually the intersection of the line passing through \vec{r} and \vec{r}_0 (i.e., the dotted line in Figure 4.3) and $\partial\Omega_{\text{PML}}$. We also note that $\|\vec{r}_1 - \vec{r}_0\|$ is the local PML thickness for the corresponding PML point. The choice of exponents in (4.7) (i.e., m in the numerator and $m-1$ in the denominator) is for the purpose of possible simplifications in the derivative terms of the Jacobian tensor appearing in the complex coordinate transformation. It should be evident that the exponential term in (4.6) guarantees the decay of the transmitted wave inside Ω_{PML} along the direction of the unit vector $\hat{n}(\mathbf{P}, \mathbf{P}_0)$.

We define the complex coordinate transformation $T: \Omega_{\text{PML}} \rightarrow \Gamma$ as follows:

$$\tilde{\vec{r}} = T(\vec{r}) = \vec{r} + \frac{1}{jk} f(\xi) \hat{n}(\mathbf{P}, \mathbf{P}_0) \quad (4.8)$$

where $\tilde{\vec{r}} \in \mathbb{C}^3$ in 3D (or $\tilde{\vec{r}} \in \mathbb{C}^2$ in 2D) is the position vector of the point $\tilde{\mathbf{P}}$ in complex space. This transformation can be implemented easily in a FEM program by replacing the real-valued node coordinates inside Ω_{PML} by their complex-valued counterparts calculated by (4.8). For instance, in 3D Cartesian coordinates, the expression in (4.8) becomes explicitly

$$\tilde{x} = x + \frac{1}{jk} f(\xi) n_x(\mathbf{P}, \mathbf{P}_0) \quad (4.9.a)$$

$$\tilde{y} = y + \frac{1}{jk} f(\xi) n_y(\mathbf{P}, \mathbf{P}_0) \quad (4.9.b)$$

$$\tilde{z} = z + \frac{1}{jk} f(\xi) n_z(\mathbf{P}, \mathbf{P}_0) \quad (4.9.c)$$

where $n_x(\mathbf{P}, \mathbf{P}_0)$, $n_y(\mathbf{P}, \mathbf{P}_0)$ and $n_z(\mathbf{P}, \mathbf{P}_0)$ are the components of the unit vector $\hat{n}(\mathbf{P}, \mathbf{P}_0)$ in x , y and z directions, respectively.

4.2. Multi-Center PML Formulation

The multi-center PML technique is basically a generalization of the locally-conformal PML technique introduced in Section 4.1. We illustrate the approach in Figure 4.4 by considering a partial cross-section of an arbitrary PML region (Ω_{PML}) designed to enclose a convex spatial domain (Ω). The point P in this figure is an arbitrary point in Ω_{PML} . In this method, some center points inside Ω are determined, and the unit vectors are calculated which denote the directions of the vectors from these centers towards the PML point, P . For the purpose of illustration, we choose two centers (P_1 and P_2) as shown in Figure 4.4. Then, the unit vectors (\hat{a}_1 and \hat{a}_2) are calculated emanating from these centers in the direction of the PML point, P . The points P_{a1} , P_{b1} , P_{a2} and P_{b2} are determined using these unit vectors. The points P_{a1} and P_{b1} are the points on $\partial\Omega$ and $\partial\Omega_{\text{PML}}$, respectively, corresponding to the center P_1 and the unit vector \hat{a}_1 . Similarly, the points P_{a2} and P_{b2} are the points on $\partial\Omega$ and $\partial\Omega_{\text{PML}}$, respectively, corresponding to the center P_2 and the unit vector \hat{a}_2 . The unit vector \hat{a}_n , which is shown by the dotted line, is the same unit vector $\hat{n}(P, P_0)$ in (4.2). Finally, θ_1 and θ_2 are the angles between \hat{a}_n and the unit vectors \hat{a}_1 and \hat{a}_2 , respectively.

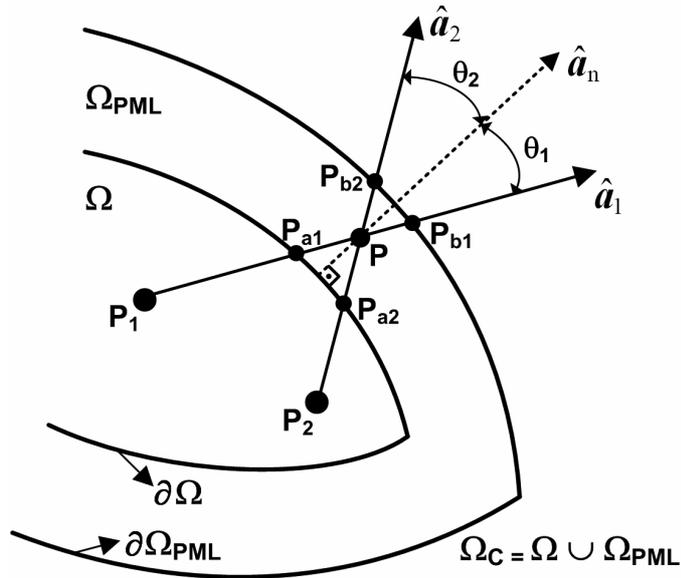


Figure 4.4. Multi-center PML implementation.

Now, similar to the locally-conformal PML approach, we assume that an outgoing plane wave is incident to the surface $\partial\Omega$, and is given by (4.4) which is rewritten here for convenience as

$$\vec{E}(\vec{r}) = \hat{a}_p \exp[-jk\hat{a}_k \cdot \vec{r}] \quad (4.10)$$

Then, using similar reasonings, we postulate that in Ω_{PML} , $\vec{E}(\vec{r})$ is given by

$$\hat{E}(\vec{r}) = \vec{E}(\vec{r}) \cdot g(\vec{r}) \quad (4.11)$$

where $g(\vec{r})$ is a function satisfying the condition $g(\vec{r})|_{\text{on } \partial\Omega} = 1$ to assure the continuity of $\vec{E}(\vec{r})$ at the interface $\partial\Omega$ to avoid numerical reflections, and is given by

$$g(\vec{r}) = \exp\left[-\alpha \left(\sum_{i=1}^N w_i \frac{(d_{ai})^m}{m(d_{bi})^{m-1}} \hat{a}_i \cdot \hat{a}_k \right)\right] \quad (4.12)$$

where α is a positive parameter, m is a positive integer (typically 2 or 3) related to the decay rate of the field strength inside the PML region (these are the same parameters as those in Section 4.1). Furthermore, N is the number of the centers, d_{ai} is the distance between P and P_{ai} for the i -th center (P_i), d_{bi} is the distance between P_{bi} and P_{ai} for P_i , and w_i is the weight (real parameter) assigned to P_i . In this formulation, the sum of all weights should add up to 1, and each weight w_i is inversely proportional to the angle θ_i , whose value is less than or equal to 45° . The value “ 45° ” is basically the threshold value in the assignment of the weights, and it is determined empirically via some numerical experiments. The details related to the choice of this threshold value are given in Section 6.1. More explicitly, for each PML point, the weights should be determined according to the following scheme:

- (i) For each PML point P , calculate the angles (θ_i 's) corresponding to the centers. If there exists only one center P_i whose angle θ_i is less than or equal to 45° , consider only this center and set the weight

corresponding to this center to 1, while the weights of all the other centers are equated to zero.

- (ii) If there exist two or more centers with angles (θ_i 's) less than or equal to 45° , consider only these centers and determine their weights such that each weight w_i is inversely proportional to the angle θ_i , while the weights of all the other centers are set to zero. This assigns the highest priority to the center whose unit vector \hat{a}_i makes the smallest angle with the unit normal vector \hat{a}_n .
- (iii) If there does not exist any center whose angle θ_i is less than or equal to 45° (undesirable case), consider all centers, and similar to step (ii), determine their weights such that each weight w_i is inversely proportional to the angle θ_i . This case is actually the indication of the improper selection of the center positions, thus, the number and the positions of the centers should be checked over in order to get more reliable results.

It should be obvious that the exponential term in (4.12) and the weight selection scheme guarantee the smooth decay of the transmitted wave inside the PML region. According to this formulation, the field inside the PML region decays along the weighted sum of the unit vectors, which are determined with respect to the centers. This weight selection scheme reveals that the multi-center PML and the locally-conformal PML approaches are related to each other in some sense, that is, for any PML point P, if the weighted sum of the unit vectors is exactly equal to the unit normal vector \hat{a}_n , then the multi-center PML reduces to the locally-conformal PML approach, because the field inside the PML region decays along \hat{a}_n in the locally-conformal PML approach. We show that the multi-center PML can also provide very accurate results as compared to the locally-conformal PML by employing a finite number of centers.

Finally, we define the complex coordinate transformation $T: \Omega_{\text{PML}} \rightarrow \Gamma$ as follows:

$$\tilde{\vec{r}} = T(\vec{r}) = \vec{r} + \frac{\alpha}{jk} \sum_{i=1}^N w_i \frac{(d_{ai})^m}{m(d_{bi})^{m-1}} \hat{a}_i \quad (4.13)$$

where $\tilde{\vec{r}} \in \mathbb{C}^3$ in 3D (or $\tilde{\vec{r}} \in \mathbb{C}^2$ in 2D) is the position vector of the point $\tilde{\mathbf{P}}$ in complex space. Similar to (4.8), this transformation can be realized by interchanging the real-valued node coordinates inside Ω_{PML} by their complex-valued counterparts calculated by (4.13). Since the local FEM matrix entries depend on the coordinate variations (see Section 4.3), the effect of the complex coordinate stretching can be easily incorporated in an existing FEM code.

The selection of the total number and positions of the centers depends directly on the geometry of the PML region Ω_{PML} . The centers should be chosen such that any PML point should have *at least one center*, whose angle θ_i is less than or equal to 45° , in order to avoid the occurrence of the undesirable case in the step (iii) in the weight selection scheme. The selection of centers is very easy for some smooth geometries (such as a spherical or cubical shell), because a single center-of-mass point of these geometries can provide successful results. The center selection scheme is generally a straightforward task depending on the geometrical structure of Ω_{PML} , irrespective of the boundary curvature. If Ω_{PML} contains elongated subregions, additional centers should be chosen inside these elongated regions in order to satisfy the 45° criterion for all PML points. However, in some geometries with sharp corners (for example, in Figure 4.5(a), the partial cross section of a conical PML region is illustrated with an internal angle less than 90°), it is not possible to find any center inside Ω which satisfies the 45° angle criterion for PML points in the close neighborhood of these corners. In this case, the PML region should be slightly modified as shown in Figure 4.5(b), and some new centers should be introduced. We should also mention that since the multi-center PML design is carried out in the

preprocessing phase (i.e., before the matrix construction phase), the increase in the number of centers creates negligible burden on the memory requirement and the processing power of the computer, compared to the memory and the processing power needed for the matrix construction and solution phases of the ordinary FEM approach.

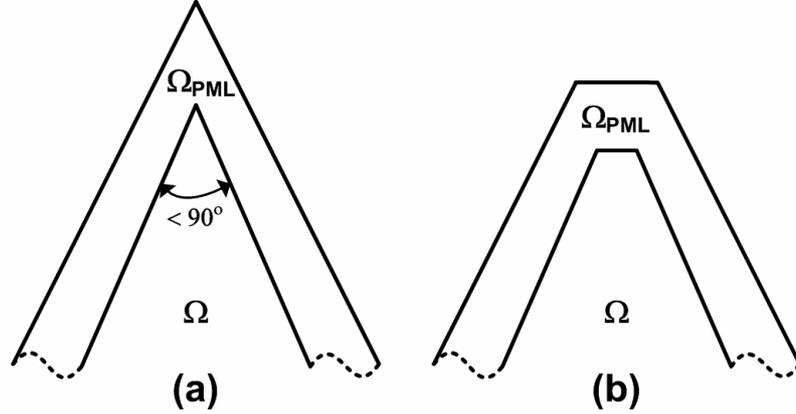


Figure 4.5. (a) The partial cross section of a conical PML region whose internal angle is less than 90° , (b) Slightly cut PML region.

4.3. Complex Space FEM Formulation

In the locally-conformal PML and multi-center PML methods, Maxwell's equations are modified through the complex coordinate transformation, given by (4.8) and (4.13), respectively. First, we define $\vec{E}^c(\tilde{\vec{r}})$ in Ω_{PML} to be the analytic continuation of $\vec{E}(\vec{r})$ to complex space, as follows

$$\vec{E}^c(\tilde{\vec{r}}) = \hat{a}_p \exp[-jk\hat{a}_k \cdot \tilde{\vec{r}}] \quad (4.14)$$

Thus, inside the complex space, the equations governing the fields $\vec{E}^c(\tilde{\vec{r}})$ and $\vec{H}^c(\tilde{\vec{r}})$ become

$$\tilde{\nabla} \times \vec{E}^c(\tilde{\vec{r}}) = -j\omega\mu_0\vec{H}^c(\tilde{\vec{r}}) \quad (4.15.a)$$

$$\tilde{\nabla} \times \vec{H}^c(\tilde{\vec{r}}) = j\omega\varepsilon_0\vec{E}^c(\tilde{\vec{r}}) \quad (4.15.b)$$

where ε_0 and μ_0 are the free-space permittivity and permeability values, respectively, $\vec{H}^c(\tilde{\vec{r}})$ is the analytic continuation of $\vec{H}(\vec{r})$ to complex space, and $\tilde{\nabla}$ is the nabla operator in complex space and is given by

$$\tilde{\nabla} = [\bar{\bar{\mathbf{J}}}^{-1}]^T \cdot \nabla \quad (4.16)$$

where $\bar{\bar{\mathbf{J}}}$ is the Jacobian tensor defined in 3D and 2D Cartesian coordinates, respectively, as

$$\bar{\bar{\mathbf{J}}} = \frac{\partial(\tilde{x}, \tilde{y}, \tilde{z})}{\partial(x, y, z)} = \begin{bmatrix} \partial\tilde{x}/\partial x & \partial\tilde{x}/\partial y & \partial\tilde{x}/\partial z \\ \partial\tilde{y}/\partial x & \partial\tilde{y}/\partial y & \partial\tilde{y}/\partial z \\ \partial\tilde{z}/\partial x & \partial\tilde{z}/\partial y & \partial\tilde{z}/\partial z \end{bmatrix} \quad (4.17.a)$$

$$\bar{\bar{\mathbf{J}}} = \frac{\partial(\tilde{x}, \tilde{y})}{\partial(x, y)} = \begin{bmatrix} \partial\tilde{x}/\partial x & \partial\tilde{x}/\partial y \\ \partial\tilde{y}/\partial x & \partial\tilde{y}/\partial y \end{bmatrix} \quad (4.17.b)$$

The fields $\vec{E}^c(\tilde{\vec{r}})$ and $\vec{H}^c(\tilde{\vec{r}})$ in the complex PML region are not Maxwellian in terms of the real coordinate variables, but this fact does not imply any difficulty for the implementation of both PML methods in a FEM application. In this section, we show that the FEM formulation can easily be carried out within the PML region which is a subset of the complex space. In Section 4.3.1, we formulate the FEM in the complex PML space using triangular isoparametric elements for the solution of the 2D scalar Helmholtz equation. Similarly, in Section 4.3.2, we formulate the FEM in the complex PML space using tetrahedral edge elements for the solution of the 3D vector wave equation.

We should note that if a Maxwellian PML is desired, both PML approaches may also be extended to obtain an anisotropic PML, that is, the resulting non-Maxwellian fields in these methods can be mapped to Maxwellian fields in an

anisotropic PML medium using some affine transformation techniques [78]. However, this anisotropic PML implementation is computationally much more complicated than the non-Maxwellian locally-conformal and multi-center PML implementations for FEM applications.

4.3.1. FEM Formulation in 2D Complex Space

We start with deriving the homogeneous Helmholtz equation in complex space using the modified Maxwell equations in (4.15) as follows (assuming TM_z case)

$$\tilde{\nabla}^2 E_z^c(\tilde{\vec{r}}) + k^2 E_z^c(\tilde{\vec{r}}) = 0 \quad (4.18)$$

The weak form of the Helmholtz equation in (4.18) can be calculated using the method of weighted residuals, and is given in complex space as

$$\int_{\Omega_{\text{PML}}} \tilde{\nabla} E_z^c(\tilde{\vec{r}}) \cdot \tilde{\nabla} w^c d\Omega - k^2 \int_{\Omega_{\text{PML}}} E_z^c(\tilde{\vec{r}}) w^c d\Omega = 0 \quad (4.19)$$

where w^c is a scalar weight function in complex space.

In FEM, we solve the weak form of the Helmholtz equation in (4.19) by discretizing the computational domain using triangular elements. In the complex coordinate transformation, the triangular elements are mapped to complex triangular elements, as illustrated in Figure 4.6.

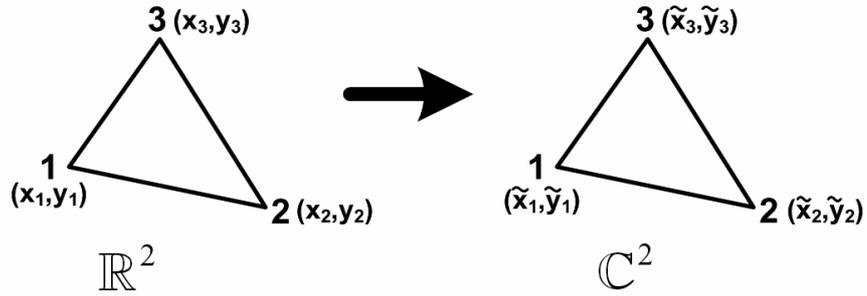


Figure 4.6. Mapping of triangular elements to complex triangular elements.

Within each element, the unknown function is approximated as

$$E_z^{c,e}(\tilde{\mathbf{r}}) = \sum_{i=1}^3 N_i(\tilde{\mathbf{r}}) E_i^{c,e} \quad (4.20)$$

where $E_i^{c,e}$ is the unknown function at the i^{th} node, and $N_i(\tilde{\mathbf{r}})$ is the scalar shape function (or approximation function) for the i^{th} node.

After substituting the expression in (4.20) into (4.19), we use the Galerkin approach where the weight functions are equal to the shape functions (i.e., $w^c = N_i(\tilde{\mathbf{r}})$). Then, the weak form of the Helmholtz equation becomes

$$\int_{\Omega_{\text{PML}}^e} \tilde{\nabla} \left(\sum_{i=1}^3 N_i(\tilde{\mathbf{r}}) E_i^{c,e} \right) \cdot \tilde{\nabla} N_j(\tilde{\mathbf{r}}) d\Omega - k^2 \int_{\Omega_{\text{PML}}^e} \left(\sum_{i=1}^3 N_i(\tilde{\mathbf{r}}) E_i^{c,e} \right) N_j(\tilde{\mathbf{r}}) d\Omega = 0 \quad (j=1,2,3) \quad (4.21)$$

From (4.21), we construct a 3×3 local matrix whose ij -th entry is given by

$$a_{ij}^e = \int_{\Omega_{\text{PML}}^e} \left[\tilde{\nabla} N_i(\tilde{\mathbf{r}}) \right] \cdot \left[\tilde{\nabla} N_j(\tilde{\mathbf{r}}) \right] d\Omega - k^2 \int_{\Omega_{\text{PML}}^e} N_i(\tilde{\mathbf{r}}) N_j(\tilde{\mathbf{r}}) d\Omega \quad (4.22)$$

The integration in (4.22) is not performed directly in terms of the global coordinates, but the element is mapped to a master element in local coordinates using the ‘‘isoparametric mapping’’ (see Figure 4.7). In this mapping, both the global coordinates and the unknown function are expressed in terms of the same shape functions.

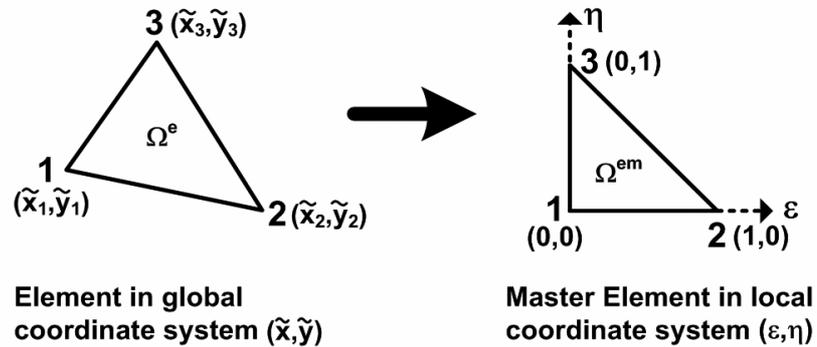


Figure 4.7. Isoparametric mapping in 2D FEM formulation.

In each local element, the scalar shape functions are defined as ([15])

$$N_1 = 1 - \varepsilon - \eta \quad (4.23.a)$$

$$N_2 = \varepsilon \quad (4.23.b)$$

$$N_3 = \eta \quad (4.23.c)$$

Using the isoparametric mapping, the coordinate variable variations are expressed in terms of the scalar shape functions and the global node coordinates (in Cartesian coordinates) as follows

$$\tilde{x} = \sum_{i=1}^3 \tilde{x}_i N_i(\varepsilon, \eta) \quad (4.24.a)$$

$$\tilde{y} = \sum_{i=1}^3 \tilde{y}_i N_i(\varepsilon, \eta) \quad (4.24.b)$$

The unknown function is also expressed as follows

$$E_z^{c,e}(\varepsilon, \eta) = \sum_{i=1}^3 N_i(\varepsilon, \eta) E_i^{c,e} \quad (4.25)$$

Using the expressions in (4.24), the Jacobian matrix is calculated as

$$J_{\text{FEM}} = \begin{bmatrix} \partial \tilde{x} / \partial \varepsilon & \partial \tilde{y} / \partial \varepsilon \\ \partial \tilde{x} / \partial \eta & \partial \tilde{y} / \partial \eta \end{bmatrix} = \begin{bmatrix} \tilde{x}_2 - \tilde{x}_1 & \tilde{y}_2 - \tilde{y}_1 \\ \tilde{x}_3 - \tilde{x}_1 & \tilde{y}_3 - \tilde{y}_1 \end{bmatrix} \quad (4.26)$$

This expression shows that the entries of the Jacobian matrix are constant, and depend on only the node coordinates.

Then, using (4.23)-(4.26), the a_{ij}^e expression in (4.22) becomes

$$a_{ij}^e = \int_{\Omega_{\text{PML}}^{\text{em}}} [\tilde{\nabla} N_i(\varepsilon, \eta)] \cdot [\tilde{\nabla} N_j(\varepsilon, \eta)] \det(J_{\text{FEM}}) d\varepsilon d\eta - k^2 \int_{\Omega_{\text{PML}}^{\text{em}}} N_i(\varepsilon, \eta) N_j(\varepsilon, \eta) \det(J_{\text{FEM}}) d\varepsilon d\eta \quad (4.27)$$

In order to evaluate (4.27), we need to calculate $\tilde{\nabla}N_i(\varepsilon, \eta)$ expression, which is simply depend on $\tilde{\nabla}\varepsilon$ and $\tilde{\nabla}\eta$ terms. The components of these terms are entirely determined by the inverse of the Jacobian matrix in (4.26), whose entries are given in terms of global node coordinates, as follows

$$\begin{aligned}\frac{\partial\varepsilon}{\partial\tilde{x}} &= [J_{\text{FEM}}^{-1}]_{1,1}, & \frac{\partial\varepsilon}{\partial\tilde{y}} &= [J_{\text{FEM}}^{-1}]_{2,1} \\ \frac{\partial\eta}{\partial\tilde{x}} &= [J_{\text{FEM}}^{-1}]_{1,2}, & \frac{\partial\eta}{\partial\tilde{y}} &= [J_{\text{FEM}}^{-1}]_{2,2}\end{aligned}\quad (4.28)$$

where $[J_{\text{FEM}}^{-1}]_{ij}$ is the ij -th entry of the inverse Jacobian matrix.

The formulation in this section shows that the a_{ij}^e expression in (4.27), which is the ij -th entry of the element matrix, is evaluated in terms of the nodal coordinates. Consequently, the FEM formulation can easily be implemented in complex space by using the complex-valued node coordinates obtained via the complex coordinate transformation. Although the evaluation of a_{ij}^e in terms of the nodal coordinates is given for triangular elements, similar derivations are obviously possible for other types of elements (e.g., quadrilateral).

4.3.2. FEM Formulation in 3D Complex Space

We start with deriving the vector wave equation in complex space using the modified Maxwell equations in (4.15) as follows

$$\tilde{\nabla} \times \tilde{\nabla} \times \vec{E}^c(\tilde{\vec{r}}) - k^2 \vec{E}^c(\tilde{\vec{r}}) = 0 \quad (4.29)$$

The weak form of this wave equation can be calculated using the method of weighted residuals, and is given in complex space as

$$\int_{\Omega_{\text{PML}}} [\tilde{\nabla} \times \vec{E}^c(\tilde{\vec{r}})] \cdot [\tilde{\nabla} \times \vec{W}^c] d\Omega - k^2 \int_{\Omega_{\text{PML}}} \vec{E}^c(\tilde{\vec{r}}) \cdot \vec{W}^c d\Omega = 0 \quad (4.30)$$

where \vec{W}^c is a vector weight function in complex space.

In FEM, we solve the weak form of the wave equation in (4.30) by discretizing the computational domain using tetrahedral edge elements. In the complex coordinate transformation, the tetrahedral elements are mapped to complex tetrahedral elements, as illustrated in Figure 4.8.

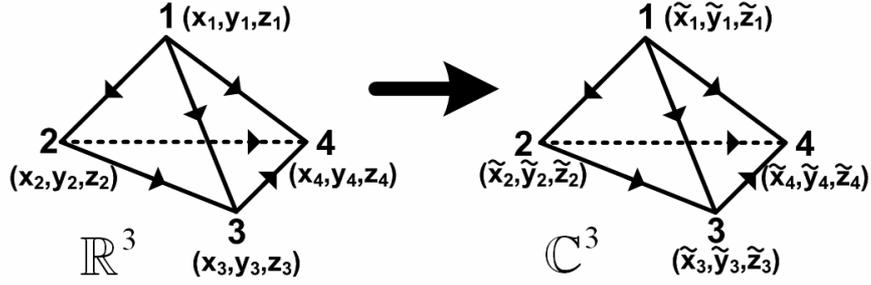


Figure 4.8. Mapping of tetrahedral elements to complex tetrahedral elements.

Within each element, the unknown field is approximated as

$$\vec{E}^{c,e}(\vec{r}) = \sum_{i=1}^6 \vec{N}_i(\vec{r}) E_i^{c,e} \quad (4.31)$$

where $E_i^{c,e}$ is the unknown tangential field along the i^{th} edge, and $\vec{N}_i(\vec{r})$ is the vector shape function for the i^{th} edge.

When the expression in (4.31) is substituted into (4.30), and we use the Galerkin approach where the weight functions are chosen to be equal to the shape functions (i.e., $\vec{W}^c = \vec{N}_j(\vec{r})$), the weak form of the wave equation becomes

$$\int_{\Omega_{\text{PML}}^c} \left[\vec{\nabla} \times \left(\sum_{i=1}^6 \vec{N}_i(\vec{r}) E_i^{c,e} \right) \right] \cdot \left[\vec{\nabla} \times \vec{N}_j(\vec{r}) \right] d\Omega - k^2 \int_{\Omega_{\text{PML}}^c} \left(\sum_{i=1}^6 \vec{N}_i(\vec{r}) E_i^{c,e} \right) \cdot \vec{N}_j(\vec{r}) d\Omega = 0 \quad (j=1, 2, \dots, 6) \quad (4.32)$$

From (4.32), we construct a 6×6 local matrix whose entries are given by

$$a_{ij}^e = \int_{\Omega_{\text{FML}}^e} [\tilde{\nabla} \times \tilde{N}_i(\tilde{\mathbf{r}})] \cdot [\tilde{\nabla} \times \tilde{N}_j(\tilde{\mathbf{r}})] d\Omega - k^2 \int_{\Omega_{\text{FML}}^e} \tilde{N}_i(\tilde{\mathbf{r}}) \cdot \tilde{N}_j(\tilde{\mathbf{r}}) d\Omega \quad (4.33)$$

The integration in (4.33) is not carried out directly in terms of the global coordinates, but the element is mapped to a master element in local coordinates using the “isoparametric mapping” (see Figure 4.9). In this mapping, both the global coordinates and the unknown field are expressed in terms of the same shape functions.

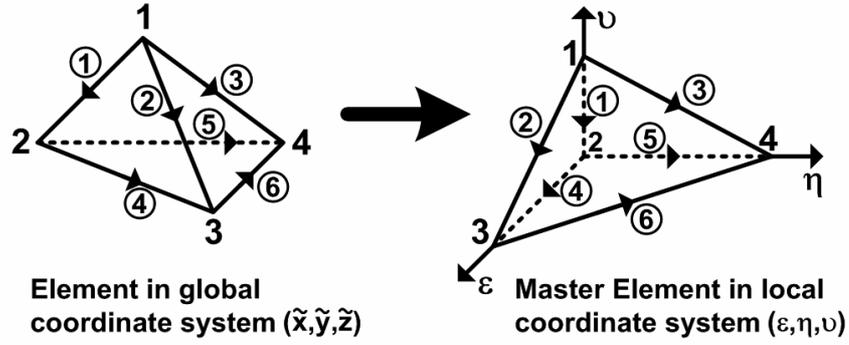


Figure 4.9. Isoparametric mapping in 3D FEM formulation.

In each local element, the vector shape functions are expressed as ([15])

$$\tilde{N}_i(\epsilon, \eta, \nu) = (N_{i_1} \tilde{\nabla} N_{i_2} - N_{i_2} \tilde{\nabla} N_{i_1}) l_i \quad (4.34)$$

where l_i is the length of the i^{th} edge, i_1 and i_2 are the nodes associated with the i^{th} edge, N_{i_1} and N_{i_2} are the scalar shape functions for the nodes i_1 and i_2 , respectively. The edge numbers and the associated nodes i_1 and i_2 are defined in Figure 4.9 by the direction of arrows over each edge. That is, for the i^{th} edge, i_1 is the node at the beginning of the arrow, and i_2 is the node at the end of the arrow. For instance, for the 1st edge, i_1 is 1 and i_2 is 2.

In each local element, the scalar shape functions are defined as ([15])

$$N_1 = 1 - \varepsilon - \eta - \nu \quad (4.35.a)$$

$$N_2 = \varepsilon \quad (4.35.b)$$

$$N_3 = \eta \quad (4.35.c)$$

$$N_4 = \nu \quad (4.35.d)$$

Using the isoparametric mapping, the coordinate variable variations are expressed in terms of the scalar shape functions and the global node coordinates (in Cartesian coordinates) as follows:

$$\tilde{x} = \sum_{i=1}^4 \tilde{x}_i N_i(\varepsilon, \eta, \nu) \quad (4.36.a)$$

$$\tilde{y} = \sum_{i=1}^4 \tilde{y}_i N_i(\varepsilon, \eta, \nu) \quad (4.36.b)$$

$$\tilde{z} = \sum_{i=1}^4 \tilde{z}_i N_i(\varepsilon, \eta, \nu) \quad (4.36.c)$$

The unknown field is also expressed as follows:

$$\vec{E}^{c,e}(\varepsilon, \eta, \nu) = \sum_{i=1}^6 \vec{N}_i(\varepsilon, \eta, \nu) E_i^{c,e} \quad (4.37)$$

Using the expressions in (4.36), the Jacobian matrix is calculated as

$$J_{\text{FEM}} = \begin{bmatrix} \partial \tilde{x} / \partial \varepsilon & \partial \tilde{y} / \partial \varepsilon & \partial \tilde{z} / \partial \varepsilon \\ \partial \tilde{x} / \partial \eta & \partial \tilde{y} / \partial \eta & \partial \tilde{z} / \partial \eta \\ \partial \tilde{x} / \partial \nu & \partial \tilde{y} / \partial \nu & \partial \tilde{z} / \partial \nu \end{bmatrix} = \begin{bmatrix} \tilde{x}_2 - \tilde{x}_1 & \tilde{y}_2 - \tilde{y}_1 & \tilde{z}_2 - \tilde{z}_1 \\ \tilde{x}_3 - \tilde{x}_1 & \tilde{y}_3 - \tilde{y}_1 & \tilde{z}_3 - \tilde{z}_1 \\ \tilde{x}_4 - \tilde{x}_1 & \tilde{y}_4 - \tilde{y}_1 & \tilde{z}_4 - \tilde{z}_1 \end{bmatrix} \quad (4.38)$$

The expression in (4.38) shows that the entries of the Jacobian matrix are constant, and depend only on the node coordinates.

Then, the a_{ij}^e expression in (4.33) becomes

$$\begin{aligned}
a_{ij}^e &= \int_{\Omega_{\text{PML}}^{\text{em}}} \left[\tilde{\nabla} \times \vec{N}_i(\varepsilon, \eta, \nu) \right] \cdot \left[\tilde{\nabla} \times \vec{N}_j(\varepsilon, \eta, \nu) \right] \det(\mathbf{J}_{\text{FEM}}) d\varepsilon d\eta d\nu - \\
& k^2 \int_{\Omega_{\text{PML}}^{\text{em}}} \vec{N}_i(\varepsilon, \eta, \nu) \cdot \vec{N}_j(\varepsilon, \eta, \nu) \det(\mathbf{J}_{\text{FEM}}) d\varepsilon d\eta d\nu
\end{aligned} \tag{4.39}$$

In order to evaluate this expression, we need to calculate both $\vec{N}_i(\varepsilon, \eta, \nu)$ and $\tilde{\nabla} \times \vec{N}_i(\varepsilon, \eta, \nu)$ expressions. It should be obvious from (4.34) and (4.35), these expressions depend only on $\tilde{\nabla} \varepsilon$, $\tilde{\nabla} \eta$ and $\tilde{\nabla} \nu$ terms. The components of these terms are entirely determined by the inverse of the Jacobian matrix in (4.38), whose entries are simply given in terms of global node coordinates, as follows:

$$\begin{aligned}
\frac{\partial \varepsilon}{\partial \tilde{x}} &= [\mathbf{J}_{\text{FEM}}^{-1}]_{1,1}, \quad \frac{\partial \varepsilon}{\partial \tilde{y}} = [\mathbf{J}_{\text{FEM}}^{-1}]_{2,1}, \quad \frac{\partial \varepsilon}{\partial \tilde{z}} = [\mathbf{J}_{\text{FEM}}^{-1}]_{3,1} \\
\frac{\partial \eta}{\partial \tilde{x}} &= [\mathbf{J}_{\text{FEM}}^{-1}]_{1,2}, \quad \frac{\partial \eta}{\partial \tilde{y}} = [\mathbf{J}_{\text{FEM}}^{-1}]_{2,2}, \quad \frac{\partial \eta}{\partial \tilde{z}} = [\mathbf{J}_{\text{FEM}}^{-1}]_{3,2} \\
\frac{\partial \nu}{\partial \tilde{x}} &= [\mathbf{J}_{\text{FEM}}^{-1}]_{1,3}, \quad \frac{\partial \nu}{\partial \tilde{y}} = [\mathbf{J}_{\text{FEM}}^{-1}]_{2,3}, \quad \frac{\partial \nu}{\partial \tilde{z}} = [\mathbf{J}_{\text{FEM}}^{-1}]_{3,3}
\end{aligned} \tag{4.40}$$

where $[\mathbf{J}_{\text{FEM}}^{-1}]_{ij}$ is the ij -th entry of the inverse Jacobian matrix.

This analysis shows that the a_{ij}^e expression in (4.39) is evaluated in terms of the nodal coordinates. Consequently, the FEM formulation can easily be implemented in complex space by using the complex-valued node coordinates obtained via the complex coordinate transformation. Although the evaluation of the a_{ij}^e expression in terms of the nodal coordinates is given for tetrahedral elements, it is obvious that similar derivations are possible for other types of elements (for example, hexahedral).

4.4. Summary of the PML Methods

The analysis of the PML methods leads us to the following conclusions:

(i) In the implementation of the methods, the differential geometric properties of the PML-free space interface have no direct effect on the application of the methods. The analytic continuation operation works equally well even in the presence of discontinuities in the curvature of the interface. This is because the stretching operation is locally-defined without using any particular coordinate system. In addition, the transformation $T: \Omega_{\text{PML}} \rightarrow \Gamma$ guarantees the decay of outgoing waves away from the interface. This decay behavior holds at all PML points, even in problematic regions close to the interface points where the surface curvature is discontinuous.

(ii) Another important observation is the continuity of $T: \Omega_{\text{PML}} \rightarrow \Gamma$, interpreted as follows:

Given any $\varepsilon > 0$, there exists $\delta > 0$ (δ depends on ε) such that:

$$\|T(\vec{r}) - T(\vec{r}^*)\| < \varepsilon, \text{ whenever } \|\vec{r} - \vec{r}^*\| < \delta \quad (4.41)$$

which implies that two closely-located points \vec{r} and \vec{r}^* in Ω_{PML} are mapped to also closely-located points $\tilde{\vec{r}}$ and $\tilde{\vec{r}}^*$ in Γ . This follows from the definition of the transformation, and this result holds for all points in Ω_{PML} , even close to the interface points with curvature jumps in problematic regions. It is obvious that this result is critical in the implementation of the complex space FEM, explained in Section 4.3.

(iii) Classical cartesian and cylindrical PML approaches in FEM literature which are usually realized in the design of PMLs over rectangular and circular regions, respectively, are actually the special cases of the locally-conformal PML method in terms of the complex coordinate transformation. For

instance, if the PML region surrounds a rectangular spatial domain as shown in Figure 4.10(a), the classical cartesian PML approach considers the following definition of the transformation

$$\begin{array}{lll}
 \text{In Region I:} & \text{In Region II:} & \text{In Region III:} \\
 \tilde{x} = x + \frac{\alpha}{jk}(x - x_0) & \tilde{y} = y + \frac{\alpha}{jk}(y - y_0) & \tilde{x} = x + \frac{\alpha}{jk}(x - x_0) \\
 \tilde{y} = y & \tilde{x} = x & \tilde{y} = y + \frac{\alpha}{jk}(y - y_0)
 \end{array} \tag{4.42}$$

Similarly, if the PML region surrounds a circular spatial domain as shown in Figure 4.10(b), the classical PML approach in cylindrical coordinates defines the following transformation

$$\tilde{r} = r + \frac{\alpha}{jk}(r - r_0) \tag{4.43}$$

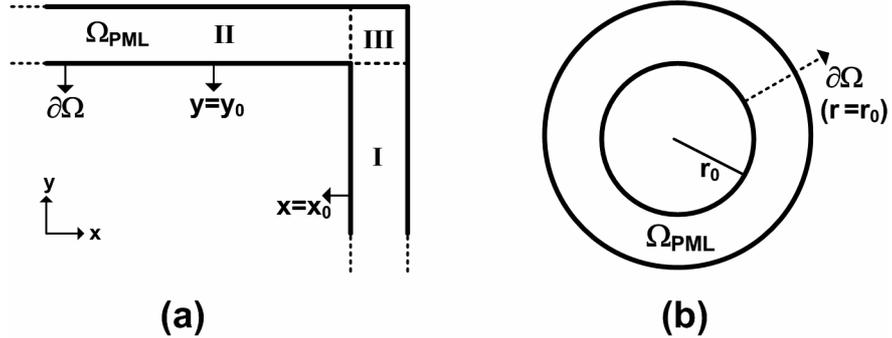


Figure 4.10. Classical PML implementations: (a) Cartesian, (b) Cylindrical.

Hence, it should be obvious that the locally-conformal PML method, whose parameter m is 1, reduces to the classical PML approach in such PML regions. However, the classical PML approaches *fail* if the PML domain encloses an arbitrarily-shaped domain, because inner and outer PML surfaces must be defined on ‘constant coordinate surfaces’ in these approaches. The locally-conformal PML method remedies this bottleneck. In addition, although the conformity of the outer PML surface to the inner

PML surface is desired, the geometries of the inner and outer PML surfaces need *not* be identical, providing a great flexibility in the design of arbitrarily-shaped PML regions.

(iv) The value of α must be specified properly to satisfy the smooth decay of the transmitted wave inside Ω_{PML} . In addition, the magnitude of the transmitted field must attain a negligible value on $\partial\Omega_{\text{PML}}$ (i.e., $g(\vec{r})|_{\text{on } \partial\Omega_{\text{PML}}} \approx 0$). With regard to the last requirement, one can claim that the value of α must be as large as possible. However, we also desire that the decay rate must not exceed a certain value. In addition, a larger value for α yields deformations in the transformed node coordinates calculated by (4.8) or (4.13) due to very large imaginary parts. In other words, the shape of the mesh elements in complex space becomes poor in quality due to the improper transformation of the coordinates. This may result in ill-conditioning in the global matrix equation, yielding inaccurate analysis results. Thus, an optimal value of α is essential for a successful PML realization. Practically, for a PML thickness between $\lambda/4$ and $\lambda/2$, the range of values $5k \leq \alpha \leq 10k$ provides reliable results for an efficient PML design.

- (v) The algorithm of the PML methods in a FEM program is summarized as:
- Construct Ω_{PML} designed over a convex region Ω . Preserve the node information (i.e., ID's and coordinates) inside Ω_{PML} , and on the boundaries $\partial\Omega$ and $\partial\Omega_{\text{PML}}$, separately.
 - Replace the node coordinates inside Ω_{PML} by their complex counterparts calculated by the complex coordinate transformation in (4.8) or (4.13).
 - For each element inside Ω_{PML} , construct the local matrix which is evaluated by (4.27) in 2D, or by (4.39) in 3D.
 - Assemble the global matrix by combining the local matrices.
 - Solve the global matrix equation for the unknown fields.

CHAPTER 5

DOMAIN DECOMPOSITION METHODS (DDMs): “FB-DDM” & “ILF-DDM”

In this chapter, we introduce two DDM approaches, which we name as “Forward-Backward DDM (FB-DDM)” and “Iterative Leap-field DDM (ILF-DDM)” [110]-[112]. These methods decompose an electrically-large electromagnetic problem into several independent subproblems, and combine the solutions of all subproblems in order to reach the global solution. The most distinguished feature of the proposed methods is the utilization of the locally-conformal PML method along the boundaries of the subdomains. The most attractive advantage of these methods is the considerable reduction in the memory requirements and computation time. These methods usually converge in a few number of iterations (or a single iteration in some cases) and they are suitable for the parallel computation of the subproblems.

5.1. Forward-Backward DDM (FB-DDM)

The Forward-Backward DDM (FB-DDM) is basically established on the classical alternating Schwarz method with overlapping subdomains [64]. However, the FB-DDM implements the Schwarz method by taking into account certain physical aspects of the original (whole) problem. That is, the Schwarz method is improved in terms of efficiency and rate of convergence by considering the geometry of the problem and the expected field behavior inside the original domain. This is achieved by the implementation of the locally-

conformal PML method along the boundaries of the subdomains. In some problems involving smooth geometries, the FB-DDM converges in just a single forward iteration (i.e., the method is non-iterative), where the problem in each subdomain is solved only once by appropriately defined subdomains and additional PML regions attached to each subdomain. In other challenging geometries, the initial forward iteration of the FB-DDM provides an initial guess ‘close’ to the exact values of the BCs of each subdomain, unlike the ‘arbitrary’ initial guess in the Schwarz method. After the initial forward iteration, the problems defined on the subdomains are solved iteratively, similar to the Schwarz method, in a forward-backward fashion until convergence is achieved. It is obvious that the ‘better’ initial guess for the BCs provides an increased convergence rate by decreasing the number of the forward-backward iterations during the field refinement process.

This section is organized as follows: In Section 5.1.1, we overview briefly the classical alternating Schwarz method. In Section 5.1.2, we draw a general picture for the implementation procedure of the FB-DDM algorithm in the problem of electromagnetic scattering from PEC objects. Then, in Section 5.1.3, we discuss some important issues related to the successful implementation of the method (such as partitioning scheme, order of subdomains, etc.) with regard to the physics of the problem. Finally, in Section 5.1.4, we present some additional remarks on how to implement the algorithm in some special problems.

5.1.1. Review of the Classical Alternating Schwarz Method

The classical alternating Schwarz method solves several overlapping subdomain problems iteratively with Dirichlet BCs in each subdomain. The algorithm starts with an arbitrary initial guess for BCs (usually zero BCs), and solves the subdomains iteratively by modifying the BCs in each iteration until convergence is reached in the solution.

We illustrate the method in Figure 5.1 by considering two subdomains [79]. We assume that we want to solve the following boundary value problem (BVP)

$$\begin{aligned} Lu &= f & \text{in } \Omega \\ u &= g & \text{on } \partial\Omega \end{aligned} \quad (5.1)$$

where L is a general operator, and the unknown u , f and g are scalar or vector functions depending on the problem. Moreover, $\partial\Omega$ denotes the boundary of Ω . We also let u_i^n denote the approximate solution in $\Omega_i \cup \partial\Omega_i$ after n iterations, and $u_1^n|_{\Gamma_2}$ be the restriction of u_1^n to Γ_2 (similarly, $u_2^n|_{\Gamma_1}$ is the restriction of u_2^n to Γ_1). The alternating Schwarz method begins with an initial guess u_2^0 for the values in Ω_2 along Γ_1 (usually $u_2^0|_{\Gamma_1} = 0$). Then, for each $n = 1, 2, 3, \dots$, we solve iteratively the following first BVP for u_1^n (first half-step)

$$\begin{aligned} Lu_1^n &= f & \text{in } \Omega_1 \\ u_1^n &= g & \text{on } \partial\Omega_1 \setminus \Gamma_1 \\ u_1^n &= u_2^{n-1}|_{\Gamma_1} & \text{on } \Gamma_1 \end{aligned} \quad (5.2)$$

and then the following second BVP for u_2^n (second half-step)

$$\begin{aligned} Lu_2^n &= f & \text{in } \Omega_2 \\ u_2^n &= g & \text{on } \partial\Omega_2 \setminus \Gamma_2 \\ u_2^n &= u_1^{n-1}|_{\Gamma_2} & \text{on } \Gamma_2 \end{aligned} \quad (5.3)$$

where $\partial\Omega_i \setminus \Gamma_i$ denotes the points on $\partial\Omega_i$ that are not on Γ_i .

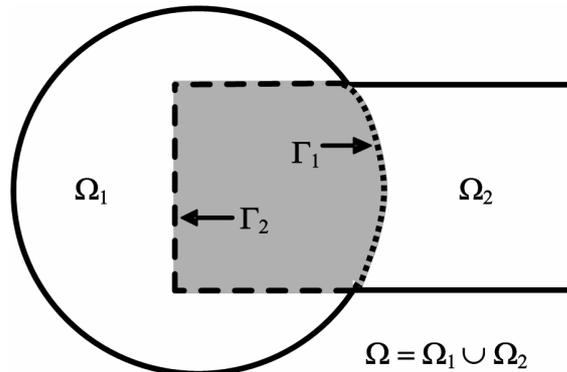


Figure 5.1. Schwarz's original figure.

5.1.2. FB-DDM Algorithm

The algorithm starts with a suitable partitioning of the original (whole) problem into some number of overlapping subdomains. For the purpose of illustrating a partitioning scheme with two subdomains, we show the original problem and the partitioned problem for a cubic PEC scatterer (or square cylinder in 2D), in Figure 5.2. Similarly, for a spherical PEC scatterer (or circular cylinder in 2D), we illustrate the original problem and the decomposed problem in Figure 5.3. In these figures, the subdomains Ω_i include both the free-space and the PML regions which already exist in the original domain and are bounded by their corresponding colors.

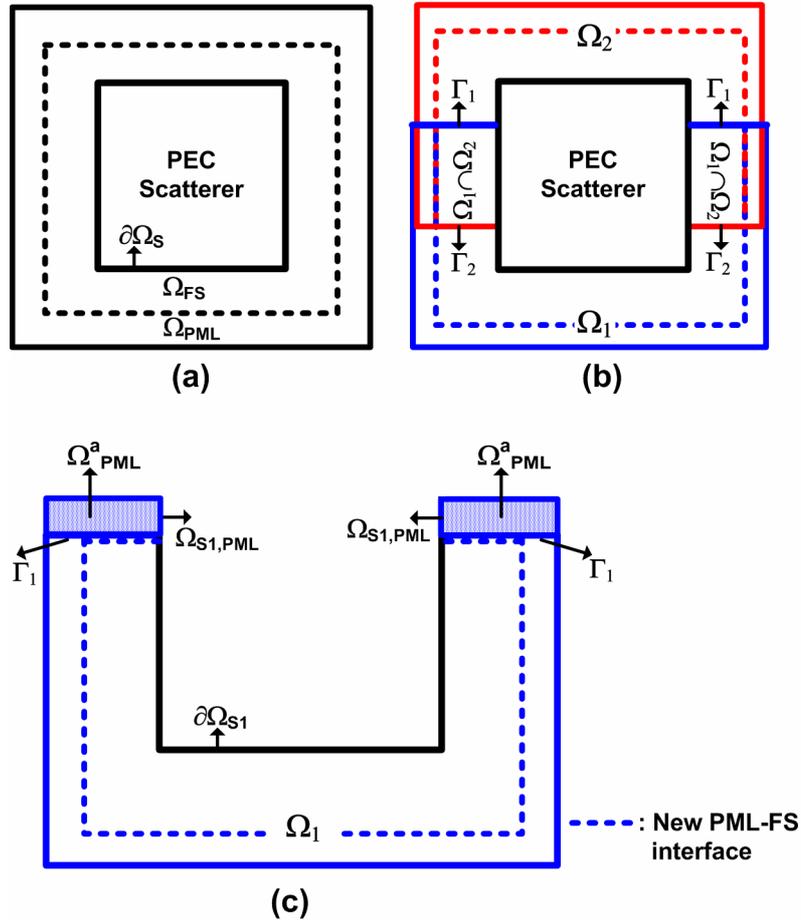


Figure 5.2. FB-DDM for cube with two subdomains: (a) Original problem, (b) Partitioned problem, (c) First subdomain with additional PML region.

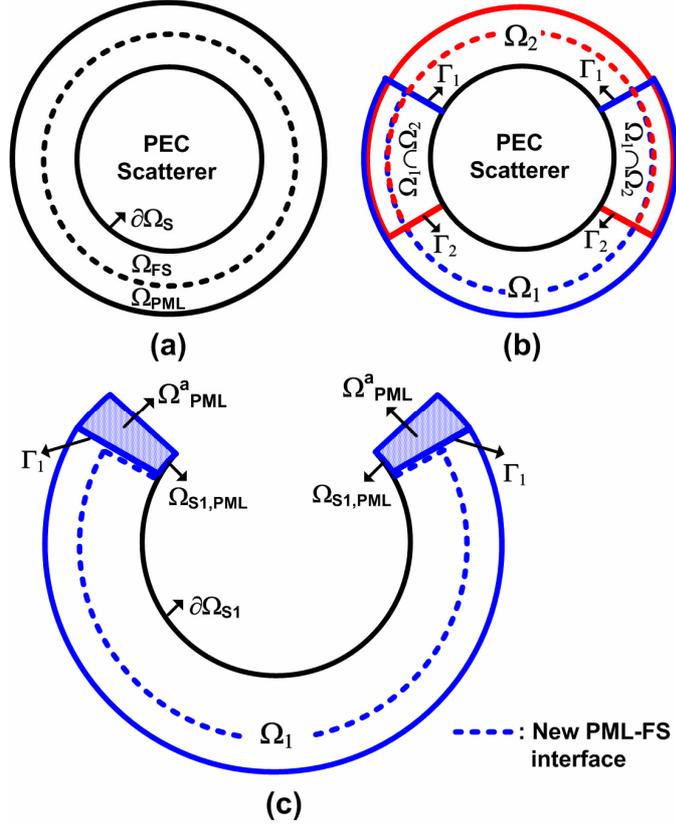


Figure 5.3. FB-DDM for sphere with two subdomains: (a) Original problem, (b) Partitioned problem, (c) First subdomain with additional PML region.

In order to better grasp the implementation procedure of the algorithm without loosing any generality, we assume that the original problem is decomposed into three subdomains as demonstrated in Figure 5.4 for cube and in Figure 5.5 for sphere. In the first forward iteration of the FB-DDM algorithm, we first solve the BVP in the first domain Ω_1 using additional PML region (Ω_{PML}^a) attached to the boundary Γ_1 . More specifically, for the 2D scattering problem, we solve the following BVP governed by the Helmholtz equation with a Dirichlet type BC

Inside the free-space region:

$$\nabla^2 E_1(\vec{r}) + k^2 E_1(\vec{r}) = 0 \quad \text{in } \Omega_{\text{FS1}} \quad (5.4.a)$$

$$\text{with BC: } E_1(\vec{r}) = -E^{\text{inc}}(\vec{r}) \quad \text{on } \partial\Omega_{\text{S1}} \quad (5.4.b)$$

Inside the PML region:

$$\tilde{\nabla}^2 E_1^c(\tilde{\vec{r}}) + k^2 E_1^c(\tilde{\vec{r}}) = 0 \quad \text{in } \Omega_{\text{PML1}} \cup \Omega_{\text{PML}}^a \quad (5.4.c)$$

$$\text{with BC: } E_1^c(\tilde{\vec{r}}) = -E^{inc}(\tilde{\vec{r}}) \quad \text{on } \partial\Omega_{\text{S1,PML}} \quad (5.4.d)$$

where $E^{inc}(\vec{r})$ and $E^{inc}(\tilde{\vec{r}})$ are the incident plane waves evaluated in real and complex coordinates, respectively, $E_1(\vec{r})$ and $E_1^c(\tilde{\vec{r}})$ are scattered fields in real and complex coordinates, respectively, $\tilde{\nabla}$ is the nabla operator in complex space, and the superscript ‘c’ represents the analytical continuation to complex space. It is worth mentioning that we have simplified the notations in order to avoid confusion, thus, $E_1 = E_{z1}^s$ and $E^{inc} = E_z^{inc}$ using the notations in Chapter 2. Similarly, for the 3D scattering problem, we solve the following BVP governed by the vector wave equation with a Dirichlet type BC

Inside the free-space region:

$$\nabla \times \nabla \times \vec{E}_1(\vec{r}) - k^2 \vec{E}_1(\vec{r}) = 0 \quad \text{in } \Omega_{\text{FS1}} \quad (5.5.a)$$

$$\text{with BC: } \hat{n} \times \vec{E}_1(\vec{r}) = -\hat{n} \times \vec{E}^{inc}(\vec{r}) \quad \text{on } \partial\Omega_{\text{S1}} \quad (5.5.b)$$

Inside the PML region:

$$\tilde{\nabla} \times \tilde{\nabla} \times \vec{E}_1^c(\tilde{\vec{r}}) - k^2 \vec{E}_1^c(\tilde{\vec{r}}) = 0 \quad \text{in } \Omega_{\text{PML1}} \cup \Omega_{\text{PML}}^a \quad (5.5.c)$$

$$\text{with BC: } \hat{n} \times \vec{E}_1^c(\tilde{\vec{r}}) = -\hat{n} \times \vec{E}^{inc}(\tilde{\vec{r}}) \quad \text{on } \partial\Omega_{\text{S1,PML}} \quad (5.5.d)$$

where $\vec{E}^{inc}(\vec{r})$ and $\vec{E}^{inc}(\tilde{\vec{r}})$ are the incident plane waves evaluated in real and complex coordinates, respectively, $\vec{E}_1(\vec{r})$ and $\vec{E}_1^c(\tilde{\vec{r}})$ are scattered fields in real and complex coordinates, respectively. Using the notations in Chapter 3, $\vec{E}_1 = \vec{E}_1^s$. It is clear from (5.4.d) and (5.5.d) that the field also decays smoothly along the scatterer boundary which is inside the PML region (i.e., $\partial\Omega_{\text{S1,PML}}$) due to the locally-conformal PML implementation. The region Ω_{PML}^a provides a smooth field decay along the direction of the arrows pointing to the (Ω_{PML}^a)

symbols in the figures. If arbitrary ‘incorrect’ initial values were used along Γ_1 (as in the classical Schwarz method) instead of using Ω_{PML}^a , there would occur some artificial reflections from the boundary Γ_1 . Thus, Ω_{PML}^a gets rid of these reflections, and provides a more accurate solution inside the first subdomain.

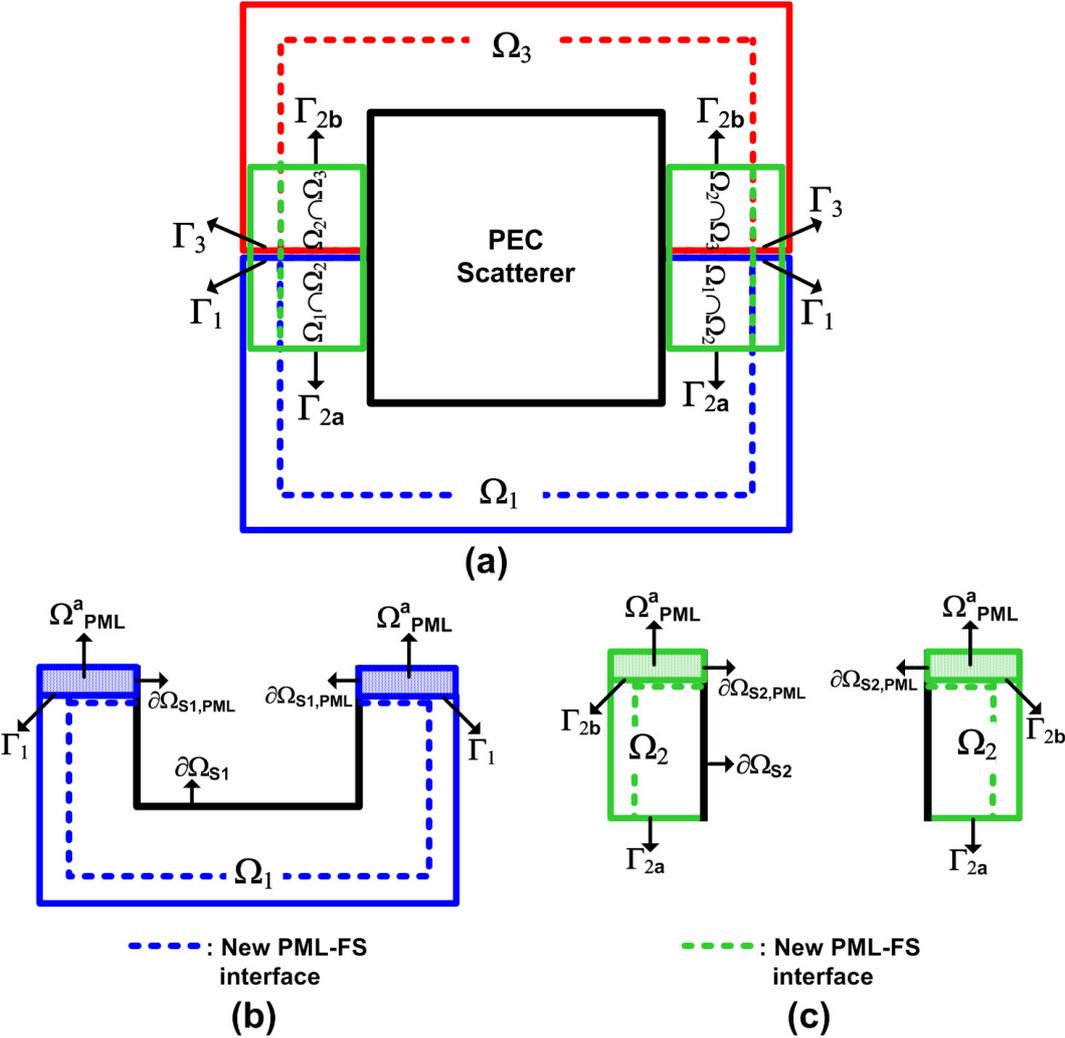


Figure 5.4. FB-DDM for cube with three subdomains: (a) Partitioned problem, (b-c) First and second subdomains with additional PML regions.

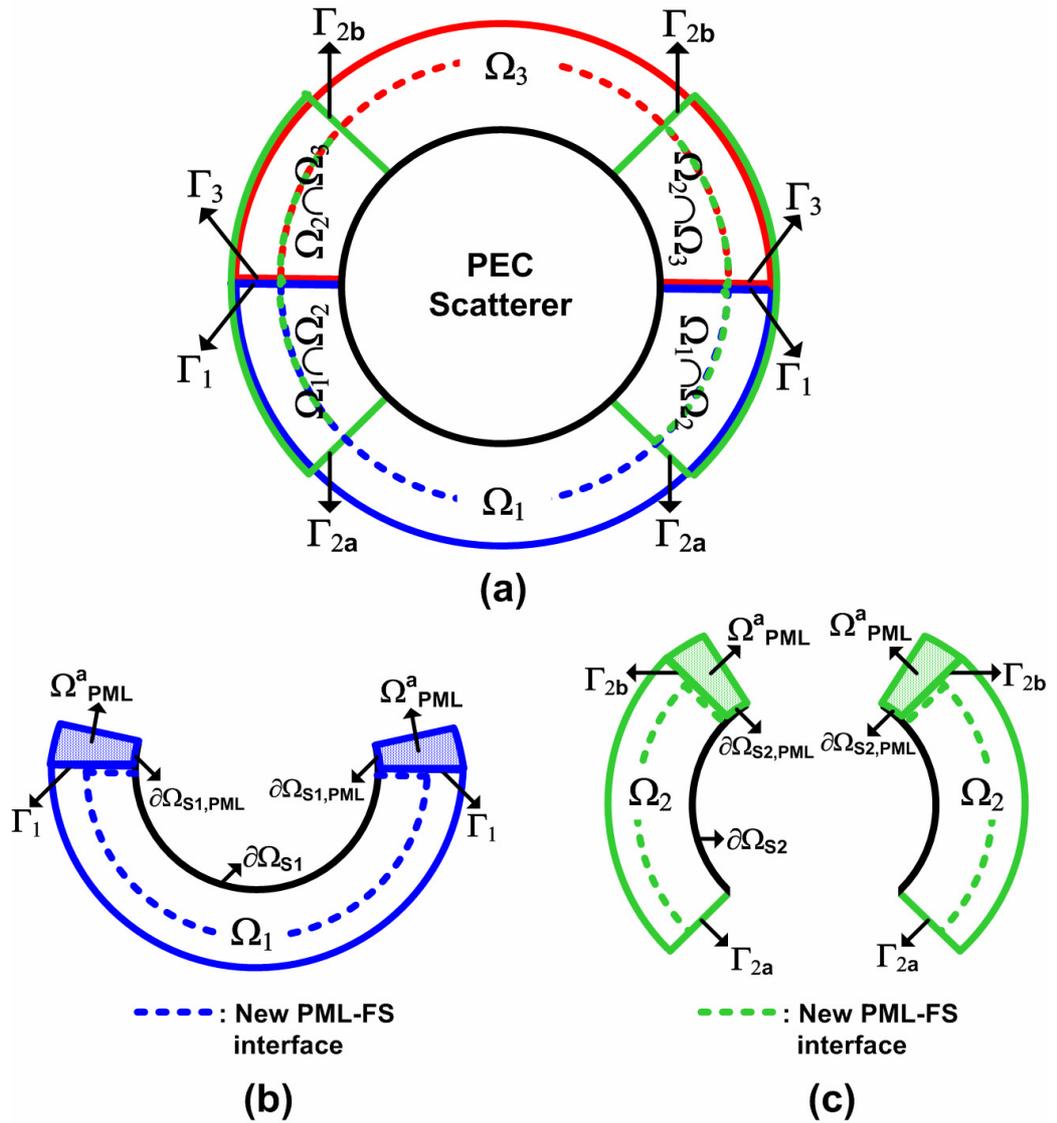


Figure 5.5. FB-DDM for sphere with three subdomains: (a) Partitioned problem, (b-c) First and second subdomains with additional PML regions.

Since the solution of the problem on Ω_1 induces field values along the boundary Γ_{2a} , we then solve the problem on the second subdomain Ω_2 using the field values along Γ_{2a} , and employing Ω_{PML}^a attached to the boundary Γ_{2b} . More precisely, we solve the following BVP in 2D

Inside the free-space region:

$$\nabla^2 E_2(\vec{r}) + k^2 E_2(\vec{r}) = 0 \quad \text{in } \Omega_{\text{FS2}} \quad (5.6.a)$$

$$\text{with BC: } E_2(\vec{r}) = \begin{cases} -E^{\text{inc}}(\vec{r}) & \text{on } \partial\Omega_{\text{S2}} \\ E_1(\vec{r}) & \text{on } \Gamma_{2a} \end{cases} \quad (5.6.b)$$

Inside the PML region:

$$\tilde{\nabla}^2 E_2^c(\tilde{\vec{r}}) + k^2 E_2^c(\tilde{\vec{r}}) = 0 \quad \text{in } \Omega_{\text{PML2}} \cup \Omega_{\text{PML}}^a \quad (5.6.c)$$

$$\text{with BC: } E_2^c(\tilde{\vec{r}}) = -E^{\text{inc}}(\tilde{\vec{r}}) \quad \text{on } \partial\Omega_{\text{S2,PML}} \quad (5.6.d)$$

Similarly, we solve the following BVP in 3D

Inside the free-space region:

$$\nabla \times \nabla \times \vec{E}_2(\vec{r}) - k^2 \vec{E}_2(\vec{r}) = 0 \quad \text{in } \Omega_{\text{FS2}} \quad (5.7.a)$$

$$\text{with BC: } \hat{n} \times \vec{E}_2(\vec{r}) = \begin{cases} -\hat{n} \times \vec{E}^{\text{inc}}(\vec{r}) & \text{on } \partial\Omega_{\text{S2}} \\ \hat{n} \times \vec{E}_1(\vec{r}) & \text{on } \Gamma_{2a} \end{cases} \quad (5.7.b)$$

Inside the PML region:

$$\tilde{\nabla} \times \tilde{\nabla} \times \vec{E}_2^c(\tilde{\vec{r}}) - k^2 \vec{E}_2^c(\tilde{\vec{r}}) = 0 \quad \text{in } \Omega_{\text{PML2}} \cup \Omega_{\text{PML}}^a \quad (5.7.c)$$

$$\text{with BC: } \hat{n} \times \vec{E}_2^c(\tilde{\vec{r}}) = -\hat{n} \times \vec{E}^{\text{inc}}(\tilde{\vec{r}}) \quad \text{on } \partial\Omega_{\text{S2,PML}} \quad (5.7.d)$$

Since the solution of the problem on Ω_2 yields field values along the boundary Γ_3 , we then solve the problem on the third subdomain Ω_3 using the values along Γ_3 (without using Ω_{PML}^a). More precisely, we solve the following BVP in 2D

Inside the free-space region:

$$\nabla^2 E_3(\vec{r}) + k^2 E_3(\vec{r}) = 0 \quad \text{in } \Omega_{\text{FS3}} \quad (5.8.a)$$

$$\text{with BC: } E_3(\vec{r}) = \begin{cases} -E^{\text{inc}}(\vec{r}) & \text{on } \partial\Omega_{\text{S3}} \\ E_2(\vec{r}) & \text{on } \Gamma_3 \end{cases} \quad (5.8.b)$$

Inside the PML region:

$$\tilde{\nabla}^2 E_3^c(\tilde{\vec{r}}) + k^2 E_3^c(\tilde{\vec{r}}) = 0 \quad \text{in } \Omega_{\text{PML}3} \quad (5.8.c)$$

and the following BVP in 3D

Inside the free-space region:

$$\nabla \times \nabla \times \vec{E}_3(\vec{r}) - k^2 \vec{E}_3(\vec{r}) = 0 \quad \text{in } \Omega_{\text{FS}3} \quad (5.9.a)$$

$$\text{with BC: } \hat{n} \times \vec{E}_3(\vec{r}) = \begin{cases} -\hat{n} \times \vec{E}^{inc}(\vec{r}) & \text{on } \partial\Omega_{\text{S}3} \\ \hat{n} \times \vec{E}_2(\vec{r}) & \text{on } \Gamma_3 \end{cases} \quad (5.9.b)$$

Inside the PML region:

$$\tilde{\nabla} \times \tilde{\nabla} \times \vec{E}_3^c(\tilde{\vec{r}}) - k^2 \vec{E}_3^c(\tilde{\vec{r}}) = 0 \quad \text{in } \Omega_{\text{PML}3} \quad (5.9.c)$$

In this way, the first forward iteration of the FB-DDM algorithm is completed. In problems involving ‘smooth’ geometries, the first forward iteration is sufficient to get accurate results by properly defining the subdomains and the additional PML regions. We leave the detailed discussion of these issues to Section 5.1.3, and we here consider only the basic steps of the algorithm. We should also note that we calculate the mean value of the field values in the overlapping regions. For instance, in the overlapping region where Ω_1 and Ω_2 intersect, we assume that the value of the field is the mean of \vec{E}_1 and \vec{E}_2 .

Next, let us assume that the first forward iteration fails to provide an accurate approximate solution. In this case, similar to the classical Schwarz method, we carry out the backward iteration by first solving the problem on Ω_2 using the new boundary values along Γ_{2a} and Γ_{2b} , and then solving the problem on Ω_1 using the new boundary values along Γ_1 . Then, we implement the forward iteration by solving the problem on Ω_2 using the new boundary values along Γ_{2a} and Γ_{2b} , and then solving the problem on Ω_3 using the new boundary values along Γ_3 . Similarly, we continue with the subsequent forward-backward

iterations until convergence is achieved. In these iterations, we do not employ any additional PML region. To be more precise, for the i^{th} subdomain at the n^{th} step (i.e., the step denotes each stage where a single subproblem is solved), we solve the following BVP in 2D

Inside the free-space region:

$$\nabla^2 E_i(\vec{r}) + k^2 E_i(\vec{r}) = 0 \quad \text{in } \Omega_{\text{FS},i} \quad (5.10.a)$$

$$\text{with BC: } E_i(\vec{r}) = \begin{cases} -E^{\text{inc}}(\vec{r}) & \text{on } \partial\Omega_{\text{S},i} \\ E^{n-1}(\vec{r}) & \text{on } \Gamma_i \end{cases} \quad (5.10.b)$$

Inside the PML region:

$$\tilde{\nabla}^2 E_i^c(\tilde{\vec{r}}) + k^2 E_i^c(\tilde{\vec{r}}) = 0 \quad \text{in } \Omega_{\text{PML},i} \quad (5.10.c)$$

and the following BVP in 3D

Inside the free-space region:

$$\nabla \times \nabla \times \vec{E}_i(\vec{r}) - k^2 \vec{E}_i(\vec{r}) = 0 \quad \text{in } \Omega_{\text{FS},i} \quad (5.11.a)$$

$$\text{with BC: } \hat{n} \times \vec{E}_i(\vec{r}) = \begin{cases} -\hat{n} \times \vec{E}^{\text{inc}}(\vec{r}) & \text{on } \partial\Omega_{\text{S},i} \\ \hat{n} \times \vec{E}^{n-1}(\vec{r}) & \text{on } \Gamma_i \end{cases} \quad (5.11.b)$$

Inside the PML region:

$$\tilde{\nabla} \times \tilde{\nabla} \times \vec{E}_i^c(\tilde{\vec{r}}) - k^2 \vec{E}_i^c(\tilde{\vec{r}}) = 0 \quad \text{in } \Omega_{\text{PML},i} \quad (5.11.c)$$

where $E^{n-1}(\vec{r})$ and $\vec{E}^{n-1}(\vec{r})$ denote the most recent field values inside the whole domain at the $(n-1)^{\text{th}}$ step. Since the first forward iteration provides an ‘almost correct’ initial guess for the solution along the interfaces as well as inside the original domain, the algorithm converges faster after the first forward iteration, compared to the classical Schwarz method with arbitrary initial guess.

The FB-DDM algorithm can be generalized to cases containing an arbitrary number of subdomains in a straightforward manner. The basic steps of the FB-DDM algorithm for N subdomains are summarized as a flowchart in Figure 5.6.

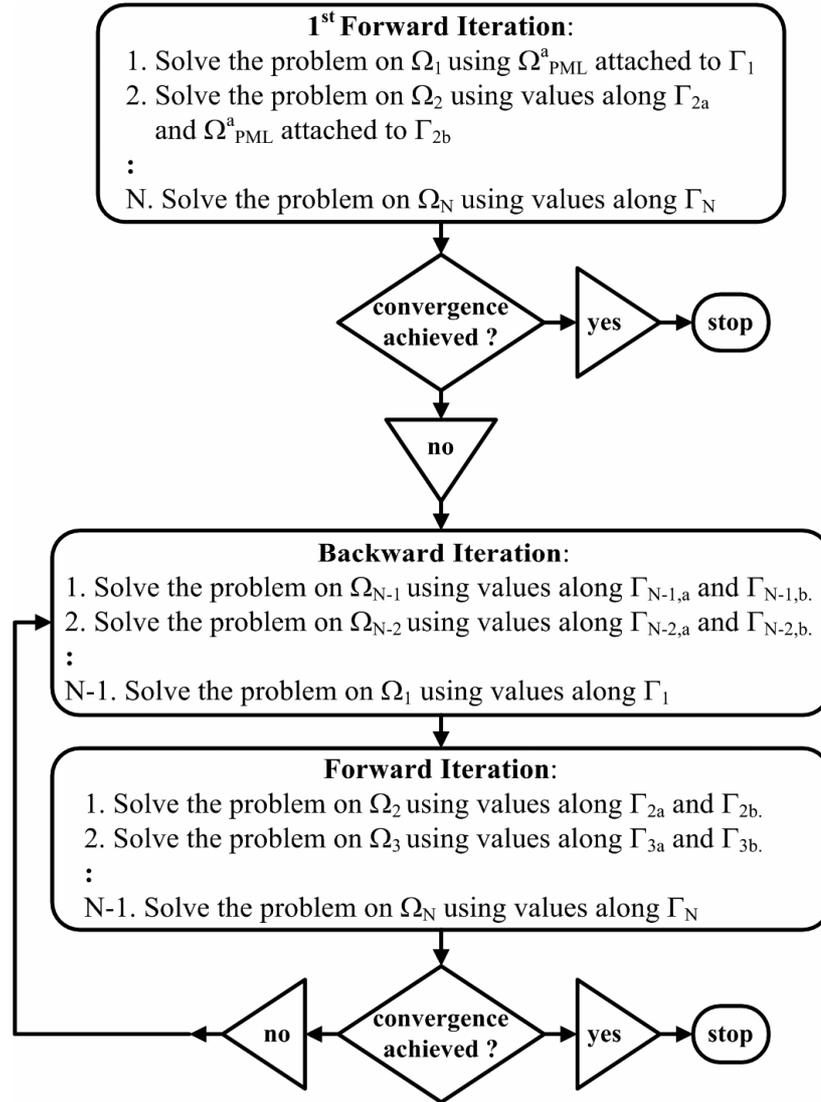


Figure 5.6. Flowchart of the FB-DDM algorithm with N subdomains.

5.1.3. Implementation Issues

In the implementation of the FB-DDM algorithm in a scattering problem, some key issues should be taken into consideration in order to increase the efficiency and the convergence rate of the algorithm. Although we touch on these issues briefly in Section 5.1.2, we will explain them in detail in this section.

It should be obvious that the success of the FB-DDM algorithm directly depends on the first forward iteration, where the problems over the subdomains are solved by introducing additional PML regions (except in the last subdomain), because the first forward iteration provides an initial guess for the forthcoming forward-backward iterations analogous to those of the classical Schwarz method. Therefore, if the first forward iteration provides a guess ‘close’ to the global solution, the following forward-backward iterations are expected to converge faster. In order to perform a successful first forward iteration; the partitioning of the original domain, the selection of the order of the subdomains, and the design of the additional attached PML regions should be done in accordance with the expected field behavior inside the original domain. If we consider the problem of scattering from ‘smooth’ objects, the first subdomain should be coincident with the most illuminated part of the original domain. In this manner, the additional PML region attached to the first subdomain will be able to absorb properly the scattered field, if this PML region is designed in such a way that the direction of the decay is in conformity with the original field behavior in this PML region. If these conditions are met, we may obtain an almost accurate field solution inside the first subdomain. This is also applicable for the remaining subdomains, but the design of the additional PML region is obviously more critical for the regions where the scattered field has a considerable magnitude. To be more precise, let us consider the 2D scattering from an infinitely-long cylinder with a circular cross-section. Assuming that the cylinder cross-section is ‘electrically-large’, the principles of ray optics can be employed. We assume that the plane wave is incident from the left side of the cylinder, as shown in Figure 5.7. The scattered field consists of four terms: the reflected field, the shadow forming field, the diffracted field originating at P_1 , and the diffracted field originating at P_2 (see Figure 5.7(c) and (d)) [80]. The plot in Figure 5.7(c) shows the rays reflected by the circular cylinder and the rays of the shadow forming field, and the plot in Figure 5.7(d) shows the diffracted rays emanating from P_1 . Thus, if we choose the subdomains as in Figure 5.7(a) in conformity with the field

behavior, we can get an almost accurate result in the first forward iteration. This is due to the fact that the direction of decay in the additional PML region (implied by the direction of arrows in Figure 5.7(b)) matches the direction of scattered field in this region. Although this requirement related to the additional PML region is no longer satisfied in the deep shadow region of the domain, this does not cause a big difficulty since the PML region can still absorb the field because there is a safe angle margin (i.e., 45°) between the direction of the decay and the direction of the scattered field (see the multi-center PML in Chapter 4).

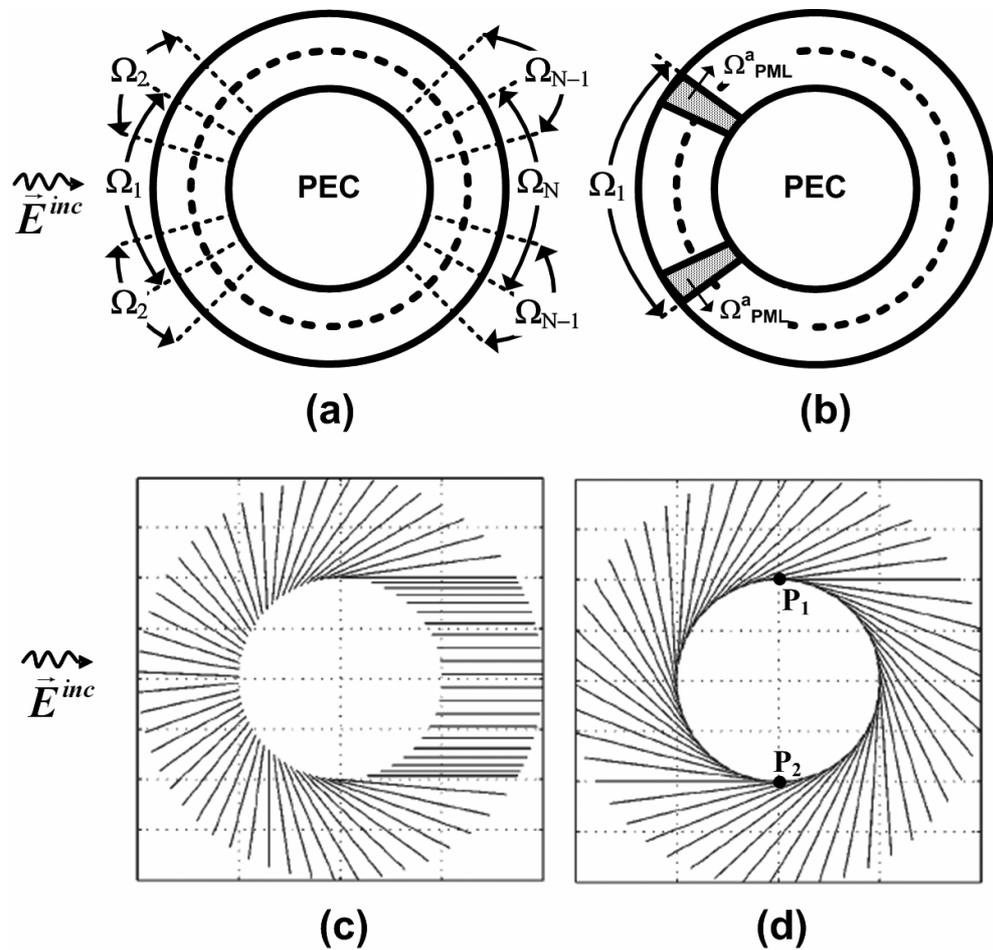


Figure 5.7. Scattering from circular cylinder: (a) Partitioned problem, (b) First subdomain with additional PML region, (c) Reflected rays and the rays of shadow forming beam, (d) Diffracted rays emanating from P_1 .

The main reason why the first forward iteration is sufficient to get accurate results in the above-mentioned manner is that none of the subdomains experiences any *considerable* field returns from other subdomains (excluding surface diffracted waves due to their rapid decaying behavior, especially in the shadow region). Therefore, the subdomains are almost independent from each other, particularly at high frequencies where the laws of ray optics dominate. If, for instance, there were a considerable field return to the first subdomain from the other subdomains in the form of a reflected field, the first subdomain would not be aware of this reflected field, and we would solve the problem in the first subdomain as if it is isolated from other subdomains. In this case, the first forward iteration is not sufficient for obtaining an accurate solution and we need additional forward-backward iterations. In order to clarify this discussion, we consider two specific scattering applications: (i) scattering from a T-shaped cylinder [see Figure 5.8(a)], and (ii) scattering from an H-shaped cylinder [see Figure 5.8(b)], supposing that the plane wave is incident from the right side of the scatterer. If there is a large amount of field activity (field interactions such as reflections, diffractions) between different subdomains in the illuminated part of the original domain, the first subdomain should always be chosen to be coincident to the region where the field activity is dominant, as in the case of T-shaped cylinder. If there is no reflected field return to the first subdomain from the other subdomains in the illuminated part (even if there occurs some field activity in the shadow region in the form of diffractions), the first subdomain should be chosen as the most illuminated part of the original domain as in the case of H-shaped cylinder. In both cases, the algorithm requires some forward-backward sweeps for convergence due to the field interactions between the subdomains. However, we expect that the problem for the H-shaped cylinder should converge faster because the first subdomain can be considered as ‘more isolated’ as compared to the problem for T-shaped cylinder due to the geometry of the problem and the direction of the plane wave illumination. Thus, we can conclude that the performance of the first subdomain is crucial to improve the

convergence rate of the algorithm. The implications of these observations related to these examples will be demonstrated numerically in Chapter 7.

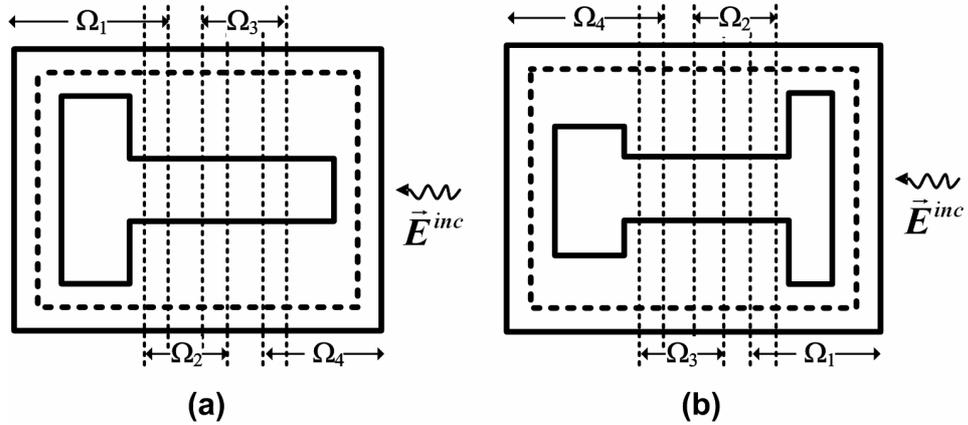


Figure 5.8. Scattering from challenging cylinders: (a) T-shaped, (b) H-shaped.

5.1.4. Special Problems

Bearing in mind the above discussion in Section 5.1.3, we can also solve some scattering problems involving special geometries, such as inlets (or U-shaped cylinders in 2D) and plates, in just the first forward iteration or at least with few number of iterations. In order to solve such problems, we modify slightly the FB-DDM algorithm without altering the main philosophy behind it.

5.1.4.1. Inlets (or U-shaped cylinders)

First, we consider the scattering from an inlet or U-shaped cylinder in 2D, as shown in Figure 5.9(a). We choose the direction of the plane wave as shown in this figure because this creates possibly the most challenging situation where there occurs many field interactions inside the inlet. If we choose the subdomains as in Figure 5.9(b) and apply the FB-DDM algorithm in the usual way, the algorithm converges to the solution in the first forward iteration (Case 1). We note that the first subdomain includes the inner part of the inlet.

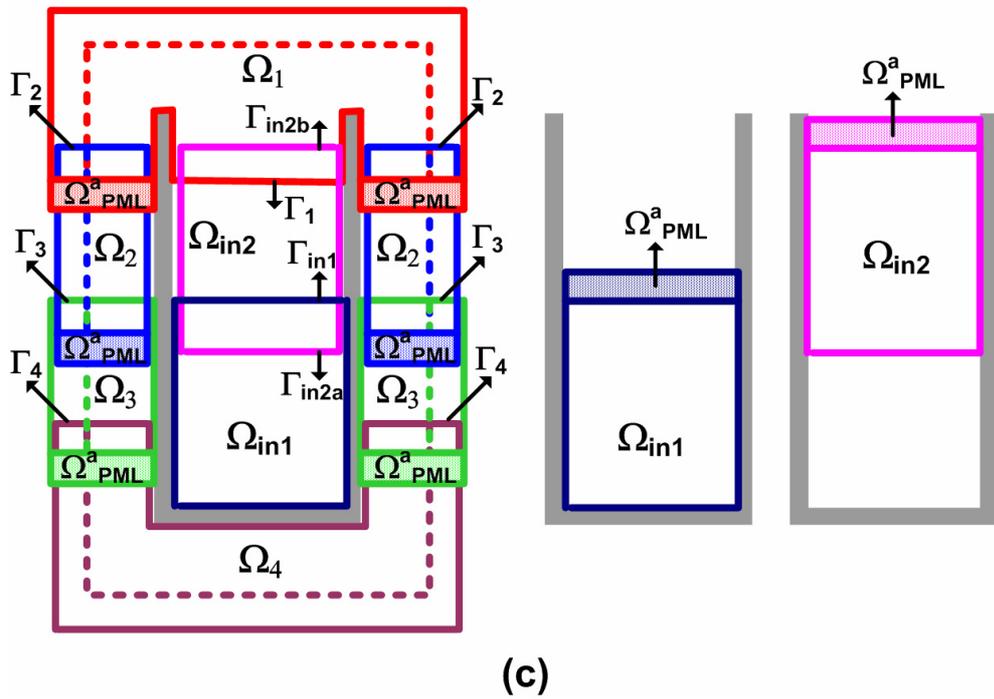
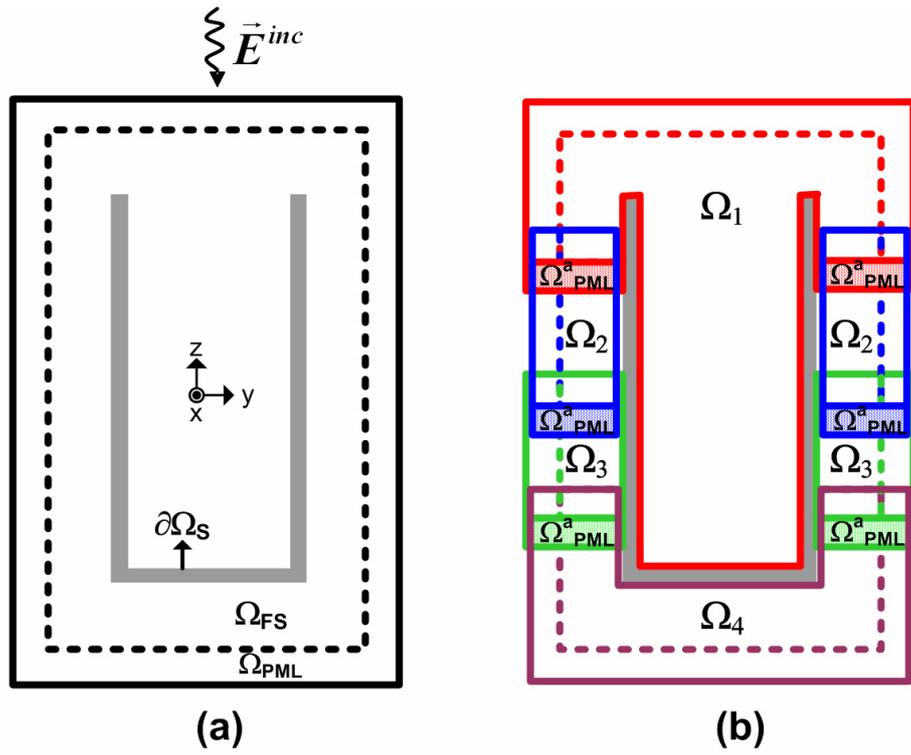


Figure 5.9. FB-DDM for inlet: (a) Original problem, (b) Partitioned problem (Case 1), (c) Partitioned problem with inner subdomains (Case 2).

A single drawback in the above approach may be the difficulty in the solution of the first subdomain problem due to larger number of unknowns if the inlet is very deep. As an alternative way (Case 2), we partition the inner part of the inlet into some number of subdomains, which are shown as $\Omega_{in,i}$ in Figure 5.9(c). Before solving the problems over outer subdomains denoted as Ω_i , we first solve the problems over inner subdomains iteratively until convergence is achieved inside the inlet. Afterwards, we process the outer subdomains in a single forward iteration using the additional PML regions. While iterating the inner subdomains, we start with the first inner subdomain Ω_{in1} with the additional PML region. Then, we solve the problem on the second inner subdomain Ω_{in2} using the new values along Γ_{in2} and employing Ω_{PML}^a . In this way, we complete the first inner forward iteration. If the iterations do not converge to the solution inside the inlet, we again solve the problem on Ω_{in1} using the new values along Γ_{in1} and without using any additional PML region (backward iteration), and then we solve the problem on Ω_{in2} using the new values along Γ_{in2} and employing Ω_{PML}^a (forward iteration). We repeat the forward-backward iterations in the same way until convergence is achieved inside the inlet. After finishing the inner iterations, we solve the problem on the first outer subdomain Ω_1 using Ω_{PML}^a and the new values along Γ_1 . Then, we solve the problem on Ω_2 using Ω_{PML}^a and the new values along Γ_2 , and we solve the problem on Ω_3 using Ω_{PML}^a and the new values along Γ_3 . Finally, we solve the problem on Ω_4 using the new values along Γ_4 . We should mention that this alternative way (Case 2) is less successful than Case 1, because the position of the inner subdomains has some effect on the accuracy and the convergence rate of the algorithm due to the field behavior (the structure of the standing wave pattern) inside the inlet. We recommend that the inner inlet subdomains with boundaries chosen at the minimum regions of the standing wave pattern predicted inside the inlet are good candidates for better convergence of the algorithm, because of the increased performance of the

additional PML regions in these minimum regions. As a summary, we draw a flowchart to show the basic steps for N number of outer subdomains and N_i number of inner subdomains in Figure 5.10.

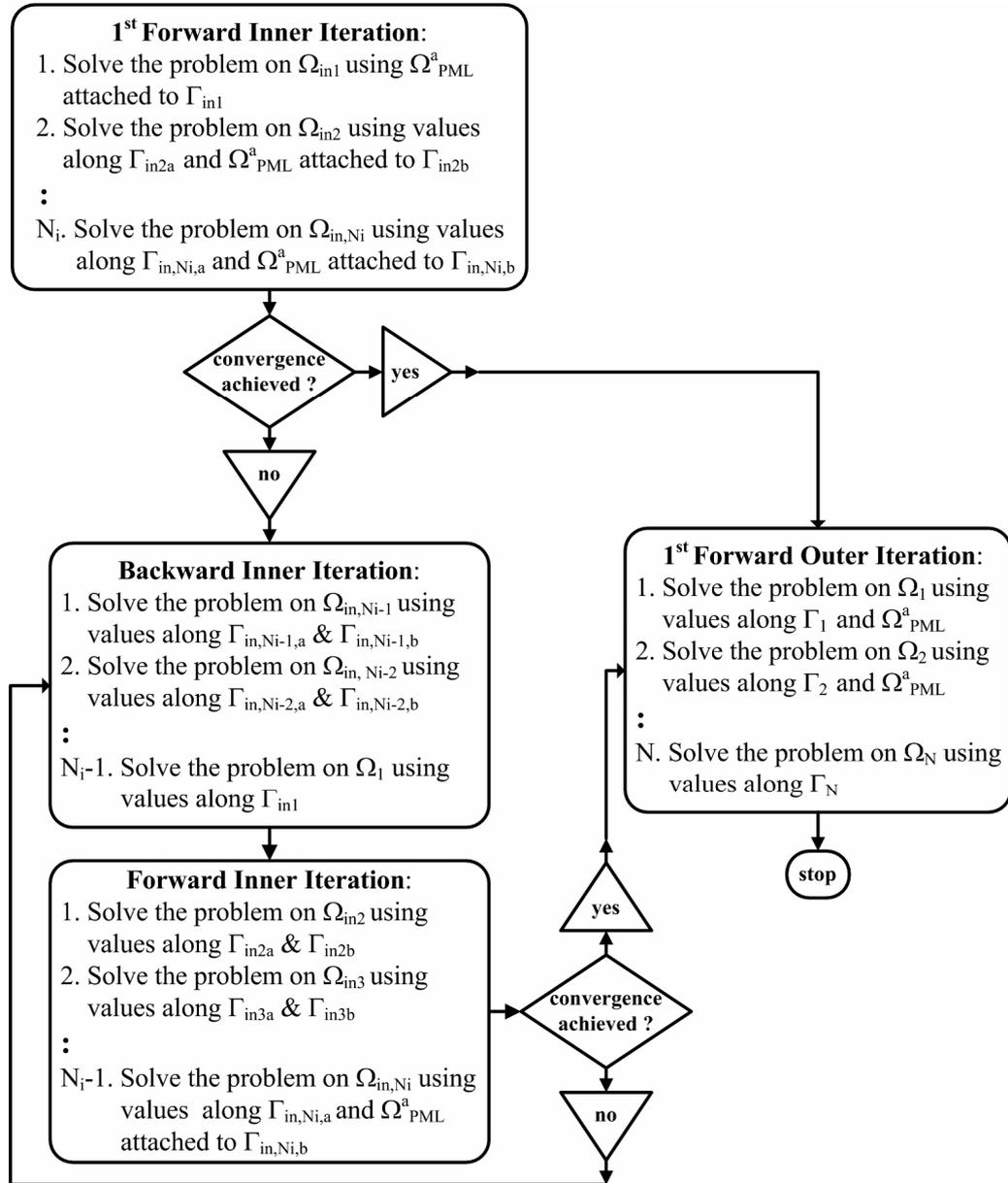


Figure 5.10. Flowchart of the FB-DDM algorithm for inlets (Case 2) with N_i inner subdomains and N outer subdomains.

5.1.4.2. Plates

As a second special problem, we consider the scattering from plates, as shown in Figure 5.11(a). It is evident that we have a limited number of possibilities for different partitioning schemes for plates. However, we can still apply the FB-DDM algorithm in the usual manner with x -partitioning or y -partitioning scheme, as shown in Figure 5.11(b) and Figure 5.11(c), respectively. Although we do not show in the figures, the subdomains have additional PML regions in the first forward iteration. The choice of the partitioning scheme (x or y) depends on the dimension of the plate and/or the polarization of the incident plane wave. If, for instance, the incident plane wave is polarized in the x -direction (or y -direction), it is better to choose the y -partitioning (or x -partitioning) scheme to get a better result in the first forward iteration. However, in a general case where the polarization is arbitrary, we can still obtain convergence by refining the solution during forward-backward iterations. We should note that the order of domains is not very important due to the almost equal illumination of the subdomains (even in the case of oblique incidence). In addition, if the dimension of the plate along the x -direction (or y -direction) is considerably larger than its dimension along the y -direction (or x -direction), it is better to decide on the x -partitioning (or y -partitioning) scheme in order to decrease the number of unknowns in each subdomain.

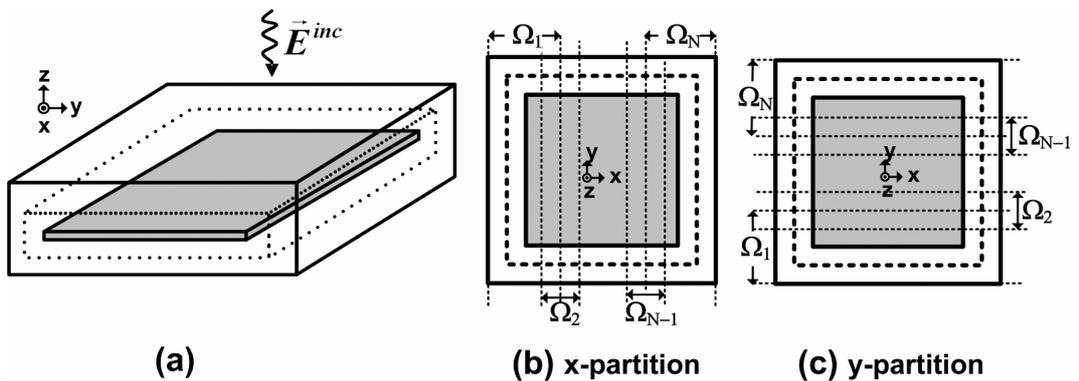


Figure 5.11. FB-DDM for plate: (a) Original problem, (b) x -partitioned problem, (c) y -partitioned problem.

Alternatively, we can apply a ‘puzzle-like’ partitioning to the plate with some slight modifications in the FB-DDM algorithm. As an example, we decompose the original domain of the square plate into 2×2 subdomains, as illustrated in Figure 5.12. We first solve the problem over an arbitrary corner subdomain Ω_{11} using Ω_{PML}^a attached to the boundaries Γ_{11x} and Γ_{11y} . Then, we solve the problems on Ω_{12} and Ω_{21} in parallel. That is, we solve the problem on Ω_{12} using the new values along Γ_{12x} and Ω_{PML}^a attached to the boundary Γ_{12y} . Similarly, we solve the problem on Ω_{21} using the new values along Γ_{21y} and Ω_{PML}^a attached to the boundary Γ_{21x} . Finally, we solve the problem on Ω_{22} using the new values along Γ_{22x} and Γ_{22y} . In this manner, we end up with the first forward iteration. If the result fails to converge to the solution, we process the backward iteration in such a way that we first solve the problems on Ω_{12} and Ω_{21} in parallel using the new values along both boundaries (i.e., Γ_{12x} and Γ_{12y} for Ω_{12} , and Γ_{21x} and Γ_{21y} for Ω_{21}), and then we solve the problem on Ω_{11} using the new values along Γ_{11x} and Γ_{11y} . After the backward iteration, we proceed with the forward iteration by solving the problems on Ω_{12} and Ω_{21} in parallel, and then solving the problem on Ω_{22} using the new values along their corresponding boundaries. We continue with the next forward-backward iterations until we reach a result sufficiently close to the solution. We should mention that, in some cases, the puzzle-like partitioning scheme may be less successful than the x - or y -partitioning schemes in terms of the convergence rate due to the polarization of the plane wave, as described above. Therefore, for instance, if the incident plane wave is polarized along the x -direction, we recommend the y -partitioning scheme instead of the puzzle-like partitioning in order to get a good result in the first forward iteration. On the other hand, for problems where the polarization is arbitrary, the puzzle-like partitioning scheme may be more attractive to reduce the number of unknowns in the subdomains. The FB-DDM algorithm with 2×2 subdomains can be generalized to $N_x \times N_y$ subdomains in a straightforward manner, as illustrated in Figure 5.13 as a flowchart.

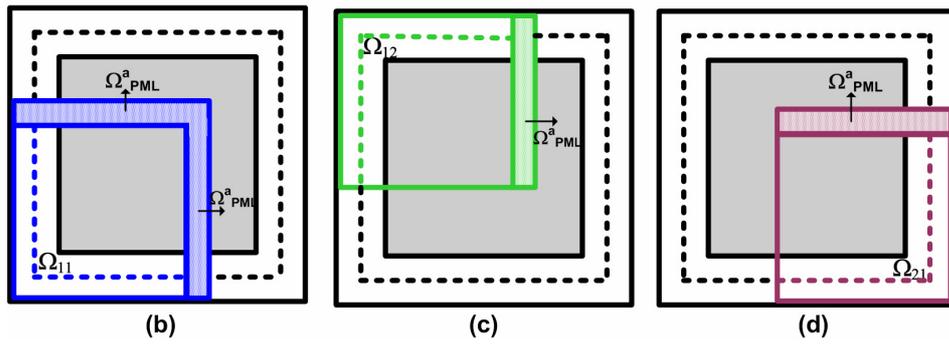
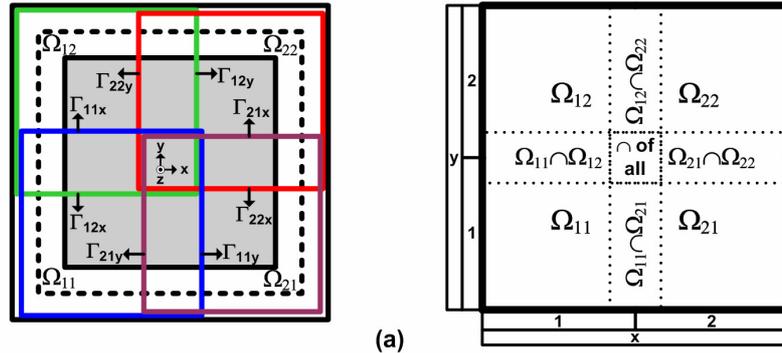


Figure 5.12. (a) ‘Puzzle-like’ partitioning of plate with 2x2 subdomains, (b-c-d) $\Omega_{11}, \Omega_{12}, \Omega_{21}$ with additional PML regions.

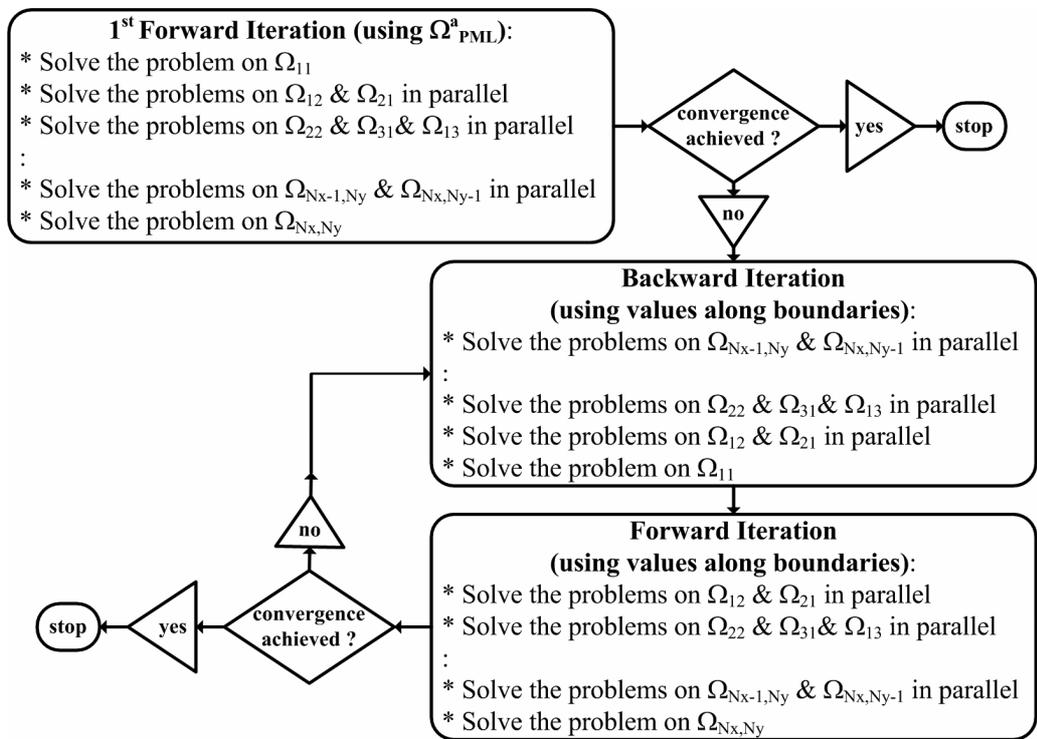


Figure 5.13. Flowchart for ‘puzzle-like’ partitioning with $N_x \times N_y$ subdomains.

5.2. Iterative Leap-field DDM (ILF-DDM)

The Iterative Leap-field DDM (ILF-DDM) is based on the consecutive employment of the FEM and the Huygens' Principle. Roughly speaking, in each iteration of the ILF-DDM, each subproblem is solved and equivalent currents are calculated using the fields radiated from the sources contained in the remaining subdomains, and these currents are reradiated over all other subdomains, until convergence is achieved. Although the proposed technique is similar to [76] which has employed the MoM/PO hybrid technique for multiple distinct objects, the crucial difference in the ILF-DDM approach is the utilization of the locally-conformal PML concept, which increases the range of applicability to *not only* cases involving multiple objects *but also* problems containing a single 'challenging' object in FEM applications. Furthermore, the ILF-DDM has some special features in terms of the easy numerical implementation of the algorithm due to the nature of the locally-conformal PML approach and the FEM itself. For example, the method yields 'small' subdomains due to the fact that the locally-conformal PML can be designed as conformal and very close to the surface of the object. The solution of each small subproblem is nothing more than a solution of an ordinary scattering problem having less number of unknowns, but with different boundary conditions. In addition, the subproblems can be solved in parallel in each iteration. Thus, ILF-DDM can reduce both the memory storage requirements and the computation time significantly.

The organization of this section is as follows: In Section 5.2.1, we overview briefly the Huygens' Principle and the radiation integrals in electromagnetics. In Section 5.2.2, we draw a general picture for the implementation procedure of the ILF-DDM algorithm in the problem of electromagnetic scattering from PEC objects. Although the algorithm can easily be extended to cases involving scatterers with arbitrarily-defined constitutive parameters, we restrict our analysis to PEC objects for the sake of brevity.

5.2.1. Review of the Huygens' Principle and Radiation Integrals

The Huygens' Principle is the replacement of actual sources by equivalent (or fictitious) sources, in such a way that both the equivalent sources and the actual sources produce the same fields in a region. For the purpose of visualization of the principle, we consider a problem in Figure 5.14(a) where we want to calculate the fields radiated by some sources and objects into the region which is outside of S . Then, we replace the original problem with an equivalent problem in Figure 5.14(b) by employing a closed surface S enclosing the sources and the objects. In order to yield the same fields outside S , the equivalent sources surrounding S should be defined from the knowledge of the fields on S as follows

$$\vec{J}_s = \hat{n} \times \vec{H} \quad (5.12.a)$$

$$\vec{M}_s = \vec{E} \times \hat{n} \quad (5.12.b)$$

where \hat{n} is the unit normal vector directed outward on S .

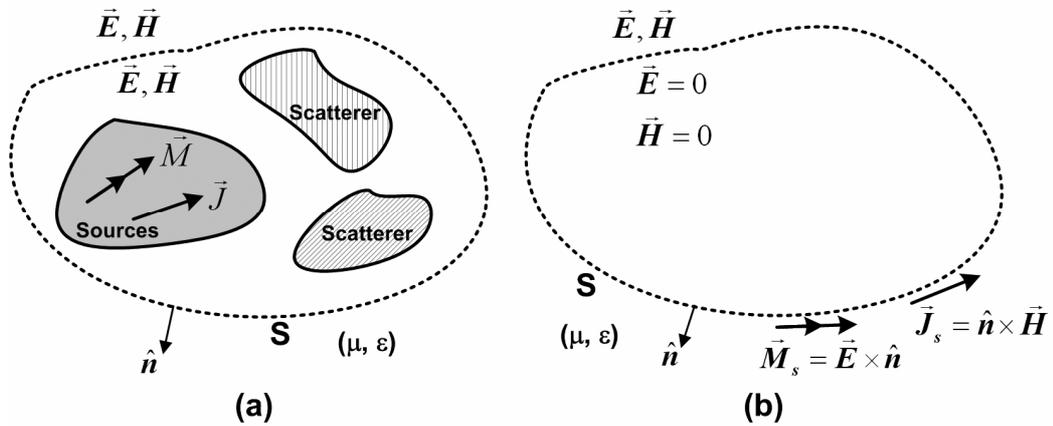


Figure 5.14. Huygens' Principle: (a) Original model, (b) Equivalent model.

In the problem of scattering from a single PEC object, if we choose the surface S to be very close to the surface of the object, the equivalent current \vec{M}_s vanishes since the total electric field is zero on the surface of the object. Therefore, \vec{J}_s residing on the surface of the object becomes the true equivalent current. The fields radiated by the current \vec{J}_s are calculated by means of the well-known radiation integrals which are derived using the Green's functions. In 2D scattering problems assuming the TM_z case (i.e., $\vec{E} = \hat{a}_z E_z$ and $\vec{J}_s = \hat{a}_z J_z$), the electric field radiated by the current density into the free-space is given by

$$E_z = -\frac{k\eta_0}{4} \oint_C J_z(\vec{r}') H_0^{(2)}(k|\vec{r} - \vec{r}'|) dl' \quad (5.13)$$

where $H_0^{(2)}$ is the Hankel function of the second kind of zeroth order (Green's function), C is the contour enclosing the object, η_0 is the free-space impedance, k is the free-space wave number, and \vec{r} and \vec{r}' are the observation and source points, respectively.

In 3D scattering problems governed by the vector wave equation, the field radiated by the surface current \vec{J}_s is expressed by

$$\vec{E}(\vec{r}) = -j\omega\mu \oint_S \bar{\bar{\Gamma}}_0(\vec{r}, \vec{r}') \cdot \vec{J}_s(\vec{r}') ds' \quad (5.14)$$

where $\bar{\bar{\Gamma}}_0$ is the free-space dyadic Green's function in closed form and is given by

$$\bar{\bar{\Gamma}}_0(\vec{r}, \vec{r}') = \left[\bar{\bar{I}} + \frac{\nabla\nabla}{k^2} \right] \cdot G_0(\vec{r}, \vec{r}') \quad (5.15)$$

where $G_0(\vec{r}, \vec{r}')$ is the free-space scalar Green's function in closed form and is given by

$$G_0(\vec{r}, \vec{r}') = \frac{\exp[-jk|\vec{r} - \vec{r}'|]}{4\pi|\vec{r} - \vec{r}'|} \quad (5.16)$$

However, in the implementation of the Green's function, if the observation point is in the source region ($\vec{r} \approx \vec{r}'$), the field in (5.13) and (5.14) should be modified in order to avoid the singularity inherent in the function. This is a possible situation if we consider the problem with a single scatterer or the problem with multiple scatterers which are very close to each other. In 2D simulations, the built-in function *besselh* in MATLAB[®] handles this problem using the small argument form of the Hankel function. However, in 3D simulations, this difficulty is solved using the formulation in [98], which excludes an infinitesimal spherical volume V_δ around \vec{r} and modifies (5.14) as follows

$$\vec{E}(\vec{r}) = -j\omega\mu \oint\!\!\!\oint_{S-S_\delta} \vec{\Gamma}_0(\vec{r}, \vec{r}') \cdot \vec{J}_s(\vec{r}') ds' - \frac{\vec{J}_s}{3j\omega\epsilon_0} \quad (5.17)$$

where S_δ is the surface of the volume V_δ . Although this equation provides a proper way to resolve the singularity in some sense, this is not a perfect solution in terms of the numerical performance, as we will discuss in Chapter 8. Therefore, special care must be exercised to calculate the field suitably in the vicinity of the source region.

In order to calculate the RCS at the end of the ILF-DDM algorithm, we also need to calculate the far-zone field radiated by the equivalent currents. The approximation of the field at the far-zone is formulated as follows (in the most general form)

$$\vec{E}_{\text{far}}(\vec{r}) = jk\eta_0 \frac{e^{-jkr}}{4\pi r} \oint\!\!\!\oint_S [\vec{r} \times (\vec{r} \times \vec{J}_s(\vec{r}'))] e^{-jk(\vec{r}' \cdot \vec{r})} dS' \quad (5.18)$$

In 2D applications, using the large argument form of the Hankel function, the expression in (5.18) reduces to

$$E_{z,\text{far}}(\vec{r}) = \frac{-jk\eta_0}{4} \sqrt{\frac{2j}{\pi kr}} e^{-jkr} \oint_C J_z(\vec{r}') \exp[jk(x' \cos \phi + y' \sin \phi)] dl' \quad (5.19)$$

where ϕ is the angle of \vec{r} with respect to the x -axis (i.e., observation angle).

5.2.2. ILF-DDM Algorithm

The initial phase of the algorithm is a suitable partitioning of the original (whole) problem into some suitably-defined subproblems. For the purpose of illustrating the method in the case of a single scatterer, we show the whole problem and the two subproblems in Figure 5.15(a) and Figure 5.15(b-c), respectively, for a dihedral scatterer (or L-shaped cylinder in 2D). Similarly, the whole problem and the three subproblems for a single trihedral scatterer (or U-shaped cylinder in 2D) are shown in Figure 5.16(a) and Figure 5.16(b-c-d), respectively. It is evident that the solution of the whole problems requires a large number of unknowns wasted in the free-space (i.e., white-space) due to the ‘challenging’ geometries of the scatterers. In other words, the *non-convex* geometry of the scatterers in these specific examples requires the employment of the excessive white-space. However, the ILF-DDM need *not* consider the spatial domain of the original problem any more, and also need *not* divide the whole domain of the problem into subdomains by keeping all parts (especially the white space portion) of the original domain, unlike a majority of the domain decomposition techniques in the literature. Instead, it creates small independent subproblems by suitably dividing the surface of the scatterer into smaller subsections and by constructing the free-space and PML regions as shown in the figures. In this partitioning scheme, a small part of the scatterer boundary remains inside the PML region ($\partial\Omega_{S,PML}$) in order to satisfy the smooth continuity of the field at the division regions of the corresponding subproblems. We solve each subproblem in a usual manner using the FEM code, with the single exception that the applied incident field decays smoothly along the scatterer boundary which is inside the PML region, due to the PML implementation along these boundaries. It is worthwhile to mention that we do not consider these scatterer boundaries inside the PML region while calculating and radiating the equivalent currents residing on these boundaries. These portions can be interpreted as ‘buffer’ regions in order to solve each subproblem

using the FEM without distorting the expected field behaviour inside the subproblem.

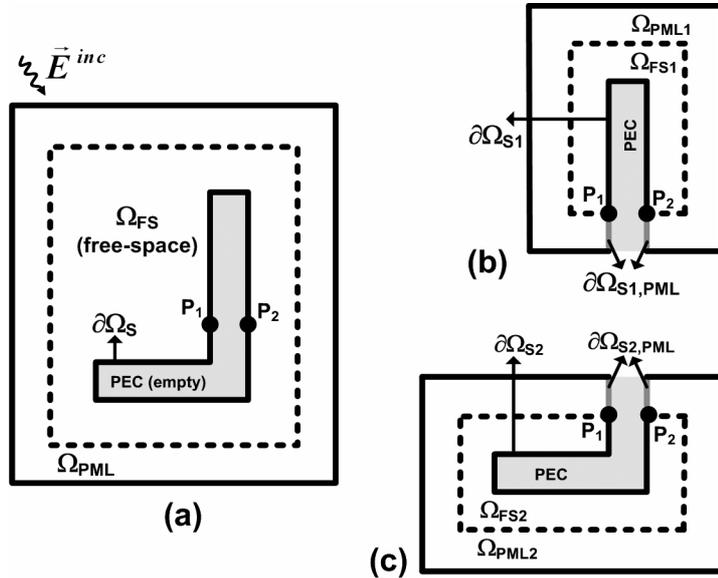


Figure 5.15. ILF-DDM for dihedral: (a) Whole problem, (b-c) Two subproblems.

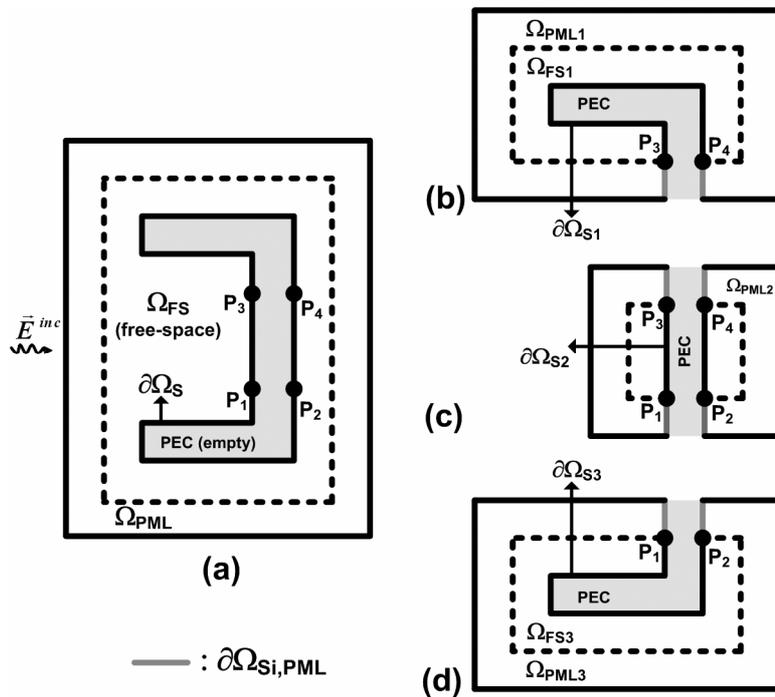


Figure 5.16. ILF-DDM for trihedral: (a) Whole problem, (b-c-d) Three subproblems.

The ILF-DDM algorithm can obviously be applied for the case of multiple scatterers in a straightforward and simpler manner. For the sake of demonstration, we show the whole problem and the corresponding subproblems in Figure 5.17(a) and Figure 5.17(b), respectively. In this case, each subproblem is constructed independently over the boundary of each scatterer. Since the scatterers are distinct, the locally-conformal PML technique is applied only for the usual mesh truncation of each subproblem.

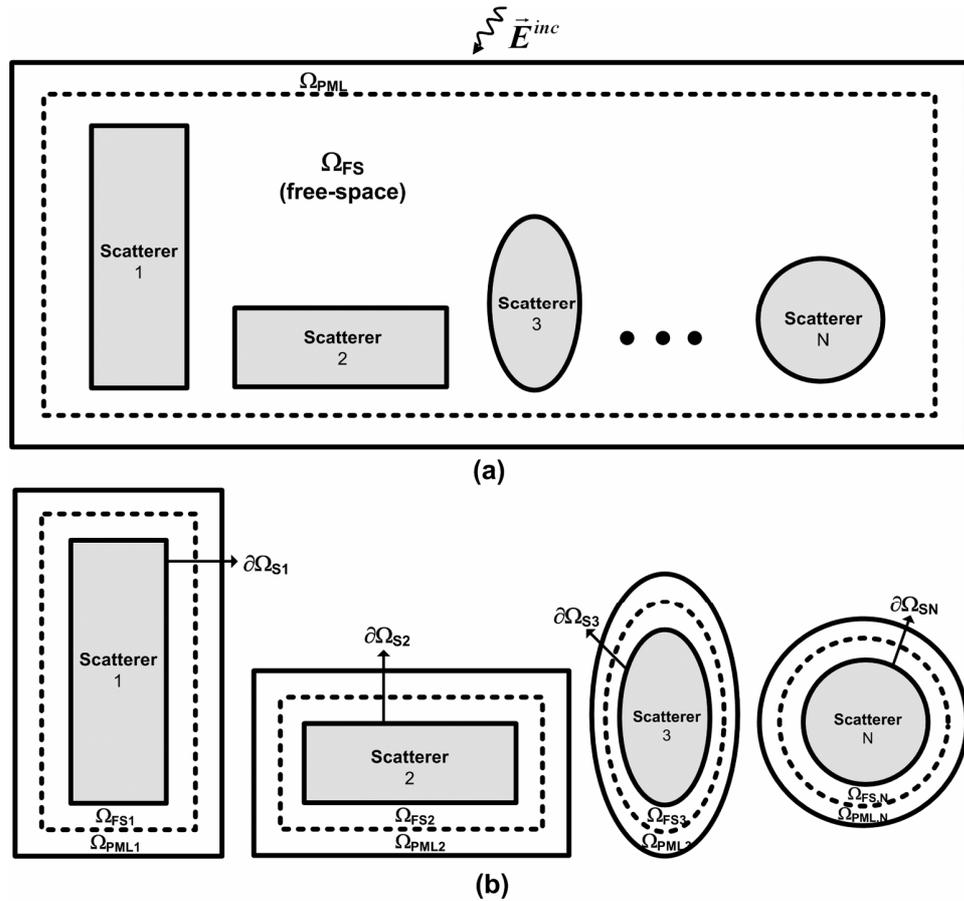


Figure 5.17. ILF-DDM for multiple objects: (a) Whole problem, (b) Subproblems.

After the proper construction of the subproblems, the ILF-DDM algorithm is implemented iteratively as shown in the flowchart in Figure 5.18. In the first iteration, we solve each subproblem independently using the incident plane wave

in order to calculate the scattered field inside the corresponding subdomain. More specifically, in 2D, we solve the following BVP governed by the Helmholtz equation with a Dirichlet type BC (for the i^{th} subproblem)

Inside the free-space region:

$$\nabla^2 E_z^s(\vec{r}) + k^2 E_z^s(\vec{r}) = 0 \quad \text{in } \Omega_{\text{FS},i} \quad (5.20.a)$$

$$\text{with BC: } E_z^s(\vec{r}) = -E_z^{\text{inc}}(\vec{r}) \quad \text{on } \partial\Omega_{\text{Si}} \quad (5.20.b)$$

Inside the PML region:

$$\tilde{\nabla}^2 E_z^{\text{sc}}(\tilde{\vec{r}}) + k^2 E_z^{\text{sc}}(\tilde{\vec{r}}) = 0 \quad \text{in } \Omega_{\text{PML},i} \quad (5.20.c)$$

$$\text{with BC: } E_z^{\text{sc}}(\tilde{\vec{r}}) = -E_z^{\text{inc}}(\tilde{\vec{r}}) \quad \text{on } \partial\Omega_{\text{Si,PML}} \quad (5.20.d)$$

where $E_z^{\text{inc}}(\vec{r})$ and $E_z^{\text{inc}}(\tilde{\vec{r}})$ are the incident plane wave functions evaluated in real and complex coordinates, respectively, $E_z^s(\vec{r})$ and $E_z^s(\tilde{\vec{r}})$ are the scattered fields in real and complex coordinates, respectively, $\tilde{\nabla}$ is the nabla operator in complex space, and the superscript ‘c’ represents the analytical continuation to complex space. Similarly, in 3D, we solve the following BVP governed by the vector wave equation (for the i^{th} subproblem)

Inside the free-space region:

$$\nabla \times \nabla \times \vec{E}^s(\vec{r}) - k^2 \vec{E}^s(\vec{r}) = 0 \quad \text{in } \Omega_{\text{FS},i} \quad (5.21.a)$$

$$\text{with BC: } \hat{n} \times \vec{E}^s(\vec{r}) = -\hat{n} \times \vec{E}^{\text{inc}}(\vec{r}) \quad \text{on } \partial\Omega_{\text{Si}} \quad (5.21.b)$$

Inside the PML region:

$$\tilde{\nabla} \times \tilde{\nabla} \times \vec{E}^{\text{sc}}(\tilde{\vec{r}}) - k^2 \vec{E}^{\text{sc}}(\tilde{\vec{r}}) = 0 \quad \text{in } \Omega_{\text{PML},i} \quad (5.21.c)$$

$$\text{with BC: } \hat{n} \times \vec{E}^{\text{sc}}(\tilde{\vec{r}}) = -\hat{n} \times \vec{E}^{\text{inc}}(\tilde{\vec{r}}) \quad \text{on } \partial\Omega_{\text{Si,PML}} \quad (5.21.d)$$

where $\vec{E}^{\text{inc}}(\vec{r})$ and $\vec{E}^{\text{inc}}(\tilde{\vec{r}})$ are the incident plane wave functions evaluated in real and complex coordinates, respectively, $\vec{E}^s(\vec{r})$ and $\vec{E}^{\text{sc}}(\tilde{\vec{r}})$ are the scattered fields in real and complex coordinates, respectively. In the case of multiple

scatterers, we solve the same wave equations, except that we omit the BC expressions in (5.20.d) and (5.21.d). In other words, the problem turns out to be a usual scattering problem which considers a single PEC scatterer.

In the first iteration, after solving each subproblem, we calculate the equivalent currents over the corresponding scatterer boundary $\partial\Omega_{s_i}$ (excluding $\partial\Omega_{s_i, PML}$) using (5.12.a). Namely, we calculate \vec{J}_i^1 where the superscript and the subscript denote the iteration number and the ID of the subproblem, respectively ($i=1, 2, \dots, N$ and N is the number of subproblems). In this way, we complete the first iteration.

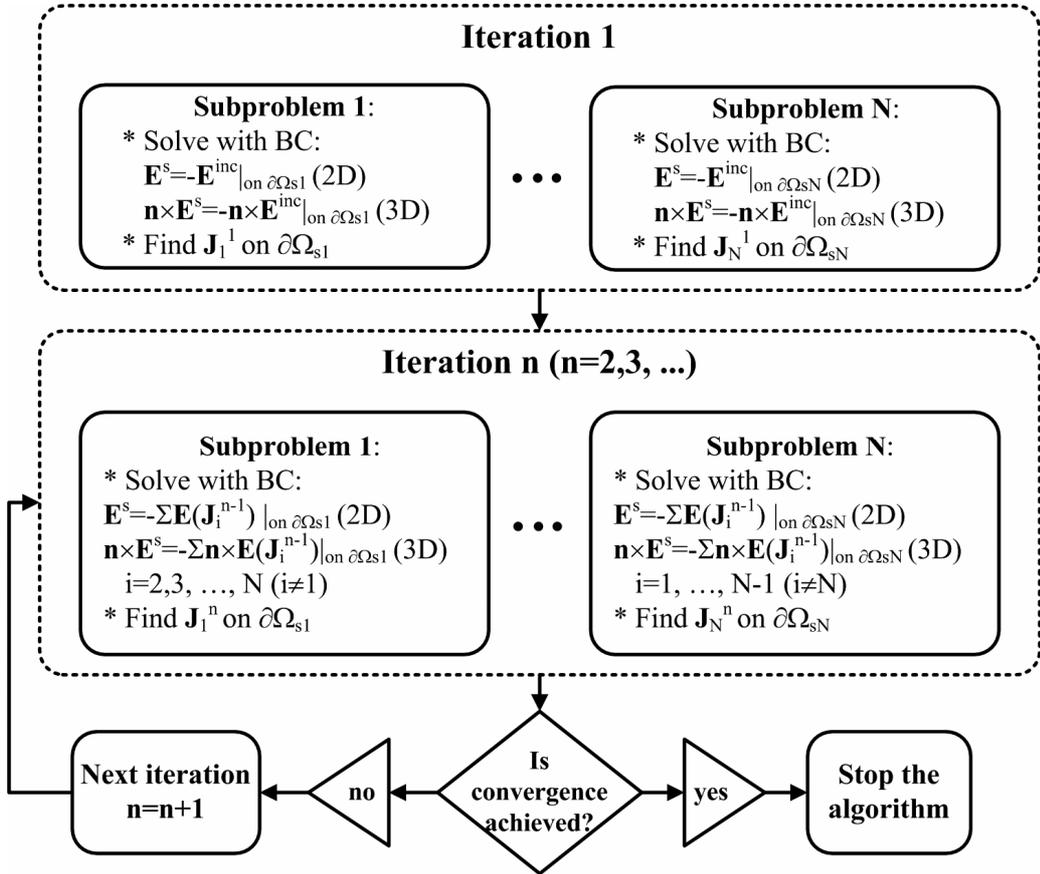


Figure 5.18. Flowchart of the ILF-DDM algorithm.

After the first iteration, the procedure for the subsequent iterations are similar to each other. In the n^{th} iteration, we solve the i^{th} subproblem using the ‘new incident fields’, which are calculated using the equivalent currents radiated from the surface of all other scatterers (i.e., \vec{J}_j^{n-1} where $j=1, 2, \dots, N$ and $j \neq i$) by means of the radiation integrals in Section 5.2.1. To be more precise, in the n^{th} iteration, we solve the i^{th} subproblem by modifying the BC in (5.20) in 2D as follows

New BC inside the free-space region:

$$E_z^s(\vec{r}) = -\sum_{j=1}^N \left\{ E_z(\vec{r}) \text{ due to } \vec{J}_j^{n-1} \right\} \quad (j \neq i) \quad \text{on } \partial\Omega_{\text{Si}} \quad (5.22.a)$$

New BC inside the PML region:

$$E_z^{sc}(\tilde{\vec{r}}) = -\sum_{j=1}^N \left\{ E_z(\vec{r}) \text{ due to } \vec{J}_j^{n-1} \right\} \cdot g(\vec{r}) \quad (j \neq i) \quad \text{on } \partial\Omega_{\text{Si,PML}} \quad (5.22.b)$$

Similarly, we modify the BC in (5.21) in 3D as follows

New BC inside the free-space region:

$$\hat{n} \times \vec{E}^s(\vec{r}) = -\sum_{j=1}^N \left\{ \hat{n} \times \vec{E}(\vec{r}) \text{ due to } \vec{J}_j^{n-1} \right\} \quad (j \neq i) \quad \text{on } \partial\Omega_{\text{Si}} \quad (5.23.a)$$

New BC inside the PML region:

$$\hat{n} \times \vec{E}^{sc}(\tilde{\vec{r}}) = -\sum_{j=1}^N \left\{ \hat{n} \times \vec{E}(\vec{r}) \text{ due to } \vec{J}_j^{n-1} \right\} \cdot g(\vec{r}) \quad (j \neq i) \quad \text{on } \partial\Omega_{\text{Si,PML}} \quad (5.23.b)$$

In (5.22.b) and (5.23.b), we prefer to use $\vec{E}(\vec{r})$ (rather than $\vec{E}(\tilde{\vec{r}})$) since it is not feasible to use complex coordinates in the radiation integrals. Instead, we just multiply the field by a monotonically decreasing exponential function given by

$$g(\vec{r}) = \exp[-f(\xi)] \quad (5.24)$$

where $f(\xi)$ is the monotonically increasing function of ξ , given in (4.7) in Chapter 4. The expression in (5.24) is equivalent to the PML action providing an ‘artificial decay’ along $\partial\Omega_{\text{Si,PML}}$. This field decay is still away from the PML-

free space interface due to the nature of the complex coordinate transformation. Then, after solving all subproblems using the above procedure, we calculate the equivalent currents \vec{J}_i^n ($i=1, 2, \dots, N$) for the n^{th} iteration using (5.12.a).

The cycle of iterations is processed until convergence is achieved. The iterations are terminated when the magnitude of the equivalent current over each scatterer boundary is sufficiently small. This is due to the fact that the magnitude of the induced current decreases monotonically (i.e., $\|\vec{J}_i^n\| < \|\vec{J}_i^{n-1}\|$), or in other words, the total induced current converges to a limit. At the end of the algorithm, we calculate the far-zone fields by radiating the total currents (i.e., the sum of currents for all iterations and for all subproblems) using (5.18) in 3D and using (5.19) in 2D.

CHAPTER 6

NUMERICAL EXPERIMENTS (PML METHODS)

In this chapter, we first illustrate how we have empirically determined the threshold value (45°) in the multi-center PML method by considering the problem of 2D scattering by an infinitely-long PEC cylinder of square cross-section. Then, we report the results of some numerical experiments to test the accuracy of the locally-conformal PML and the multi-center PML methods in 2D and 3D radiation/scattering problems. The 2D and 3D simulations are performed using the softwares FEMS^{2D} and FEMS^{3D}, respectively.

In order to measure the performance of the methods, we introduce two different kinds of mean-square (MS) error ('energy error') criteria. The first error criterion is defined as

$$E_1 = \frac{\sum_{\Omega} |\vec{E}^{\text{MC}} - \vec{E}^{\text{LC}}|^2}{\sum_{\Omega} |\vec{E}^{\text{LC}}|^2} \quad (\times 100) \quad (6.1)$$

where \vec{E}^{MC} and \vec{E}^{LC} are the fields calculated in Ω (free-space) by the multi-center PML method and the locally-conformal PML method, respectively. This error criterion compares the two PML methods. The second error criterion is defined as

$$E_2 = \frac{\sum_{\Omega} |\vec{E} - \vec{E}^{\text{analytic}}|^2}{\sum_{\Omega} |\vec{E}^{\text{analytic}}|^2} \quad (\times 100) \quad (6.2)$$

where \vec{E} is the field calculated by either the locally-conformal PML method or the multi-center PML method, and $\vec{E}^{\text{analytic}}$ is the field calculated analytically in Ω (free-space). The second error criterion is applicable only when the analytical solution of the problem is available.

The performances of the PML methods in scattering problems are also tested in terms of the RCS calculations of the objects, and are validated with the results of; (i) a standard MoM code, (ii) a standard PO code, (iii) analytical calculations, or (iv) literature, whichever applicable. In 2D simulations involving scattering problems, we have developed a MoM code in MATLAB[®]. In 3D simulations involving the problem of scattering from rectangular plates, we have developed a MoM code in the C language by converting and adapting the FORTRAN routines in [81], and a PO code in MATLAB[®] using the derivations in [82]. In addition, for the 3D scattering from circular plates, we have used the PO code in [83] using MATLAB[®]. However, we should mention that the PO method is only reliable for aspect angles $\theta \leq 20^\circ$ [83]. Finally, for the analytical solution of the scattering from circular cylinder and sphere, we have developed codes in MATLAB[®] using the Mie Series expressions in [84]. While calculating this series, we have summed at least *two* ka terms (k is the wavenumber and a is the radius) to represent the sum within a few percent accuracy [85].

6.1. Determination of the Threshold Angle Criterion in the Multi-Center PML Method

In the multi-center PML method in Section 4.2, we have mentioned that the centers should be selected in such a way that each PML point should have at least one center whose angle θ_i is less than or equal to 45° . We have determined this threshold empirically after observing many numerical simulations. In this section, we illustrate how we have designated “45°” as the threshold value. For

this purpose, we consider the problem of 2D scattering of an incident plane wave $\vec{E}^{inc} = \hat{a}_z \exp(-jkx)$ by an infinitely-long PEC cylinder of square cross-section whose edge length is 10λ . We first construct a test problem as shown in Figure 6.1, which shows only the PML region (Ω_{PML}) which is conformal to the surface of the square PEC cylinder, for the sake of simplification. The thickness of Ω_{PML} is chosen as $\lambda/4$. We note that the region Ω is only the free-space region between the scatterer boundary and the boundary $\partial\Omega$. We slightly modify the multi-center PML approach described in Section 4.2 as follows: First of all, we take four centers ($C_i, i=1,2,3,4$), and we assign these centers to *only* their corresponding edges. In other words, if an arbitrary PML point is close to the Edge- i , then the weight of the center C_i (i.e., w_i) is set to 1, while all other weights are equated to zero by brute-force. In this way, we isolate the effect of the other centers on the PML point close to the Edge- i , and thus, we can understand the range of values for the angle θ_i in order to get accurate results. In this figure, we consider the unit vector a_i at the end point of Edge- i because at this point a_i makes the “worst” angle $\theta_{i,max}$ with the unit normal vector a_n , in the sense that the value of the angle is maximized. Thus, the “worst” angle $\theta_{i,max}$ is actually the “maximum” angle for this configuration. We should also emphasize that for the PML points at the corner regions of the PML region, we employ the superposition principle (only for this test example), that is, for example at the corner region where the Edge-1 and Edge-2 intersect, we set the weights of both C_1 and C_2 to 0.5 equally. In this figure, the center C_i is placed at a distance d_c , where $d_c = \beta d$ with $\beta > 0$. Then, we have observed that as we increase the value of d_c , the analysis results have started to become accurate after some critical distance is attained. That is, after d_c becomes equal to d (i.e., $\beta=1$ and the “worst” angle $\theta_{i,max}=45^\circ$), there occurs an abrupt change in the accuracy of the results. When we increase the value of d_c further (i.e., $\beta > 1$, and $\theta_{i,max} < 45^\circ$), the error remains at a constant level, which is almost zero. Thus, we have concluded that the 45° is a good candidate for the threshold angle. When

$\theta_{i,\max}$ is 45° , which means that $\beta=1$ and $d_c=d$, all four centers are located at the same position (i.e., center-of-mass point) of the square cross-section of the cylinder. Therefore, this example with $\beta=1$ reduces to the multi-center PML implementation with “single center” selected at the center-of-mass point of the square. In Figure 6.2, we plot the error value E_1 as a function of the “worst” angle $\theta_{i,\max}$, and we show the threshold point on this plot by the black dot.

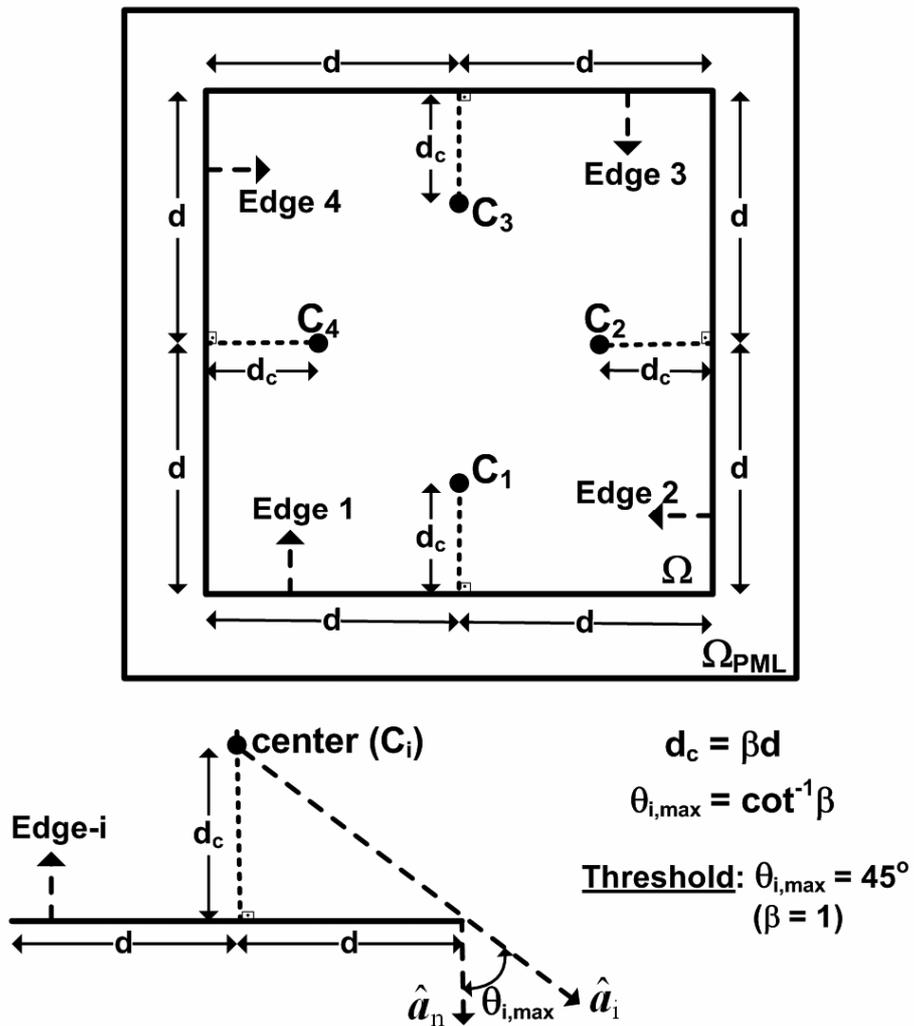


Figure 6.1. Illustration of how to determine the threshold angle criterion via a problem of scattering by an infinitely-long PEC cylinder of square cross-section.

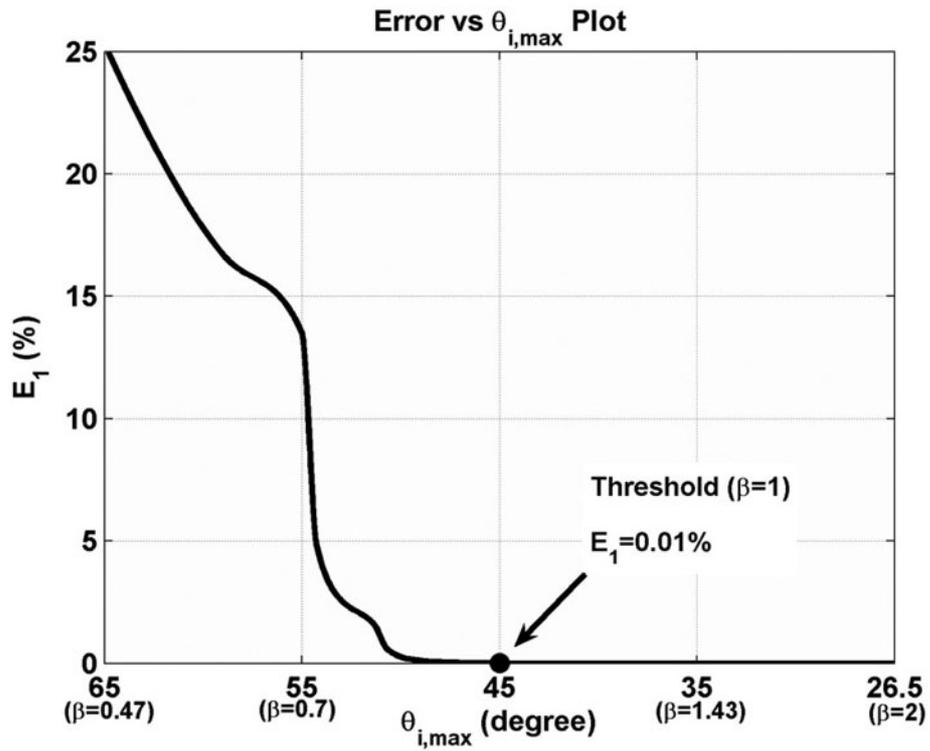


Figure 6.2. Determination of the threshold angle using the error vs $\theta_{i,\max}$ plot.

6.2. 2D Simulations

In this section, we test the locally-conformal PML and the multi-center PML techniques using infinitely-long PEC cylinders of various cross-sections. We report the results of some numerical experiments for the two problems:

- (i) scattering from an infinitely-long cylinder of arbitrary cross-section (see Section 2.1.1 and Figure 2.1)
- (ii) radiation of a single line-source inside an infinitely-long cylindrical region of arbitrary shape (see Section 2.1.2 and Figure 2.2).

In the simulation phase of the radiation problem, we resort to the “Monte Carlo” simulation technique in order to get more reliable and robust results, because the accuracy of the method may depend on the position of the line-source inside Ω_{LS}

(see Figure 2.2). In the Monte Carlo simulation technique, which is a stochastic technique based on the use of random numbers and probability statistics to investigate the problems, we determine randomly 2000 different source positions inside Ω_{LS} , and we run the FEMS^{2D} software 2000 times using these source positions and calculate the error values for each run using (6.2) in Ω_{FS} . Then, using these error values, (i) we plot the error scatter contour which shows the error value at each source position, (ii) we plot the error histogram which shows the distribution of the error values, and (iii) we calculate the error statistics (mean, variance, etc.). This process provides a global way for the numerical analysis of the locally-conformal method in the solution of the radiation problem by yielding more robust (i.e. independent of source position) analysis results.

In the following, the ‘items with bullets’ are categorized with respect to the mesh structure of the computational domain, because, in each subsection, the same mesh structure is employed in the solution of both the radiation and the scattering problems, but with different interpretations. That is, in the radiation problem, the mesh of the whole computational domain Ω_C , which is of the cylindrical shell geometry, is assumed to be constructed conformally over a region Ω_{LS} having a single line-source inside (see Figure 2.2). However, in the scattering problem, the mesh of Ω_C is assumed to be constructed conformally over a perfect electric conductor (PEC) scatterer (see Figure 2.1). Therefore, in each subsection, we first report the results of the scattering problem for a given geometry. Then, we demonstrate the results of the radiation problem for the same geometry.

The common parameters in all experiments are chosen as: k is 20π (i.e., the wavelength λ is 0.1 meter), m is 3, and α is chosen in the range between $7k$ - $10k$. In addition, the PML thickness is approximately set to $\lambda/4$, and the edge size of each triangular element in the mesh is approximately adjusted to the range between $\lambda/40$ - $\lambda/80$, depending on the size of the problem. Moreover, for the

scattering problem, we assume that the angle of incidence of the plane wave is 180° (with respect to the x -axis), unless otherwise stated. In all experiments, almost the same approach is followed to present the results (i.e., the order and the format of the plots, etc.) for the sake of uniformity.

- **Conformal PML over a circular domain:** In this part, the free-space region Ω is designed conformally over a circular domain with radius 2λ , and the PML region Ω_{PML} is constructed as conformal to Ω . The mesh structure, which is just a coarse mesh given for the purpose of visualization, is shown in Figure 6.3. In this figure, the red dot indicates the center position chosen in the multi-center PML implementation. It is evident that the multi-center PML method with a single center at the center-of-mass point of the circular domain is equivalent to the locally-conformal PML method.

For the problem of scattering from an infinitely-long PEC cylinder of circular cross section whose radius is 2λ , we plot the bistatic RCS profile and the magnitude of the current density along the scatterer boundary in Figure 6.4 and Figure 6.5, respectively. We also plot the contours of the scattered field and the total field in Figure 6.6(a) and (b), respectively, for the locally-conformal PML implementation; and of those in Figure 6.6(c) and (d), respectively, for the multi-center PML implementation. Finally, we plot the magnitude and phase of the scattered field along the x -axis as shown in Figure 6.7. The magnitude plot in Figure 6.7(a) proves the efficiency of the PML implementations, such that the magnitude of the function decays smoothly inside the PML region Ω_{PML} . We also calculate the errors; E_1 as $6.5629 \times 10^{-8} \%$, E_2 as $1.0572 \times 10^{-3} \%$ for the locally-conformal PML, and E_2 as $1.1094 \times 10^{-3} \%$ for the multi-center PML. Although the multi-center PML is equivalent to the locally-conformal PML in this example, the tiny discrepancies between the results are due to the differences in the numerical implementation of the methods.

For the radiation problem, we utilize the Monte Carlo technique by selecting randomly 2000 source positions inside Ω_{LS} . We plot the error scatter plot in Figure 6.8(a), and the error histogram in (b) for E_2 which is calculated by the locally-conformal PML method. Similarly, we plot the error scatter plot in Figure 6.8(c), and the error histogram in (d) for E_2 which is calculated by the multi-center PML method. We also show some statistical error values (i.e., mean, variance, etc.) on the plots in Figure 6.8(b) and (d). Then, we assume that a single line-source is located at the origin, and we calculate the radiated field inside the free-space. We plot the magnitude and phase of the radiated field along the x -axis as shown in Figure 6.9.

In addition, we consider the problem of scattering from a ‘dielectric’ cylinder of circular cross section whose radius is 2λ and relative permittivity is 2. The mesh structure and the bistatic RCS profile are shown in Figure 6.10 and Figure 6.11, respectively. We also plot the contours of the scattered field and the total field in Figure 6.12(a) and (b), respectively, for the locally-conformal PML implementation; and of those in Figure 6.12(c) and (d), respectively, for the multi-center PML implementation. Finally, we plot the magnitude and phase of the total field inside the cylinder along the x -axis as shown in Figure 6.13. We calculate the errors; E_1 as 1.4987×10^{-9} %, E_2 as 2.3355×10^{-2} % for the locally-conformal PML, and E_2 as 2.3550×10^{-2} % for the multi-center PML.

- **Conformal PML over a square domain:** In this part, a conformal PML is designed over a square domain whose edge length is 10λ , as shown in Figure 6.14. A single center is selected at the center-of-mass point of the square domain in the multi-center PML implementation.

For the problem of scattering from an infinitely-long PEC cylinder of square cross section, we plot the bistatic RCS profile and the magnitude of the current density along the scatterer boundary in Figure 6.15 and Figure 6.16, respectively. We also plot the contours of the scattered field and the total field in Figure 6.17(a) and (b), respectively, for the locally-conformal PML implementation; and of those in Figure 6.17(c) and (d), respectively, for the multi-center PML method. We calculate the error E_1 as $9.4986 \times 10^{-3} \%$.

For the Monte Carlo simulation of the radiation problem, we plot the error scatter plot in Figure 6.18(a), and the error histogram in (b) for E_2 which is calculated by the locally-conformal PML method. Similarly, we plot the error scatter plot in Figure 6.18(c), and the error histogram in (d) for E_2 which is calculated by the multi-center PML method.

- **Conformal PML over a rectangular domain:** In this part, a conformal PML is designed over an elongated $4\lambda \times 16\lambda$ rectangular domain, as shown in Figure 6.19. Four centers are selected in the multi-center PML implementation.

For the scattering problem, we plot the bistatic RCS profile and the magnitude of the current density along the scatterer boundary in Figure 6.20 and Figure 6.21, respectively. We also plot the contours of the scattered field and the total field in Figure 6.22(a) and (b), respectively, for the locally-conformal PML implementation; and of those in Figure 6.22(c) and (d), respectively, for the multi-center PML implementation. We calculate the error E_1 as $3.3020 \times 10^{-3} \%$. In addition, for some cases having various center positions, we plot the scattered field contours in Figure 6.23. In this figure, the ‘black ellipses’ indicate the regions where the 45° criterion is not satisfied. We also tabulate the error values E_1 in Table 6.1. In this table, Case 1 is the original case where we employ four centers shown in Figure 6.23(a).

This case is actually the best case, and shows that only four centers are sufficient to achieve acceptable results for this geometry. Case 2 employs only two centers C_1 and C_4 (see Figure 6.23(b)), Case 3 has only two centers C_2 and C_3 (see Figure 6.23(c)), and finally Case 4 has only the center-of-mass point of the rectangle (i.e., origin) (see Figure 6.23(d)). It should be evident that Cases 2-3 and 4 do not satisfy the 45° criterion for all PML points, and they fail to achieve accurate results. Case 2 is more successful than the Cases 3 and 4 although the angle threshold criterion is not satisfied for all PML points. The corners of the rectangle have a considerable effect on the accuracy of the results, and since all PML points close to the corners of the rectangle satisfy at least the threshold criterion in Case 2, the error value turns out to be less compared to the Cases 3 and 4.

Table 6.1. Error values for different center positions in the problem of scattering by an infinitely-long PEC cylinder of rectangular cross-section.

Case	Centers	E_1 (%)
(1)	C_1, C_2, C_3, C_4	3.3020×10^{-3}
(2)	C_1, C_4	0.3452
(3)	C_2, C_3	36.7823
(4)	Center-of-mass (origin)	69.4829

For the Monte Carlo simulation of the radiation problem, we plot the error scatter plot in Figure 6.24(a), and the error histogram in (b) for E_2 which is calculated by the locally-conformal PML method. Similarly, we plot the error scatter plot in Figure 6.24(c), and the error histogram in (d) for E_2 which is calculated by the multi-center PML method.

- **Conformal PML over a half-circular domain:** This part demonstrates the PML implementations over a half-circular domain. The radius of the inner boundary of the half-circular shell is 2λ , and the mesh structure is shown in Figure 6.25.

For the scattering problem, we plot the bistatic RCS profile and the magnitude of the current density along the scatterer boundary in Figure 6.26 and Figure 6.27, respectively. We also plot the contours of the scattered field and the total field in Figure 6.28 for both PML methods. We calculate the error E_1 as $1.4233 \times 10^{-5} \%$.

For the Monte Carlo simulation of the radiation problem, we plot the error scatter plots and the error histogram for E_2 in Figure 6.29 for both PML implementations.

- **Conformal PML over an elliptical domain:** In this part, we consider the conformal PML over an elliptical domain. As shown in Figure 6.30, the semi-major axis of the inner boundary of the elliptical shell is 2λ , and the axial ratio is 2.

For the scattering problem, we plot the bistatic RCS profile and the magnitude of the current density along the scatterer boundary in Figure 6.31 and Figure 6.32, respectively. We also plot the contours of the scattered field and the total field in Figure 6.33 for both PML methods. We calculate the error E_1 as $7.0296 \times 10^{-4} \%$.

For the Monte Carlo simulation of the radiation problem, we plot the error scatter plots and the error histogram for E_2 in Figure 6.34 for both PML implementations.

- **Conformal PML over an ogive domain:** The next example is a conformal PML over an ogive domain. As shown in the mesh structure in Figure 6.35, the maximum diameter of the inner boundary of the ogive shell is 2λ , and the nose angle is 75° . Since the nose angle is less than 90° , we slightly cut the PML region in order to satisfy the threshold criterion in the multi-center PML method.

For the scattering problem, we plot the bistatic RCS profile and the magnitude of the current density along the scatterer boundary in Figure 6.36 and Figure 6.37, respectively. We also plot the contours of the scattered field and the total field in Figure 6.38 for both PML methods. We calculate the error E_1 as $6.6522 \times 10^{-5} \%$.

For the Monte Carlo simulation of the radiation problem, we plot the error scatter plots and the error histogram for E_2 in Figure 6.39 for both PML implementations.

- **Conformal PML over a triangular+halfcircular domain:** In this example, a conformal PML is designed over a triangular+halfcircular domain, as shown in Figure 6.40. The radius of the inner boundary of the shell is 1.6λ , and the triangle (nose) angle is 90° .

For the scattering problem, we plot the bistatic RCS profile and the magnitude of the current density along the scatterer boundary in Figure 6.41 and Figure 6.42, respectively. The angle of incidence of the plane wave is set to 90° (nose-on incidence). We also plot the contours of the scattered field and the total field in Figure 6.43 for both PML methods. We calculate the error E_1 as $3.3869 \times 10^{-4} \%$.

For the Monte Carlo simulation of the radiation problem, we plot the error scatter plots and the error histogram for E_2 in Figure 6.44 for both PML implementations.

- **Conformal PML over a ‘sharp’ triangular+halfcircular domain:** In this example, a conformal PML is designed over a ‘sharp’ triangular+halfcircular domain, as shown in Figure 6.45. The radius of the inner boundary of the shell is 0.5λ , and the triangle (nose) angle is 20° . Since the nose angle is less than 90° , we slightly cut the PML region in order to satisfy the threshold criterion in the multi-center PML method.

For the scattering problem, we plot the bistatic RCS profile and the magnitude of the current density along the scatterer boundary in Figure 6.46 and Figure 6.47, respectively. We also plot the contours of the scattered field and the total field in Figure 6.48 for both PML methods. We calculate the error E_1 as $6.6370 \times 10^{-4} \%$.

For the Monte Carlo simulation of the radiation problem, we plot the error scatter plots and the error histogram for E_2 in Figure 6.49 for both PML implementations.

- **Conformal PML over a quadrilateral domain:** The next example considers a conformal PML designed over a quadrilateral domain as shown in Figure 6.50. The vertex coordinates of the inner boundary of the shell are (in terms of λ): vertex 1: (-2, 2), vertex 2: (-4, -2), vertex 3: (4, -4) and vertex 4: (10, 4), in Cartesian coordinates. At two vertices, we slightly cut the PML region in order to satisfy the threshold criterion in the multi-center PML method.

For the scattering problem, we plot the bistatic RCS profile and the magnitude of the current density along the scatterer boundary in Figure 6.51 and Figure 6.52, respectively. We also plot the contours of the scattered field and the total field in Figure 6.53 for both PML methods. We calculate the error E_1 as $1.5382 \times 10^{-2} \%$.

For the Monte Carlo simulation of the radiation problem, we plot the error scatter plots and the error histogram for E_2 in Figure 6.54 for both PML implementations.

- **Conformal PML over a pentagonal domain:** The next example is a conformal PML designed over a pentagonal domain (see Figure 6.55). The vertex coordinates of the inner boundary of the shell are (in terms of λ): vertex 1: (-2, 2), vertex 2: (-4, 0), vertex 3: (0, -2), vertex 4: (4, 0) and vertex 5: (2, 2), in Cartesian coordinates. At two vertices, we slightly cut the PML region in order to satisfy the threshold criterion in the multi-center PML method.

For the scattering problem, we plot the bistatic RCS profile and the magnitude of the current density along the scatterer boundary in Figure 6.56 and Figure 6.57, respectively. We also plot the contours of the scattered field and the total field in Figure 6.58 for both PML methods. We calculate the error E_1 as $6.5165 \times 10^{-3} \%$.

For the Monte Carlo simulation of the radiation problem, we plot the error scatter plots and the error histogram for E_2 in Figure 6.59 for both PML implementations.

- **Conformal PML over a hexagonal domain:** The last example is a conformal PML designed over a hexagonal domain (see Figure 6.60). The

vertex coordinates of the inner boundary of the shell are (in terms of λ): vertex 1: (2, 0), vertex 2: (1, 2), vertex 3: (-1, 2), vertex 4: (-2, 0), vertex 5: (-1, -2) and vertex 6: (1, -2), in Cartesian coordinates.

For the scattering problem, we plot the bistatic RCS profile and the magnitude of the current density along the scatterer boundary in Figure 6.61 and Figure 6.62, respectively. We also plot the contours of the scattered field and the total field in Figure 6.63 for both PML methods. We calculate the error E_1 as $3.5348 \times 10^{-5} \%$.

For the Monte Carlo simulation of the radiation problem, we plot the error scatter plots and the error histogram for E_2 in Figure 6.64 for both PML implementations.

The numerical experiments in this section demonstrate that the results of the locally-conformal PML method are sufficiently close to those of the multi-center PML method, as well as to the reference results. Thus, these examples prove that the locally-conformal PML and the multi-center PML methods are efficient absorbers for the FEM mesh truncation having very challenging geometries. We also conclude from the error scatter plots that the error value E_2 is at an acceptable level irrespective of the position of the line-source. However, the error value gets slightly higher near to the boundary $\partial\Omega_{LS}$, especially at the sharper edges of some geometries (such as quadrilateral, ogive). This observation is not surprising, since the solution of Helmholtz equation in (2.10) becomes singular at the source location. In addition, it is known that the sharp edges may have some effect on the accuracy of the results, compared to the smooth sections of the geometry.

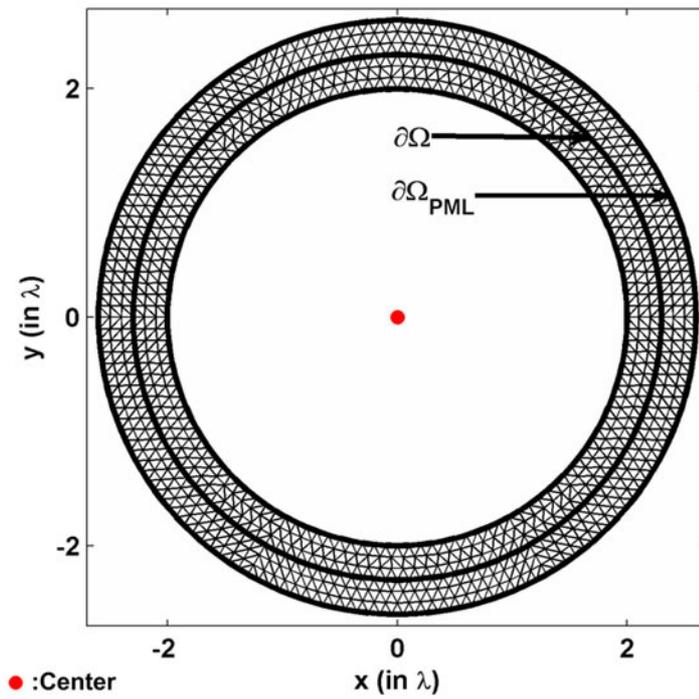


Figure 6.3. Mesh for circular domain (radius is 2λ).

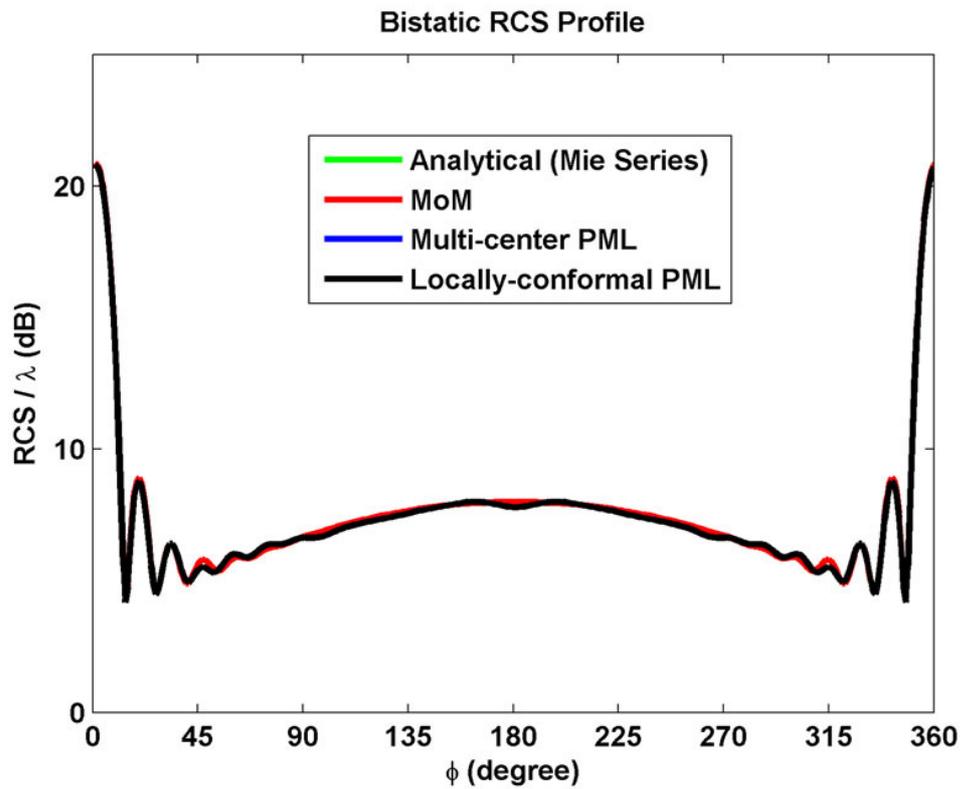


Figure 6.4. RCS profile for scattering from circular PEC cylinder.

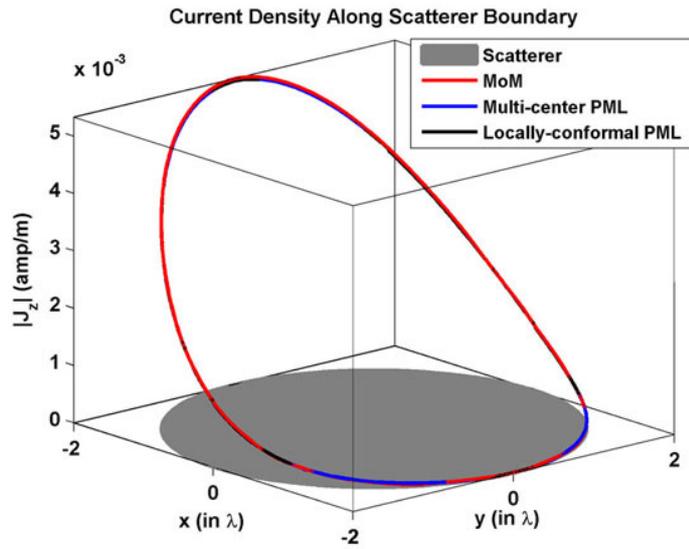


Figure 6.5. Magnitude of current density along circular PEC cylinder.

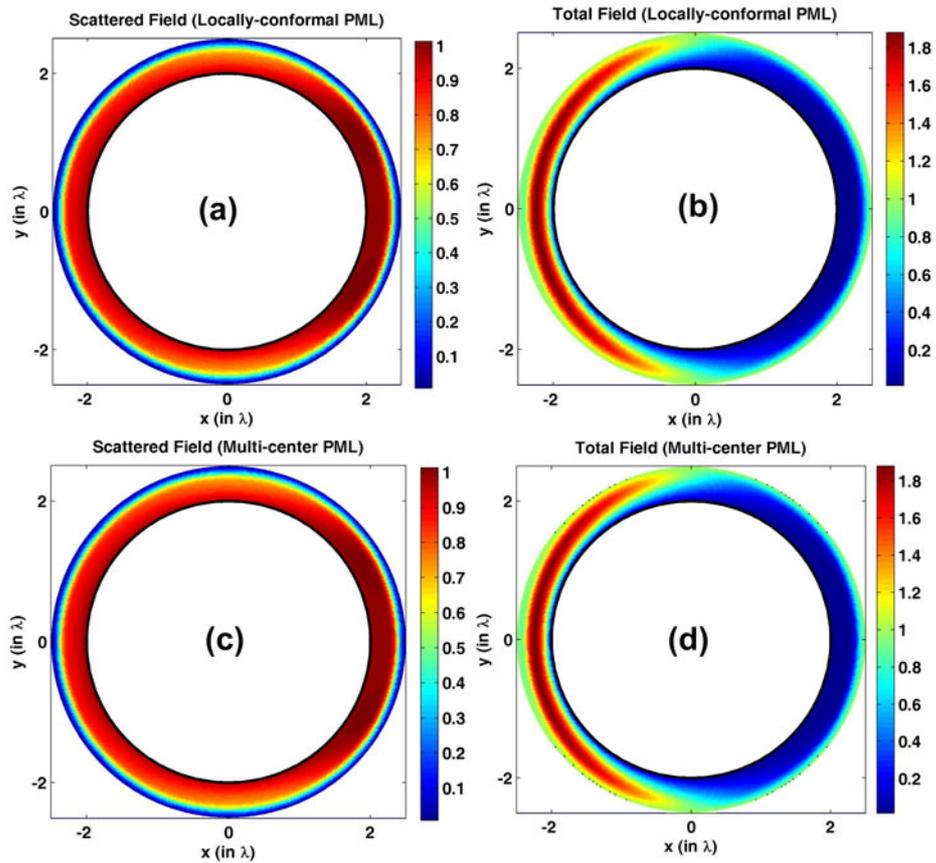


Figure 6.6. Field contours for PEC circular cylinder (magnitude): (a-b) Scattered and total fields in the locally-conformal PML, (c-d) Scattered and total fields in the multi-center PML.

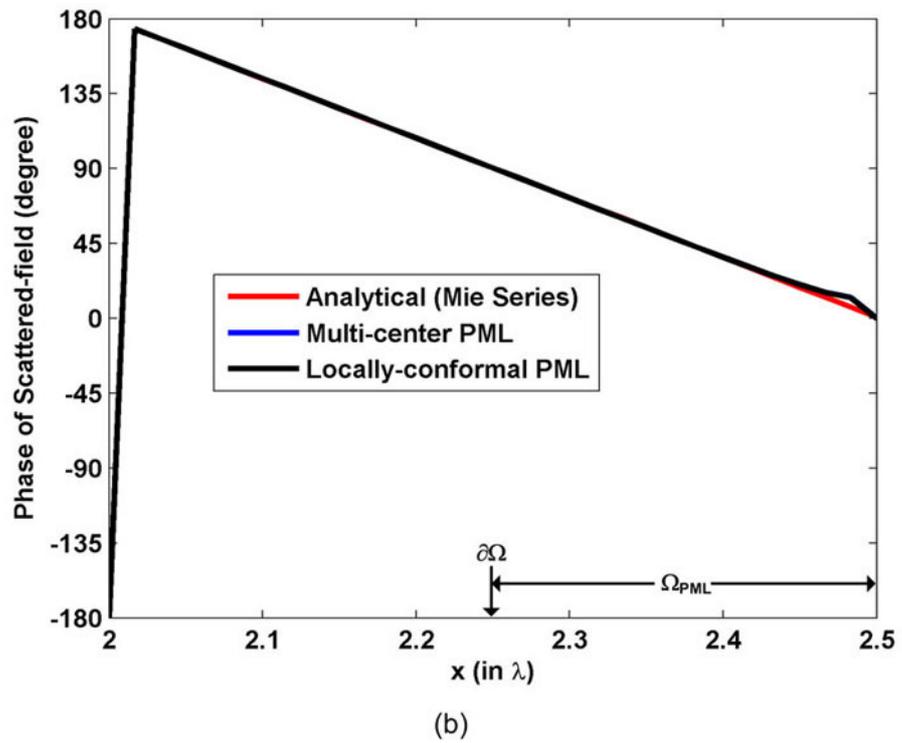
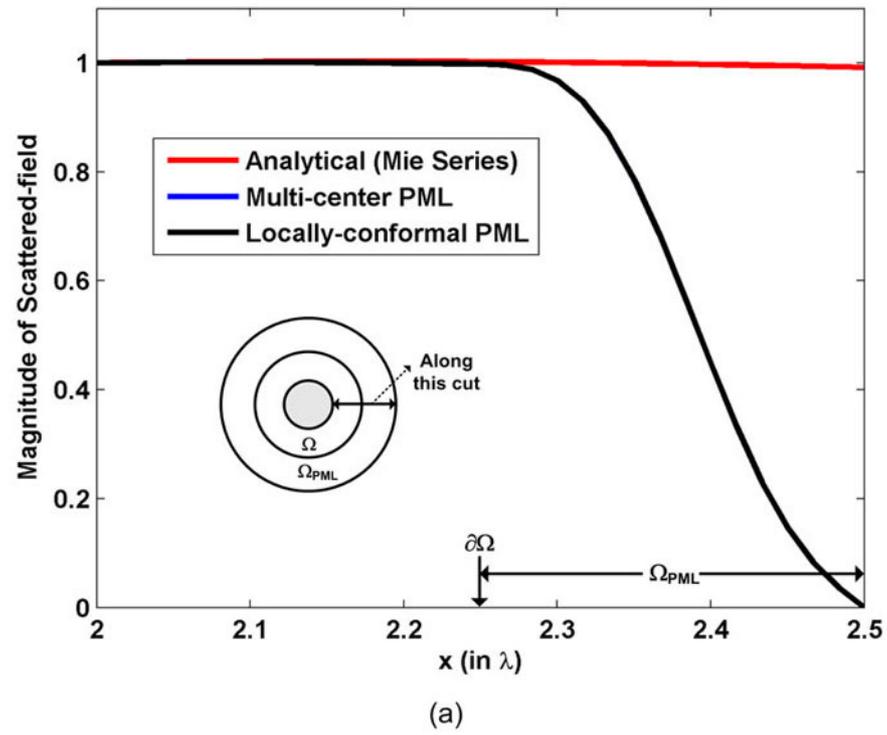


Figure 6.7. Scattered field along the x -axis for the scattering from circular PEC cylinder: (a) Magnitude, (b) Phase.

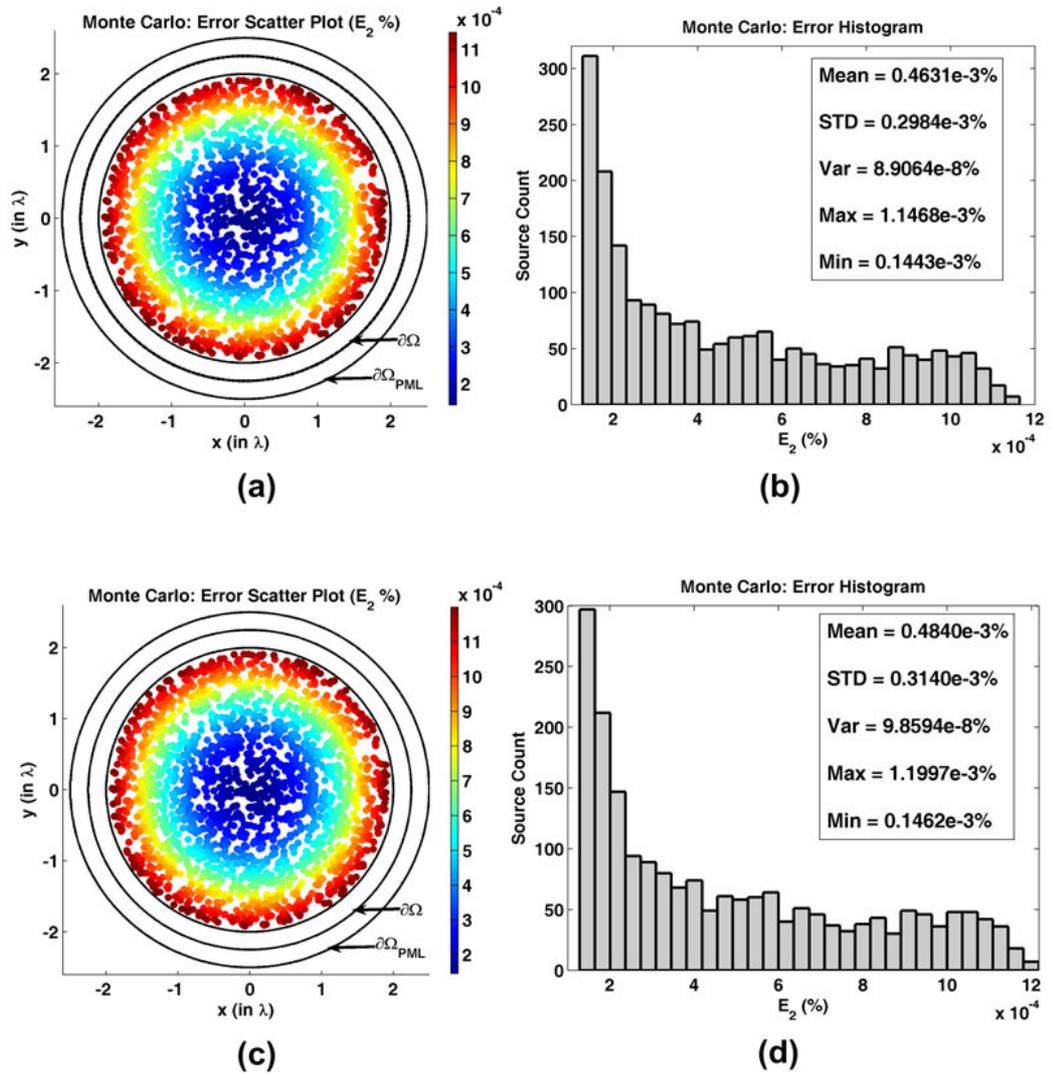
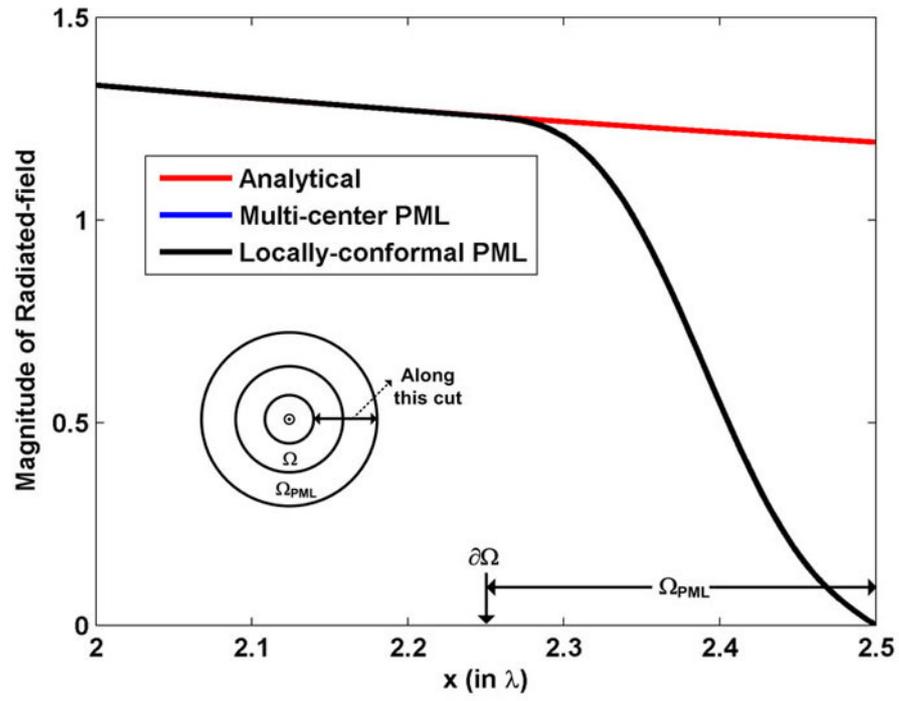
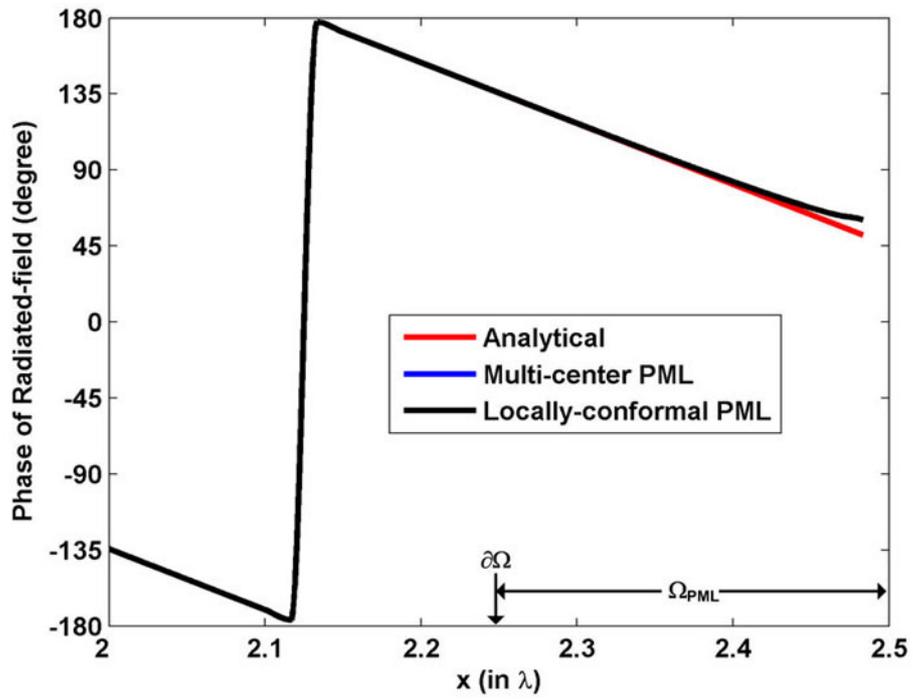


Figure 6.8. Error analysis for E_2 in the Monte Carlo Simulation of the radiation problem in circular domain: (a) Error scatter plot in the locally-conformal PML, (b) Error histogram and statistics in the locally-conformal PML, (c) Error scatter plot in the multi-center PML, (d) Error histogram and statistics in the multi-center PML.



(a)



(b)

Figure 6.9. Radiated field along the x -axis for the radiation problem in circular domain (line-source is located at the origin): (a) Magnitude, (b) Phase.

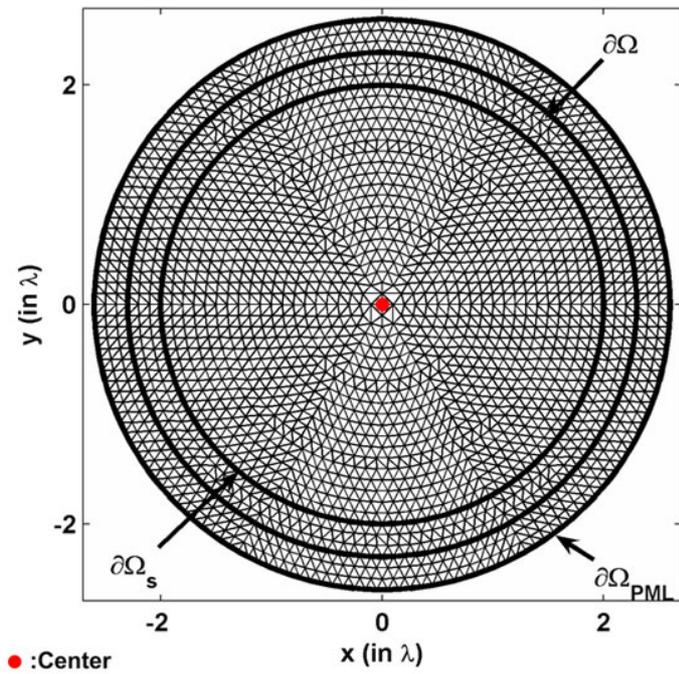


Figure 6.10. Mesh for scattering from circular dielectric cylinder (radius is 2λ).

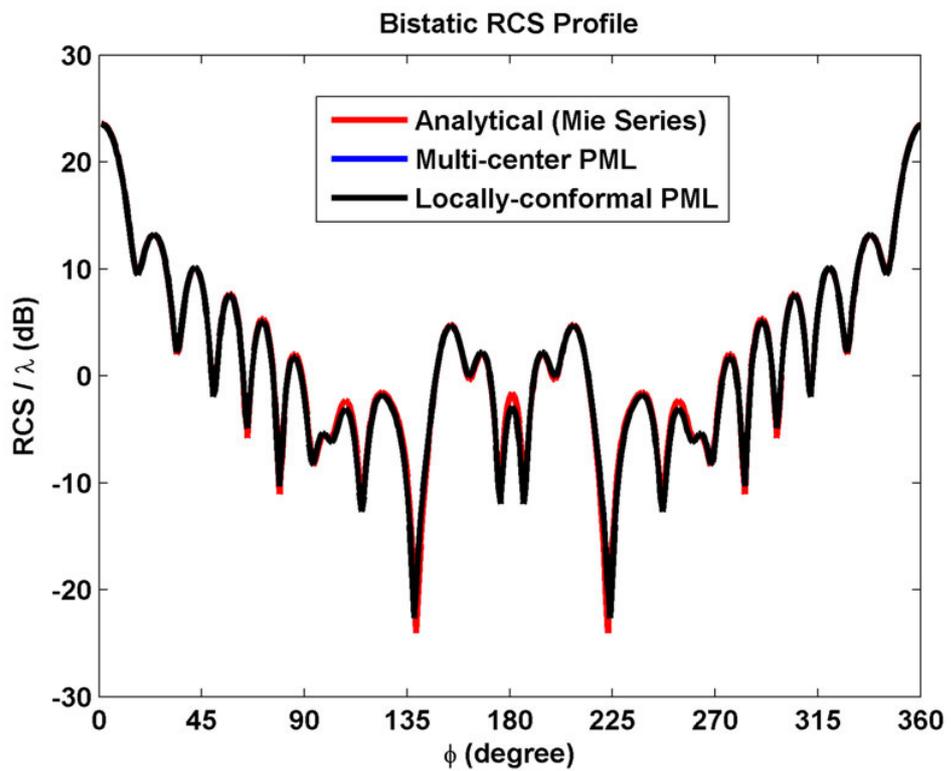


Figure 6.11. RCS profile for scattering from circular dielectric cylinder.

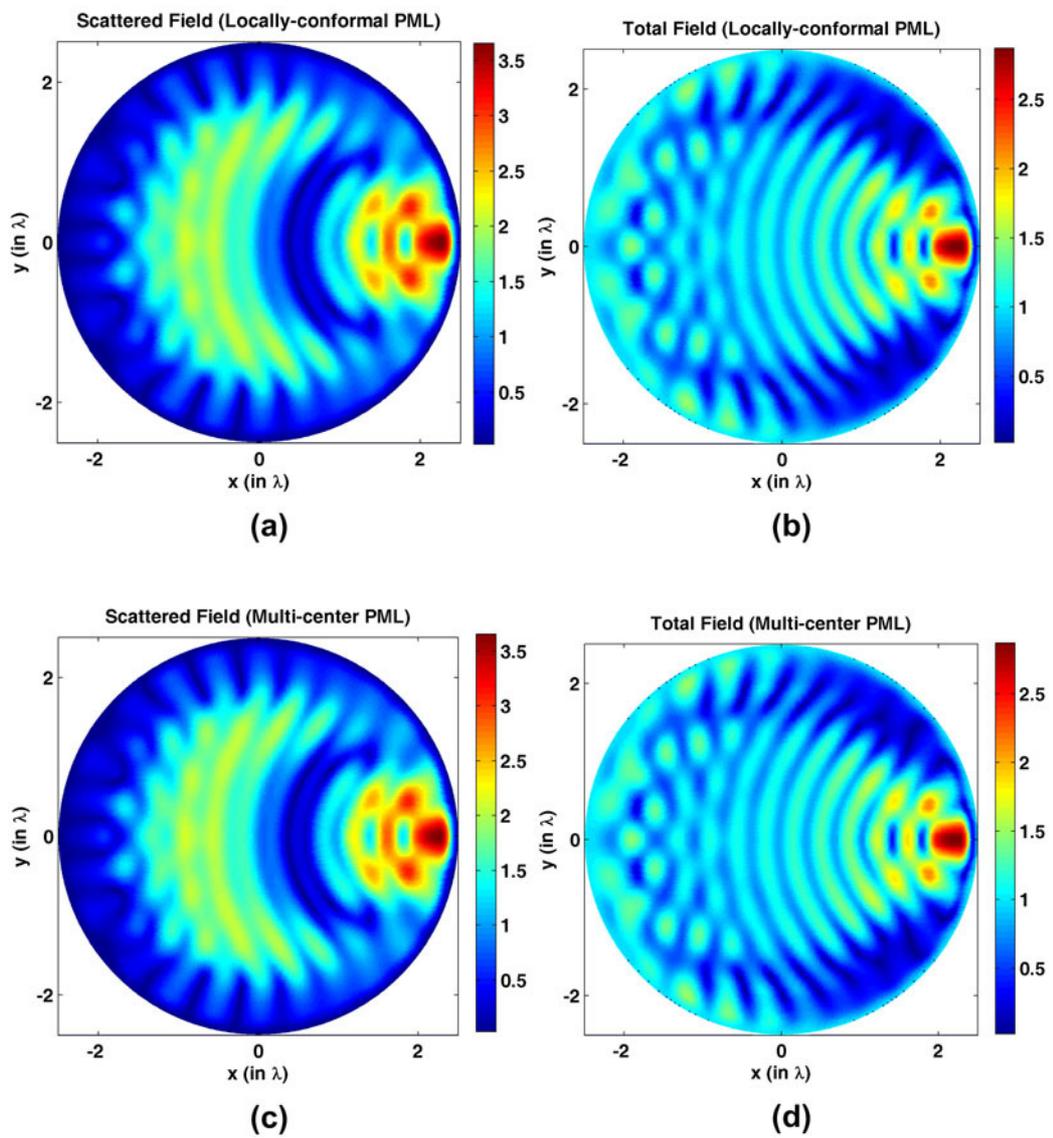


Figure 6.12. Field contours for scattering from circular dielectric cylinder (magnitude): (a-b) Scattered and total fields in the locally-conformal PML, (c-d) Scattered and total fields in the multi-center PML.

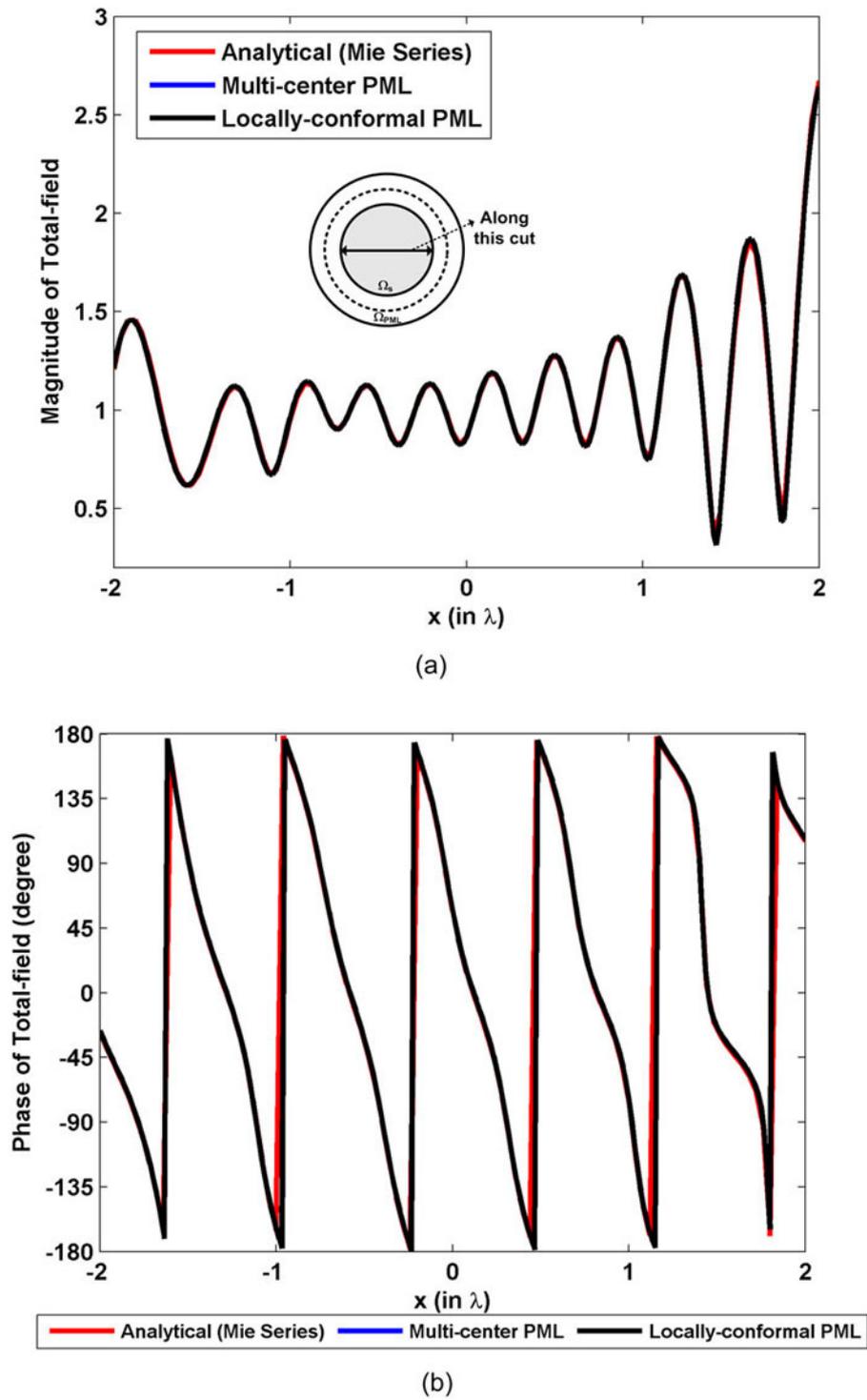


Figure 6.13. Total field inside the circular dielectric cylinder and along the x -axis: (a) Magnitude, (b) Phase.

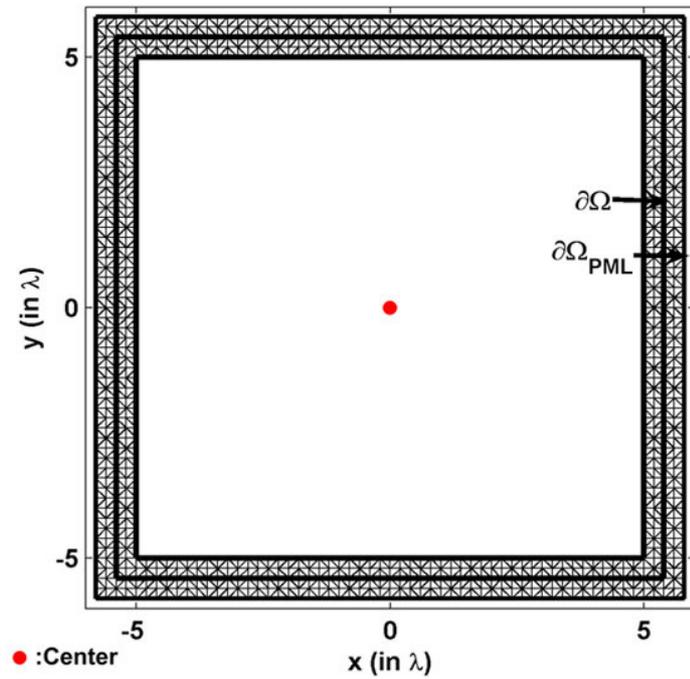


Figure 6.14. Mesh for square domain (edge-length is 10λ).

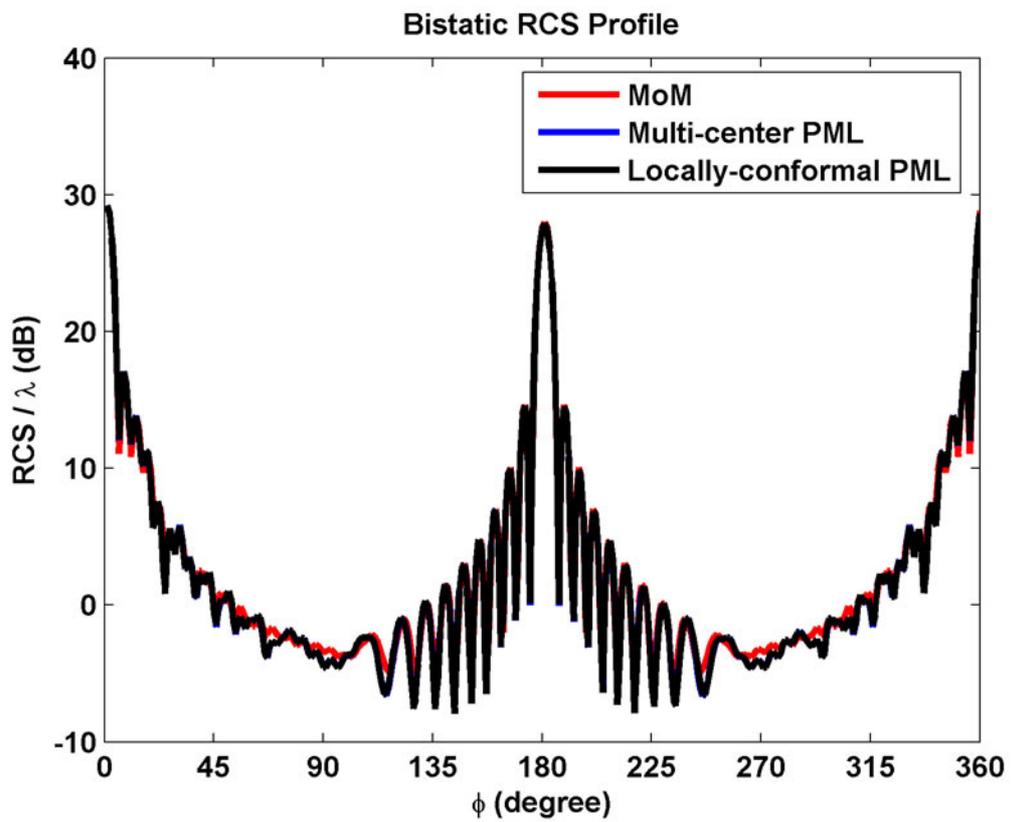


Figure 6.15. RCS profile for scattering from square PEC cylinder.

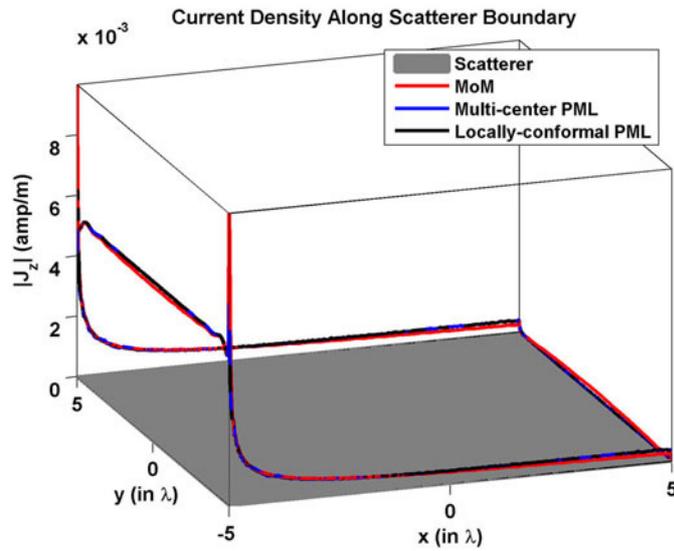


Figure 6.16. Magnitude of current density along square PEC cylinder.

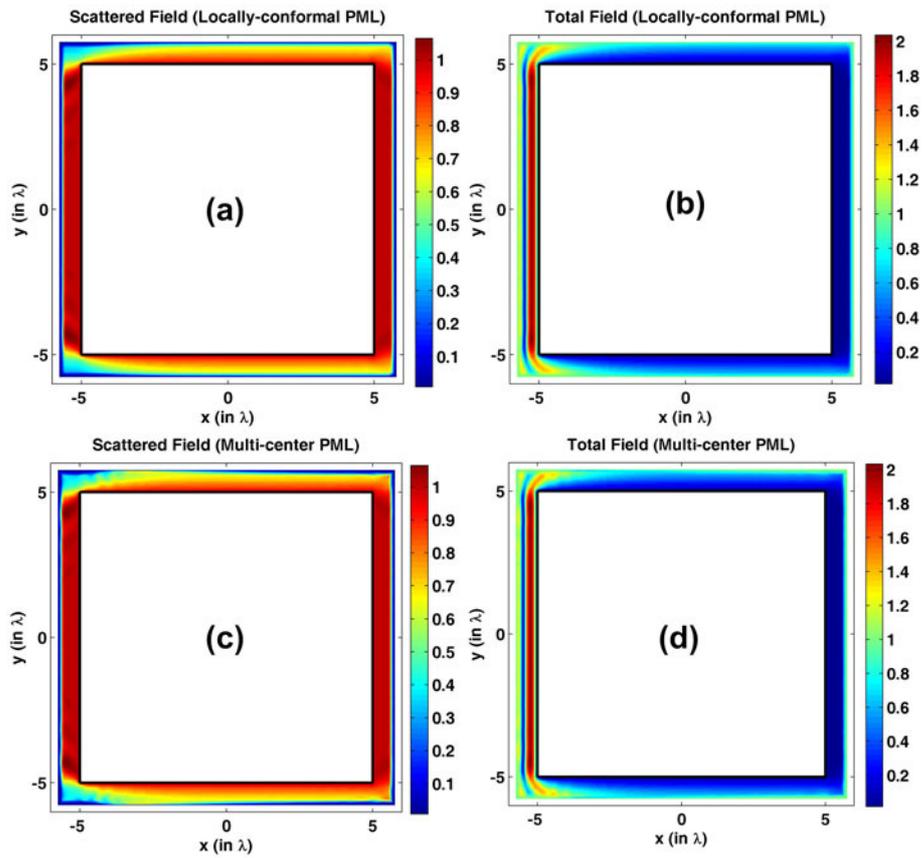


Figure 6.17. Field contours for square PEC cylinder (magnitude): (a-b) Scattered and total fields in the locally-conformal PML, (c-d) Scattered and total fields in the multi-center PML.

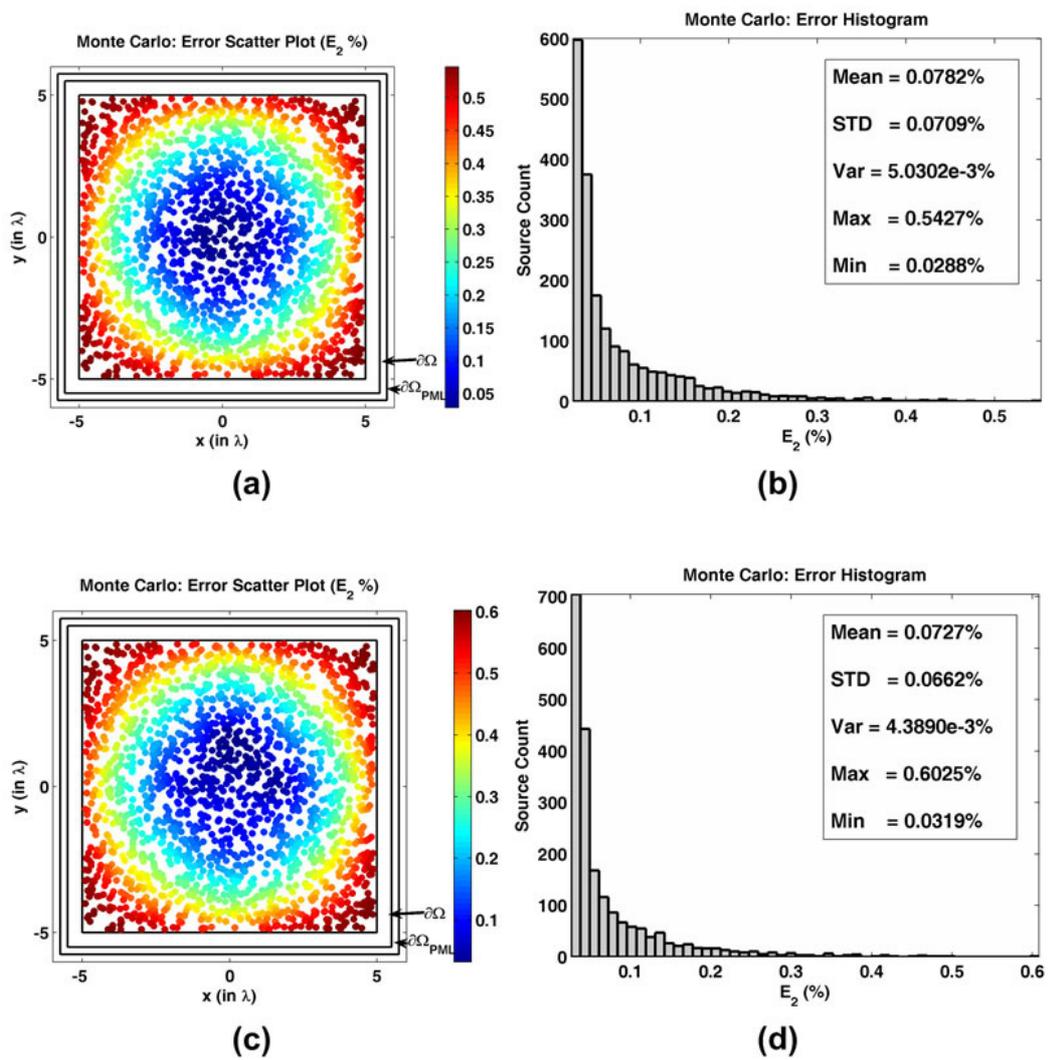


Figure 6.18. Error analysis for E_2 in the Monte Carlo Simulation of the radiation problem in square domain: (a) Error scatter plot in the locally-conformal PML, (b) Error histogram and statistics in the locally-conformal PML, (c) Error scatter plot in the multi-center PML, (d) Error histogram and statistics in the multi-center PML.

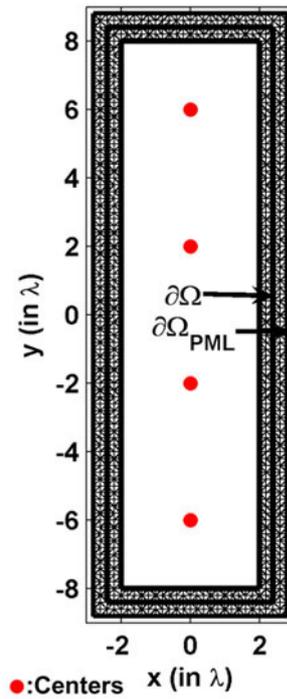


Figure 6.19. Mesh for rectangular domain ($4\lambda \times 16\lambda$).

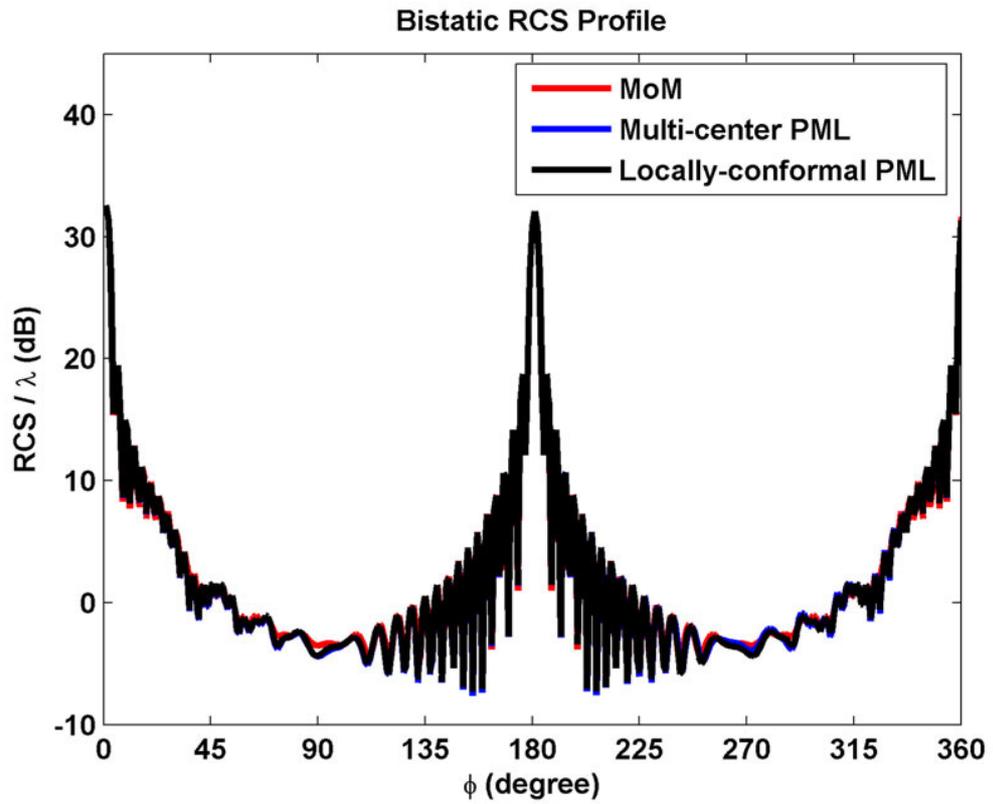


Figure 6.20. RCS profile for scattering from rectangular PEC cylinder.

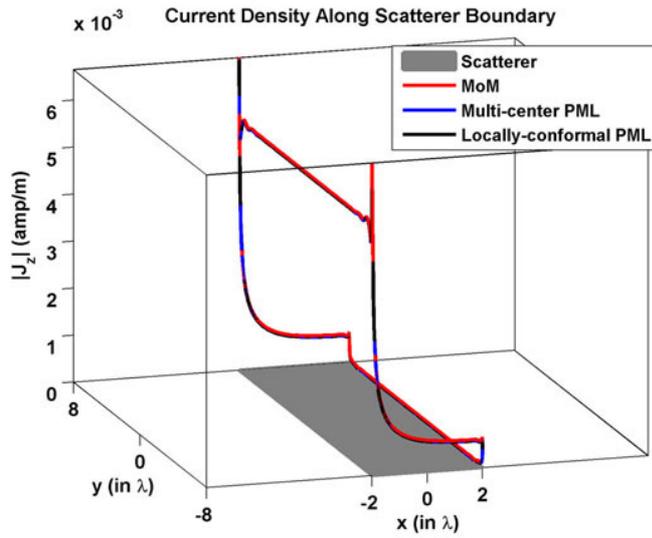


Figure 6.21. Magnitude of current density along rectangular PEC cylinder.

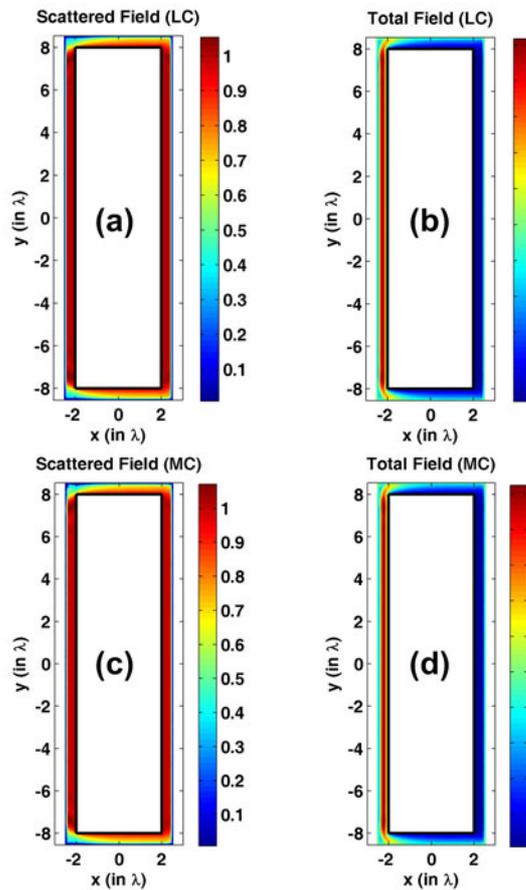


Figure 6.22. Field contours for rectangular PEC cylinder (magnitude): (a-b) Scattered and total fields in the locally-conformal PML, (c-d) Scattered and total fields in the multi-center PML.

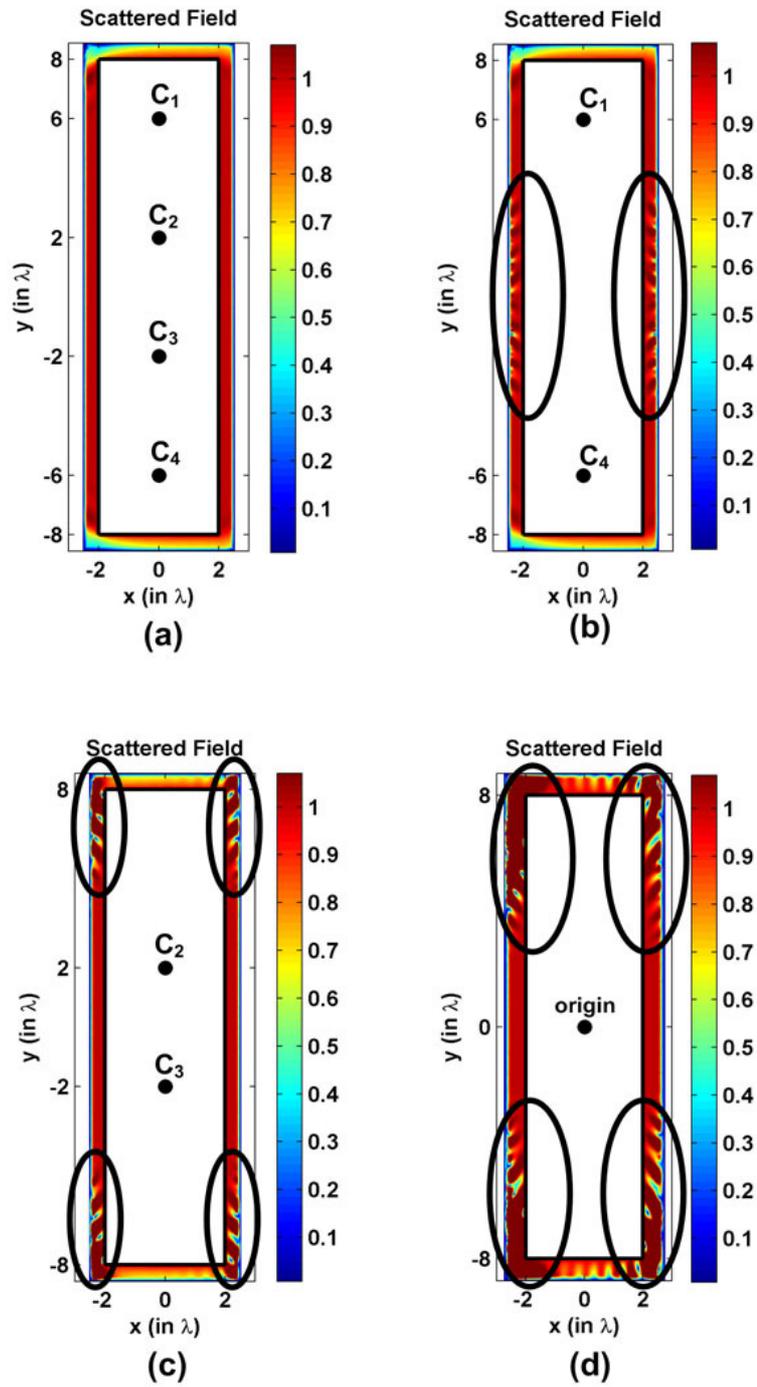


Figure 6.23. Scattered field contours (magnitude) for rectangular PEC cylinder with different centers in the multi-center PML (black ellipses indicate the regions which do not satisfy the 45° criterion): (a) Four centers, (b) Two centers (C_1 and C_4), (c) Two centers (C_2 and C_3), (d) One center at the origin.

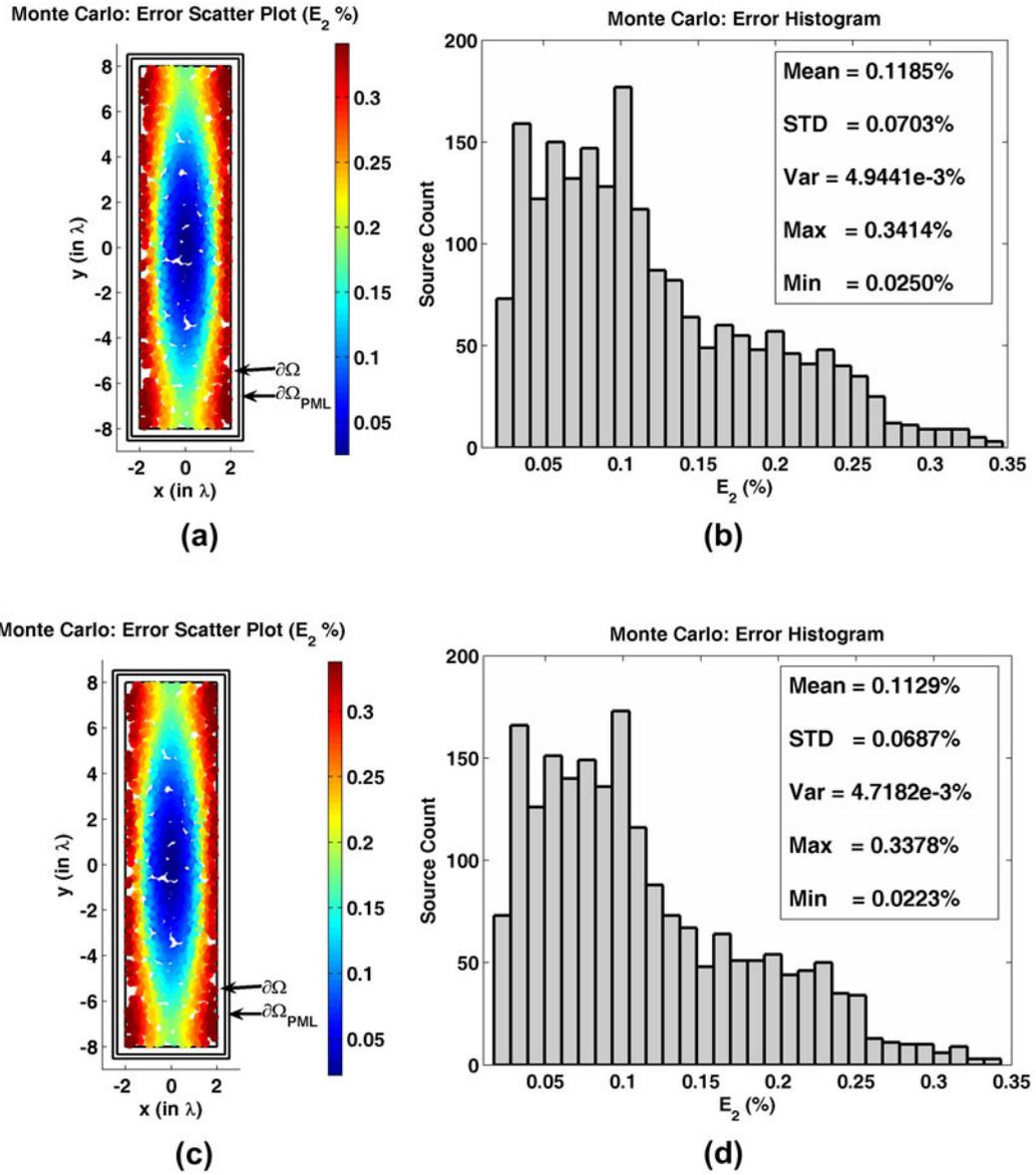


Figure 6.24. Error analysis for E_2 in the Monte Carlo Simulation of the radiation problem in rectangular domain: (a) Error scatter plot in the locally-conformal PML, (b) Error histogram and statistics in the locally-conformal PML, (c) Error scatter plot in the multi-center PML, (d) Error histogram and statistics in the multi-center PML.

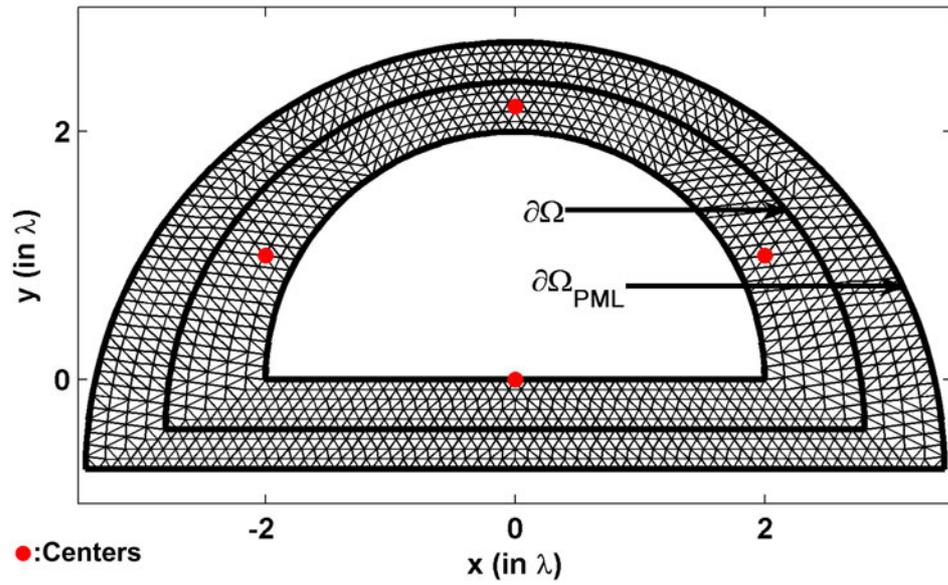


Figure 6.25. Mesh for half-circular domain (radius is 2λ).

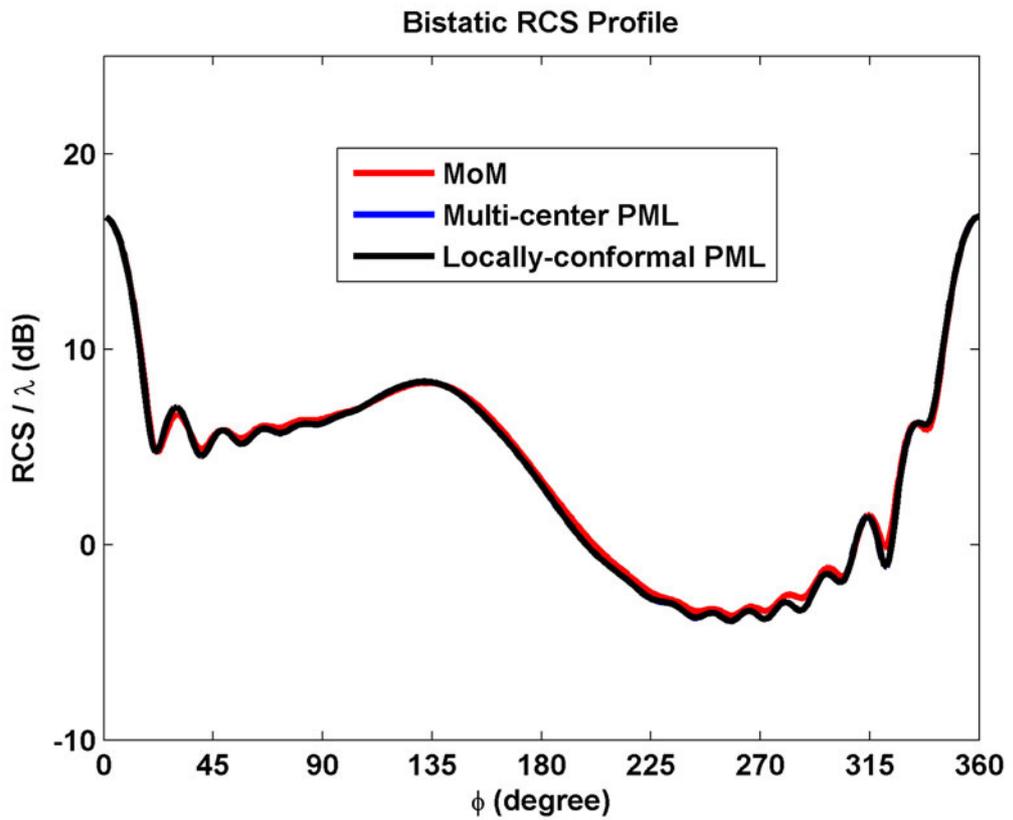


Figure 6.26. RCS profile for scattering from half-circular PEC cylinder.

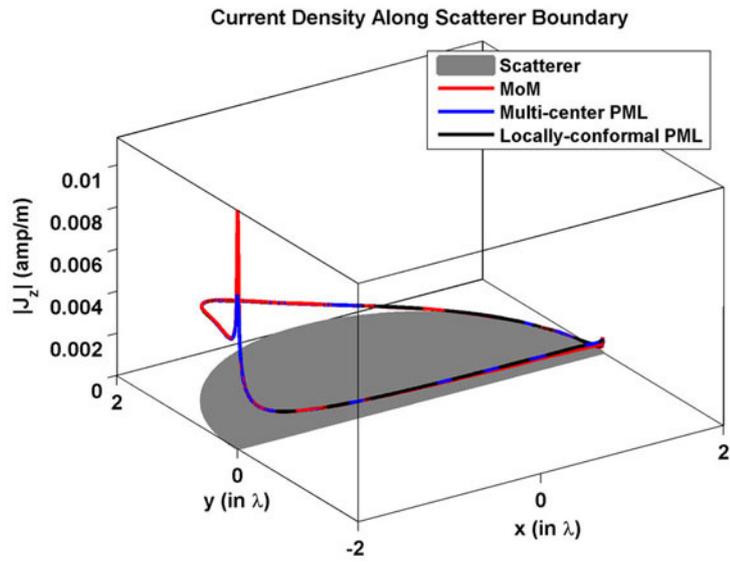


Figure 6.27. Magnitude of current density along half-circular PEC cylinder.

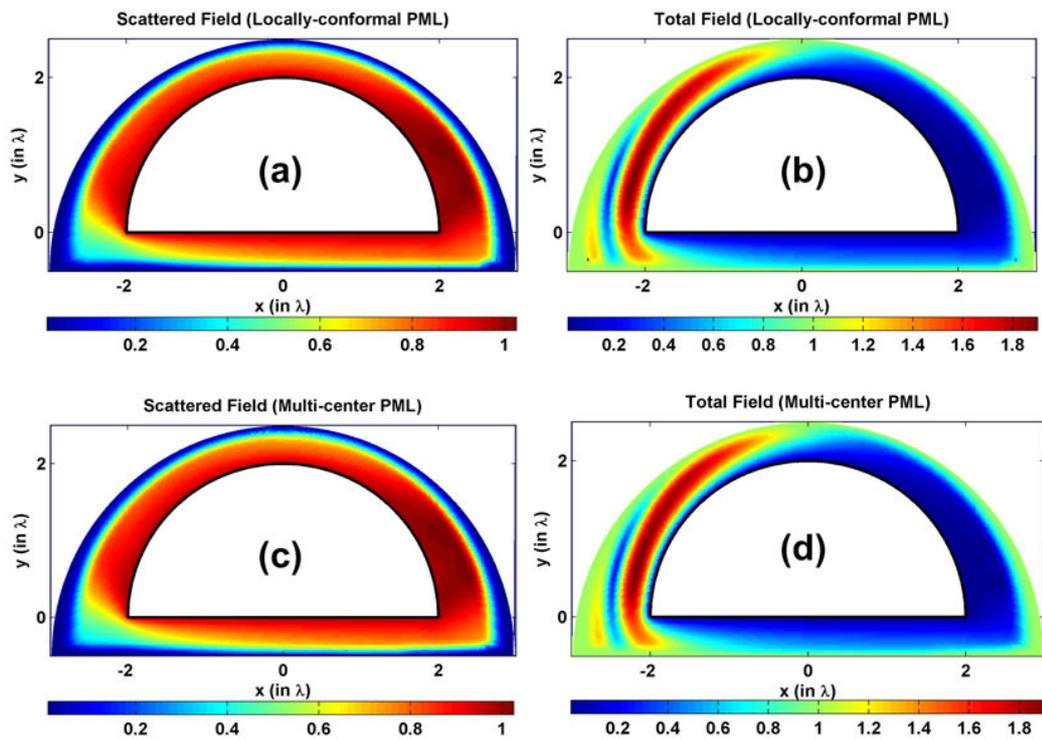


Figure 6.28. Field contours for half-circular PEC cylinder (magnitude): (a-b) Scattered and total fields in the locally-conformal PML, (c-d) Scattered and total fields in the multi-center PML.

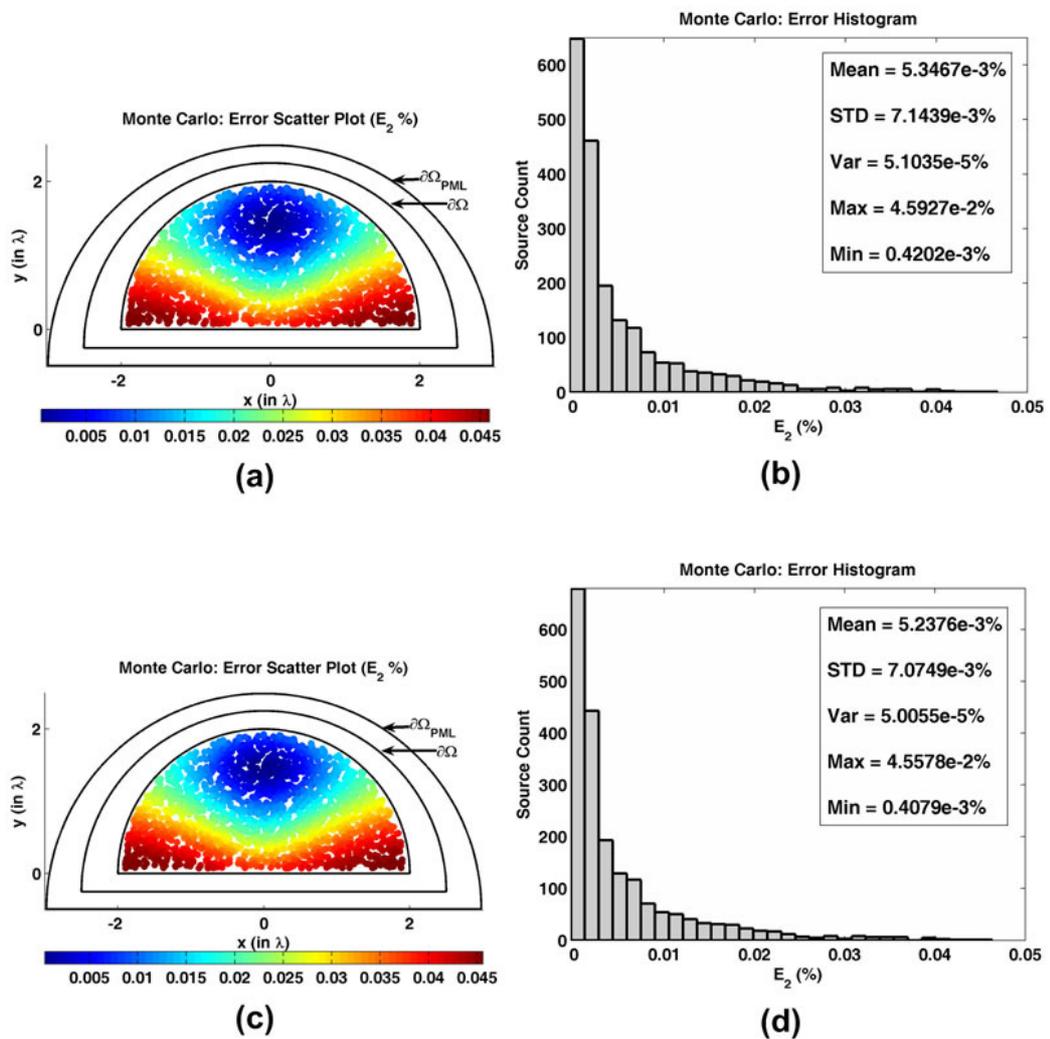


Figure 6.29. Error analysis for E_2 in the Monte Carlo Simulation of the radiation problem in half-circular domain: (a) Error scatter plot in the locally-conformal PML, (b) Error histogram and statistics in the locally-conformal PML, (c) Error scatter plot in the multi-center PML, (d) Error histogram and statistics in the multi-center PML.

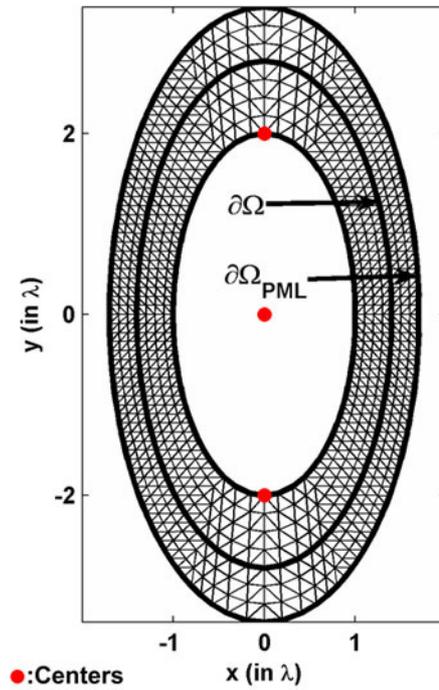


Figure 6.30. Mesh for elliptical domain (semi-major axis is 2λ , axial ratio is 2).

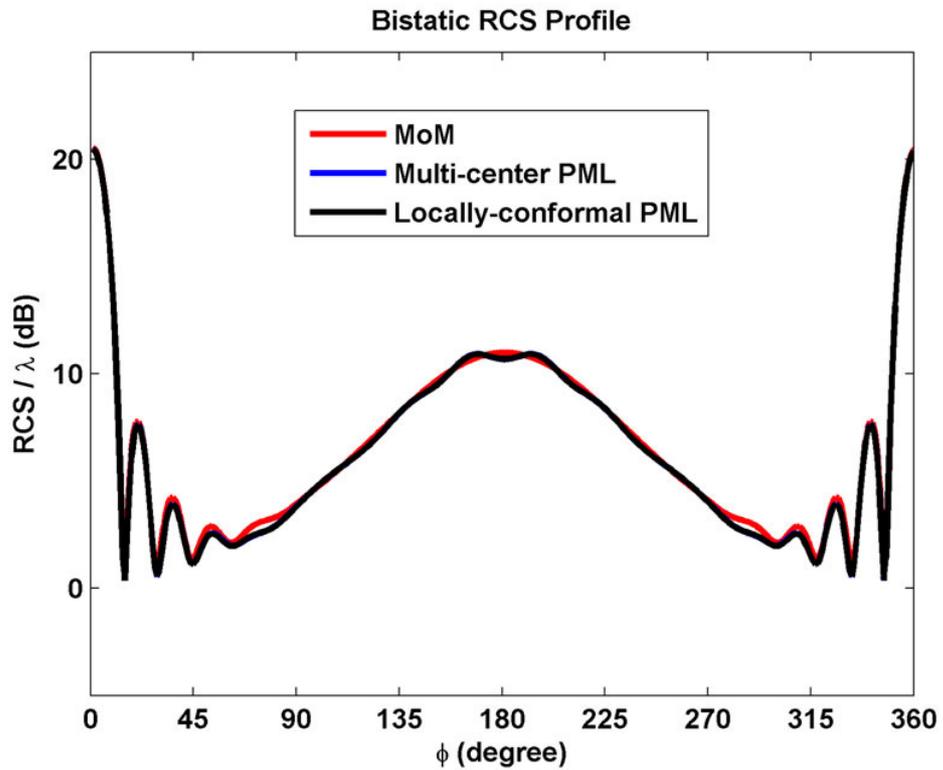


Figure 6.31. RCS profile for scattering from elliptical PEC cylinder.

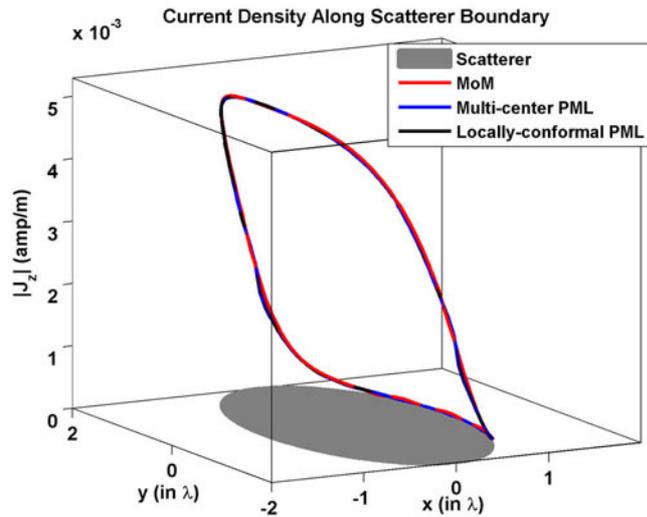


Figure 6.32. Magnitude of current density along elliptical PEC cylinder.

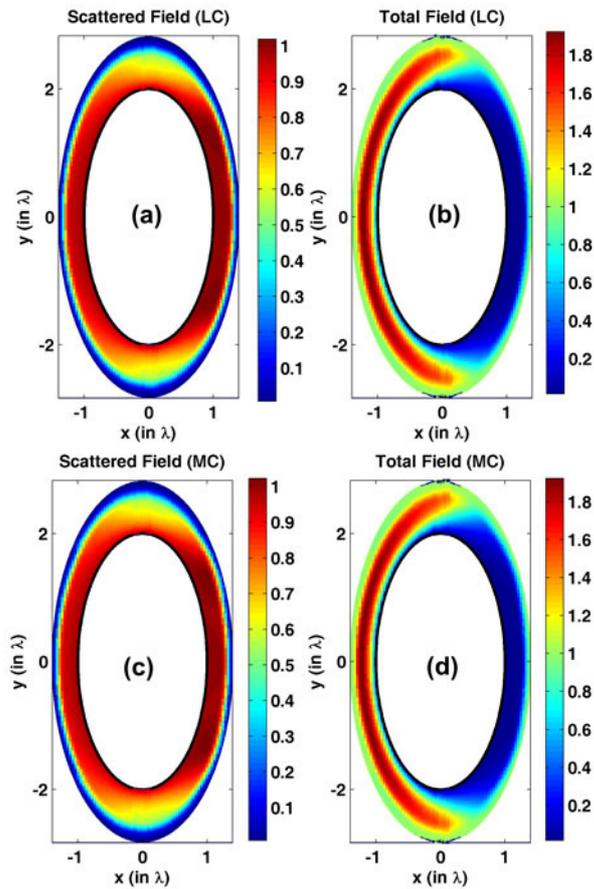


Figure 6.33. Field contours for elliptical PEC cylinder (magnitude): (a-b) Scattered and total fields in the locally-conformal PML, (c-d) Scattered and total fields in the multi-center PML.

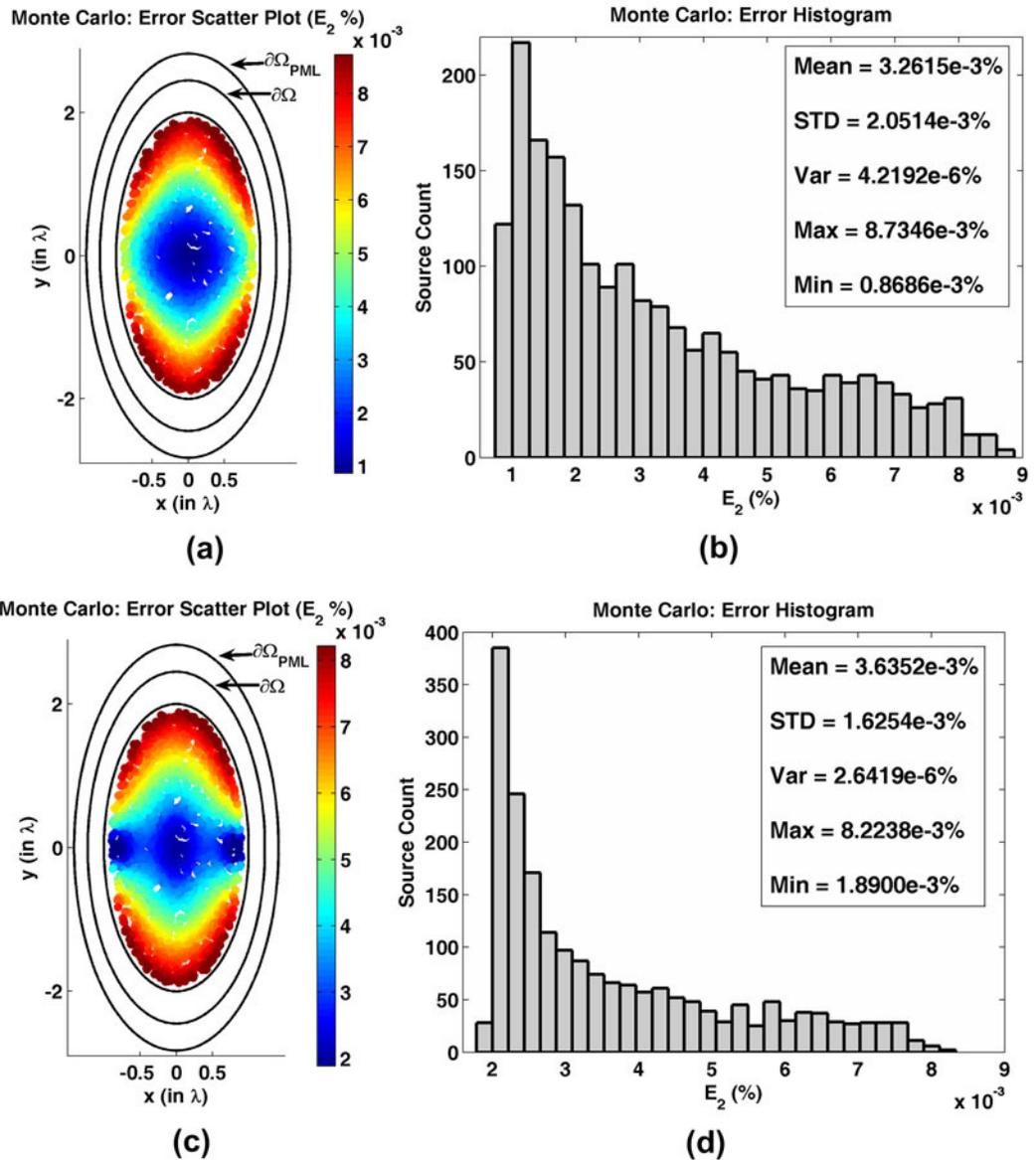


Figure 6.34. Error analysis for E_2 in the Monte Carlo Simulation of the radiation problem in elliptical domain: (a) Error scatter plot in the locally-conformal PML, (b) Error histogram and statistics in the locally-conformal PML, (c) Error scatter plot in the multi-center PML, (d) Error histogram and statistics in the multi-center PML.

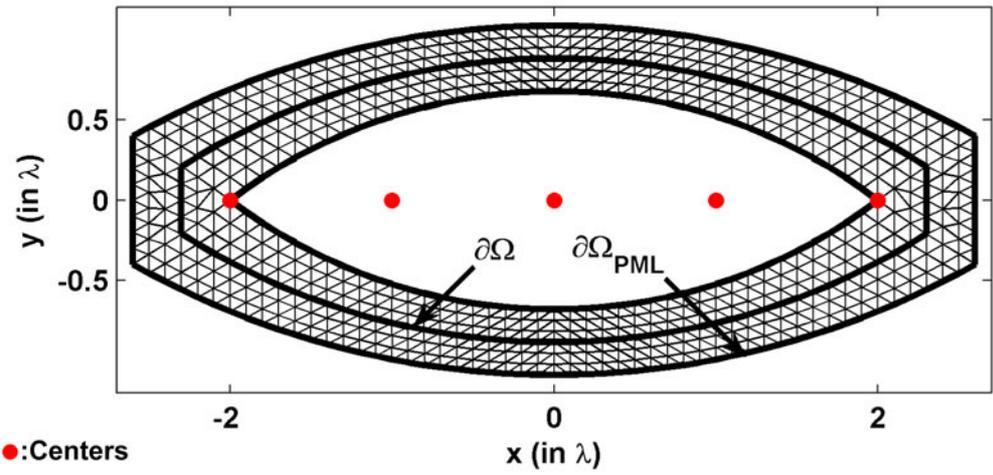


Figure 6.35. Mesh for ogive domain (max. diameter is 2λ , nose-angle is 75°).

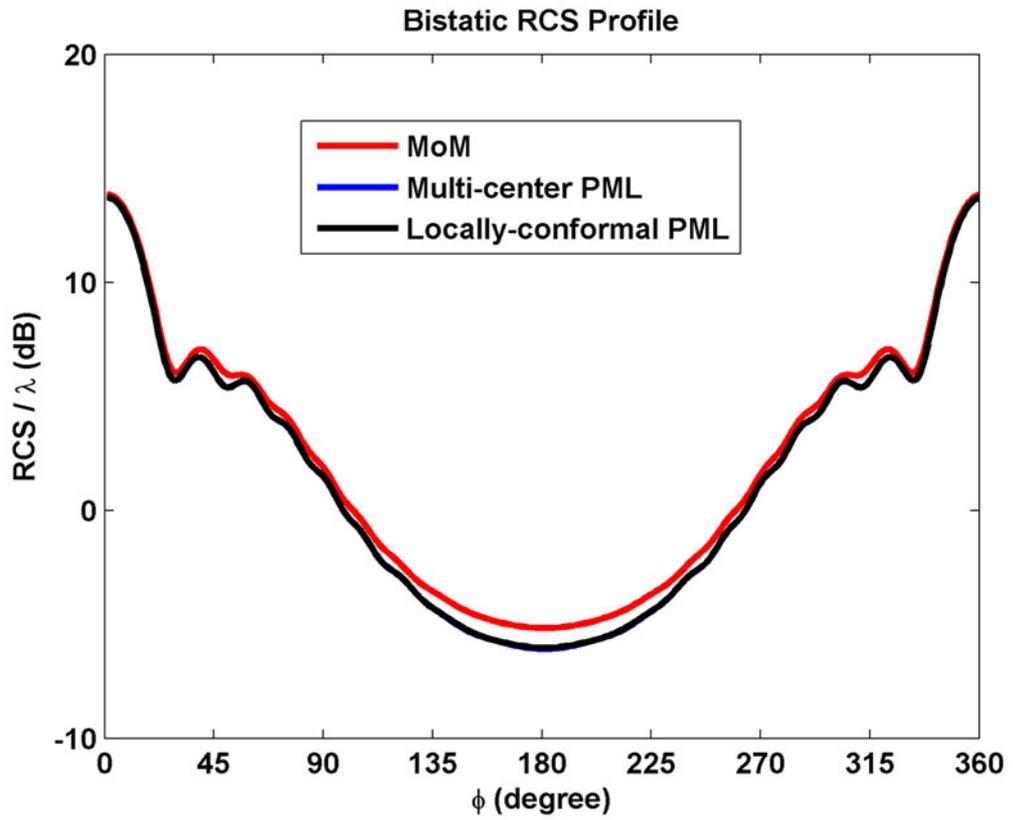


Figure 6.36. RCS profile for scattering from ogive PEC cylinder.

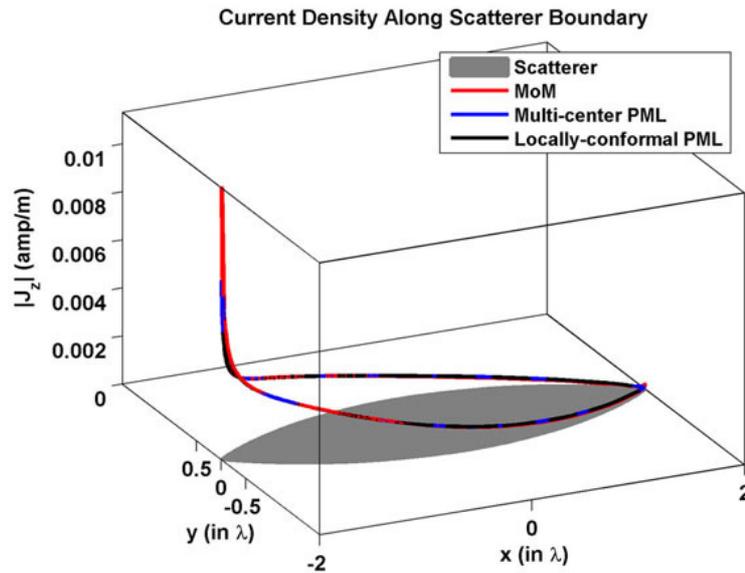


Figure 6.37. Magnitude of current density along ogive PEC cylinder.

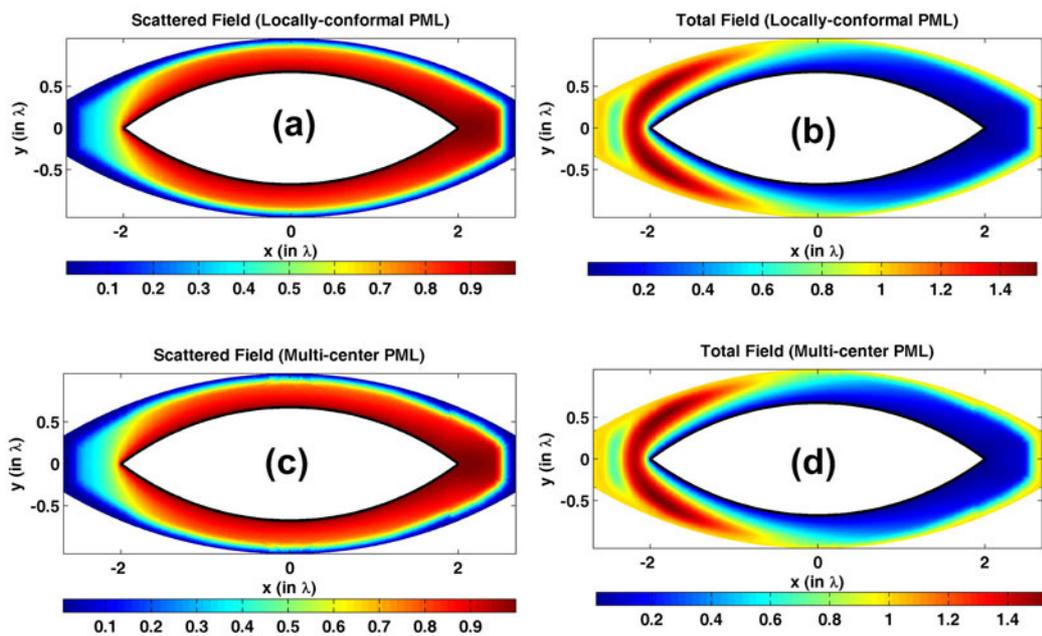


Figure 6.38. Field contours for ogive PEC cylinder (magnitude): (a-b) Scattered and total fields in the locally-conformal PML, (c-d) Scattered and total fields in the multi-center PML.

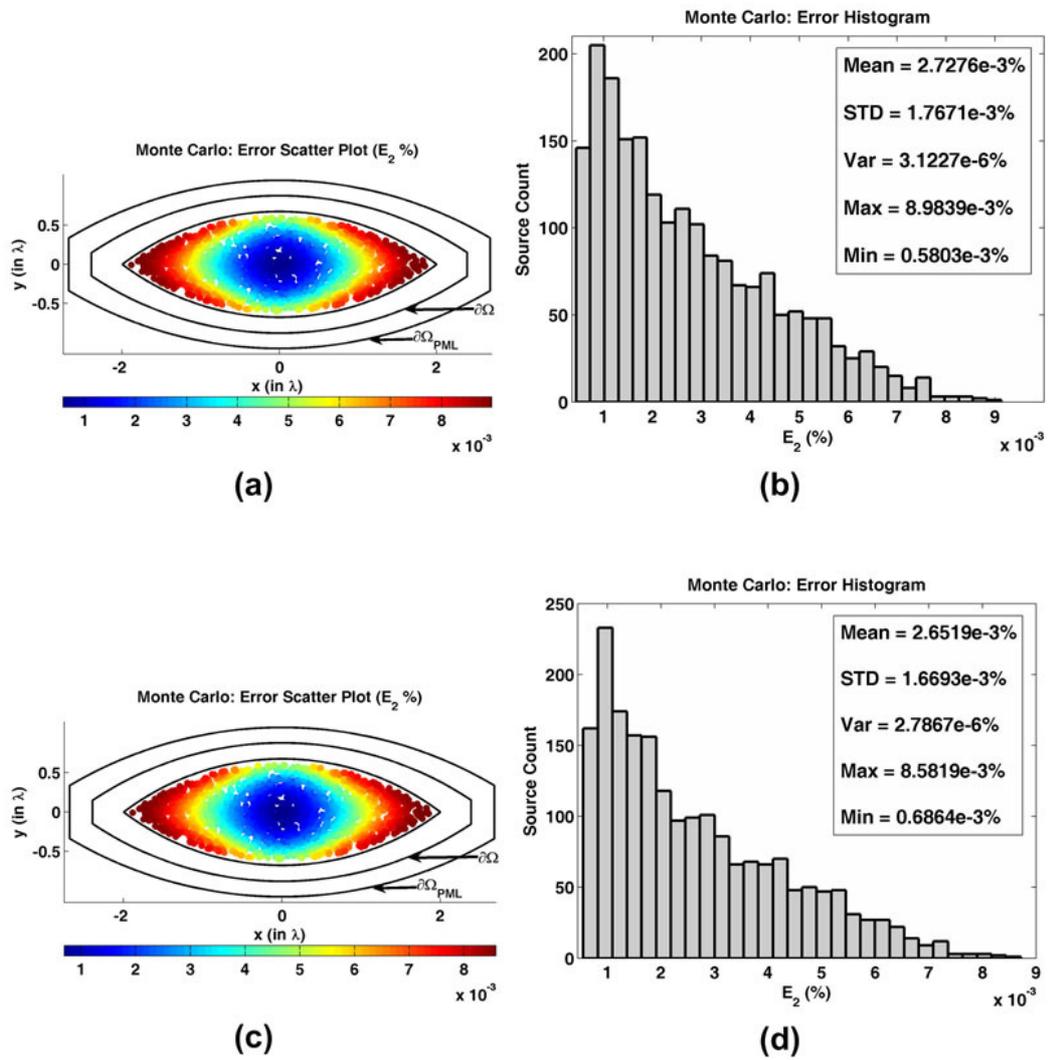


Figure 6.39. Error analysis for E_2 in the Monte Carlo Simulation of the radiation problem in ogive domain: (a) Error scatter plot in the locally-conformal PML, (b) Error histogram and statistics in the locally-conformal PML, (c) Error scatter plot in the multi-center PML, (d) Error histogram and statistics in the multi-center PML.

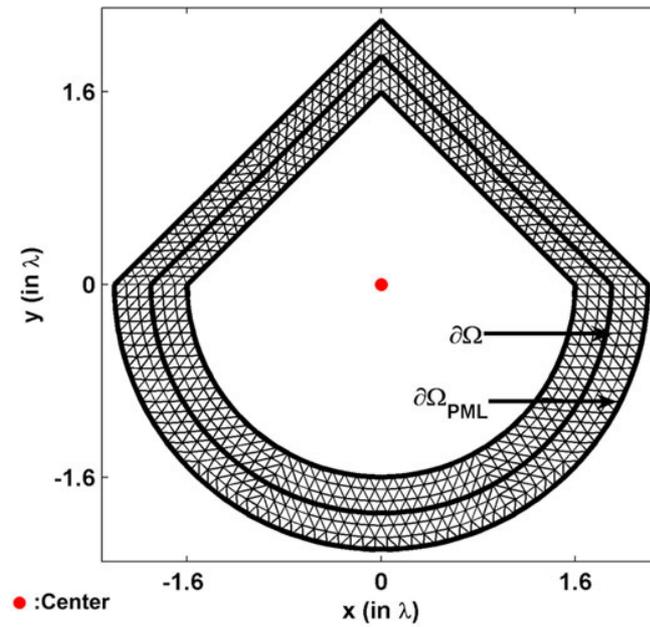


Figure 6.40. Mesh for triangular+halfcircular domain (radius 1.6λ , nose-angle 90°).

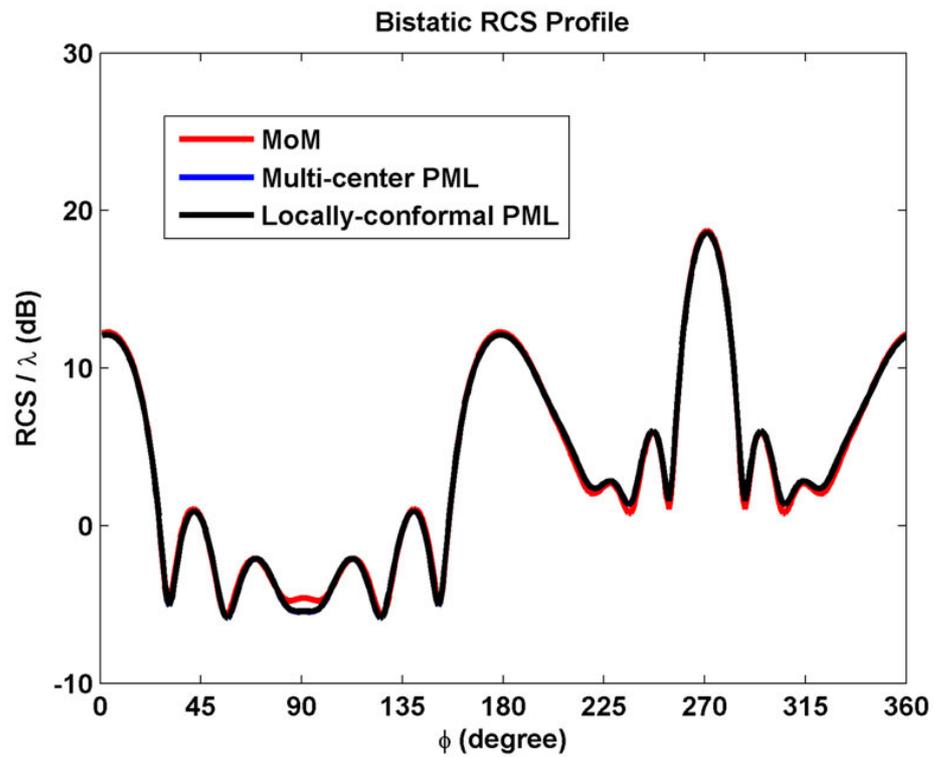


Figure 6.41. RCS profile for scattering from triangular+halfcircular PEC cylinder.

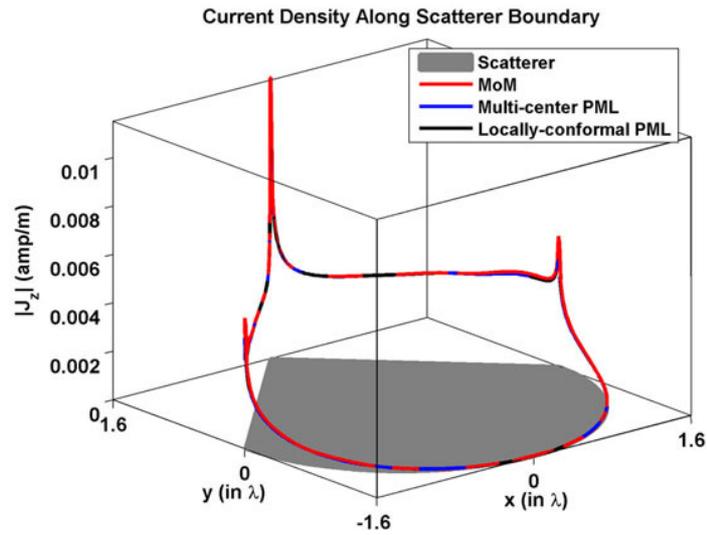


Figure 6.42. Magnitude of current density along triangular+halfcircular PEC cylinder.

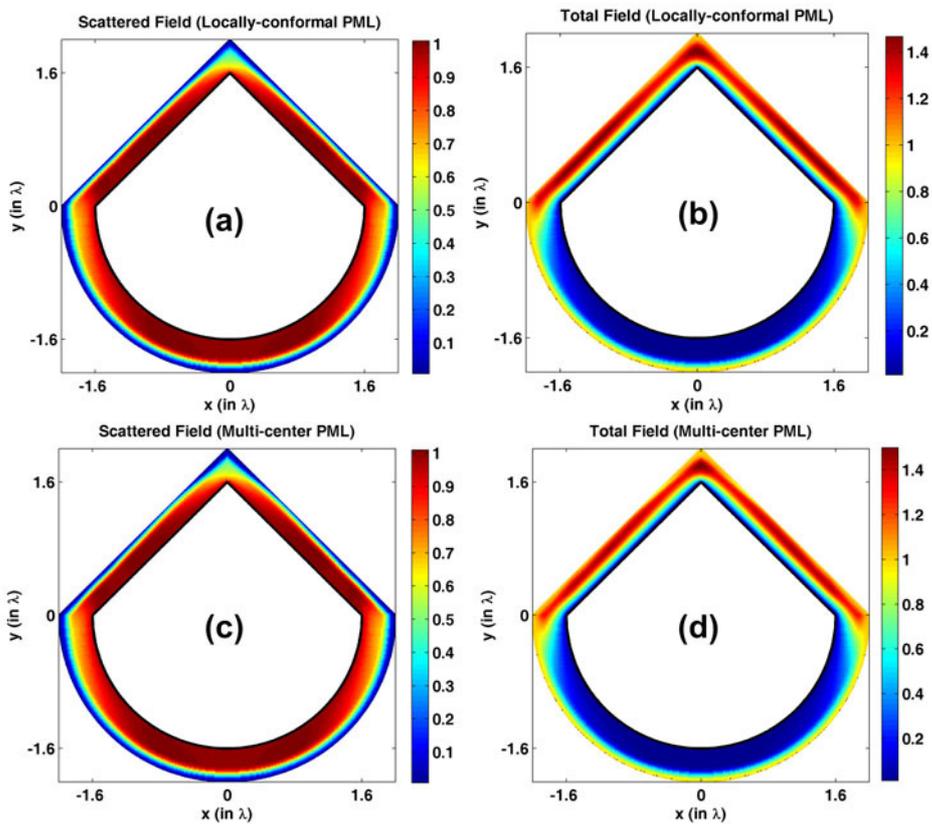


Figure 6.43. Field contours for triangular+halfcircular PEC cylinder (magnitude): (a-b) Scattered and total fields in the locally-conformal PML, (c-d) Scattered and total fields in the multi-center PML.

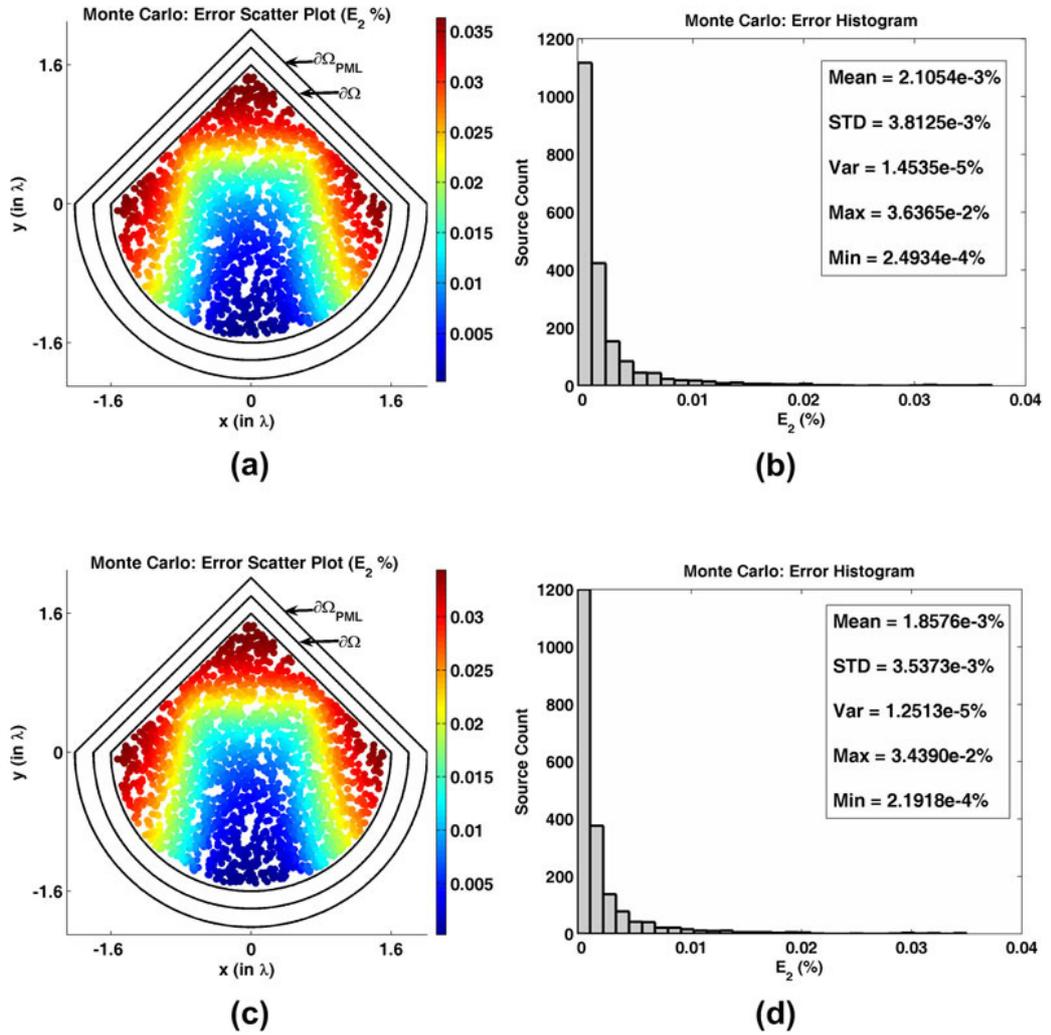


Figure 6.44. Error analysis for E_2 in the Monte Carlo Simulation of the radiation problem in triangular+halfcircular domain: (a) Error scatter plot in the locally-conformal PML, (b) Error histogram and statistics in the locally-conformal PML, (c) Error scatter plot in the multi-center PML, (d) Error histogram and statistics in the multi-center PML.

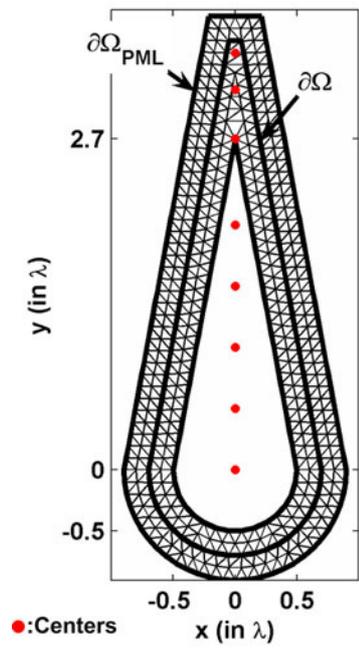


Figure 6.45. Mesh for sharp triangular+halfcircular domain (radius 0.5λ , nose-angle 20°).

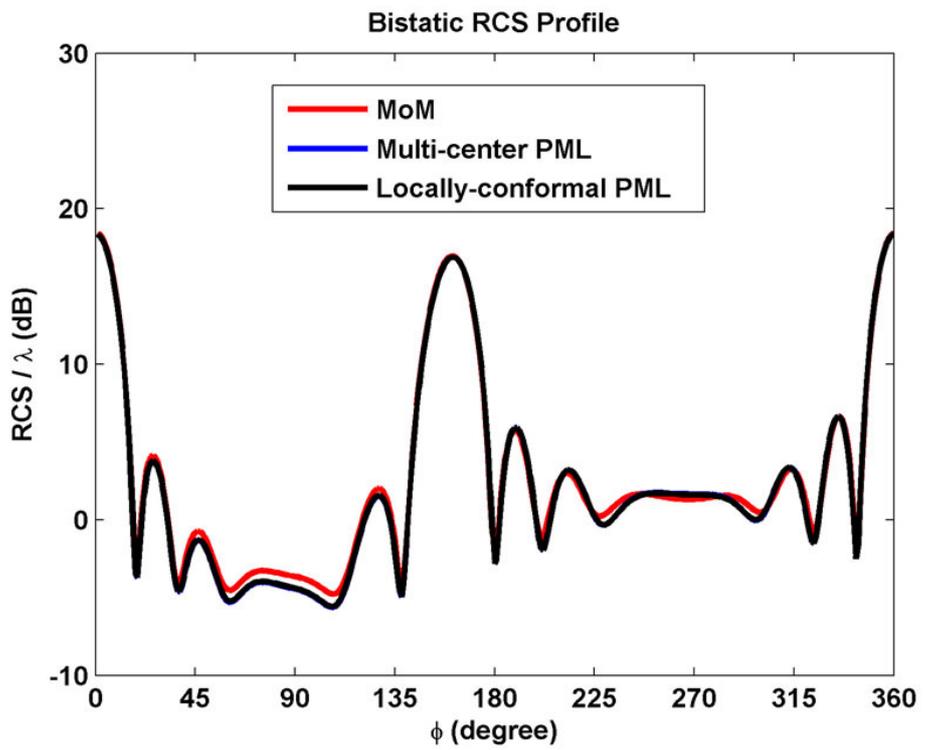


Figure 6.46. RCS profile for scattering from sharp triangular+halfcircular PEC cylinder.

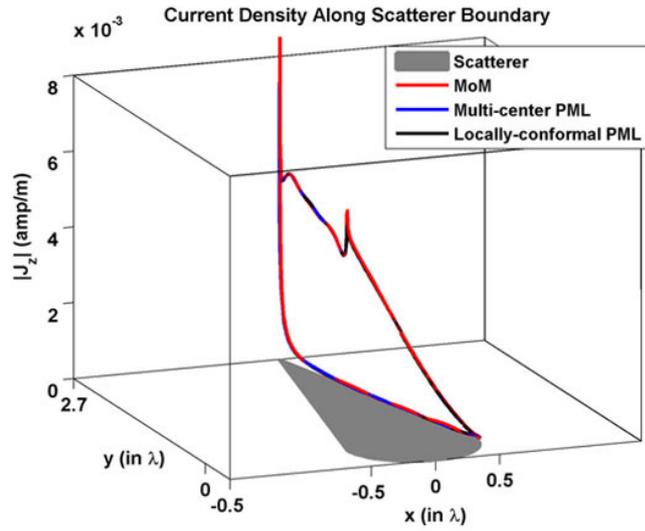


Figure 6.47. Magnitude of current density along sharp triangular+halfcircular cylinder.

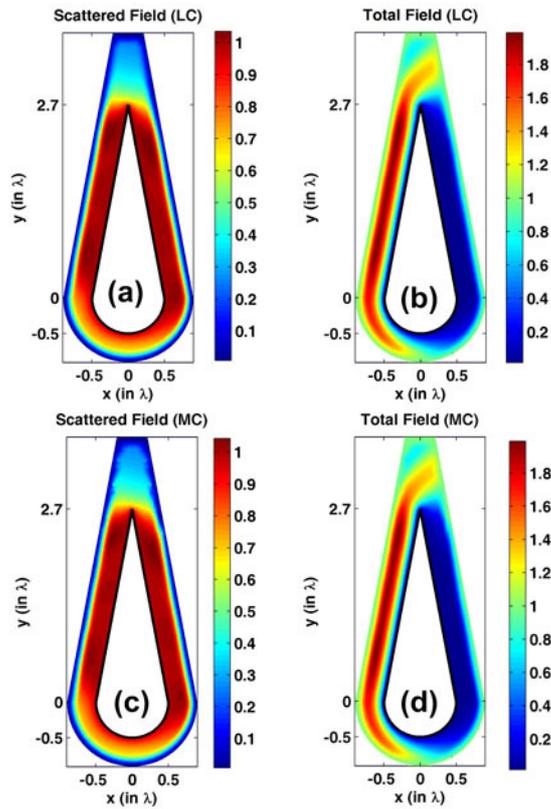


Figure 6.48. Field contours for sharp triangular+halfcircular PEC cylinder (magnitude): (a-b) Scattered and total fields in the locally-conformal PML, (c-d) Scattered and total fields in the multi-center PML.

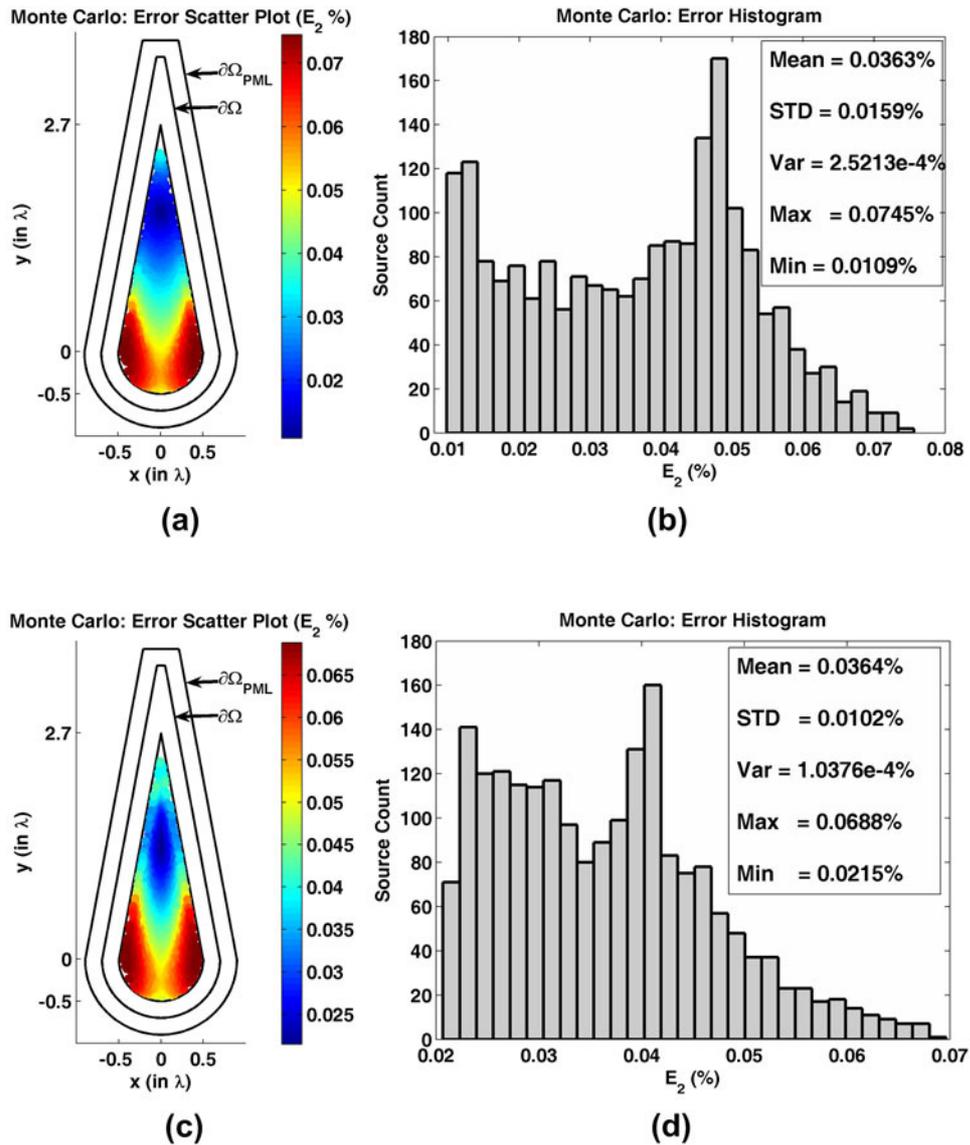


Figure 6.49. Error analysis for E_2 in the Monte Carlo Simulation of the radiation problem in sharp triangular+halfcircular domain: (a) Error scatter plot in the locally-conformal PML, (b) Error histogram and statistics in the locally-conformal PML, (c) Error scatter plot in the multi-center PML, (d) Error histogram and statistics in the multi-center PML.

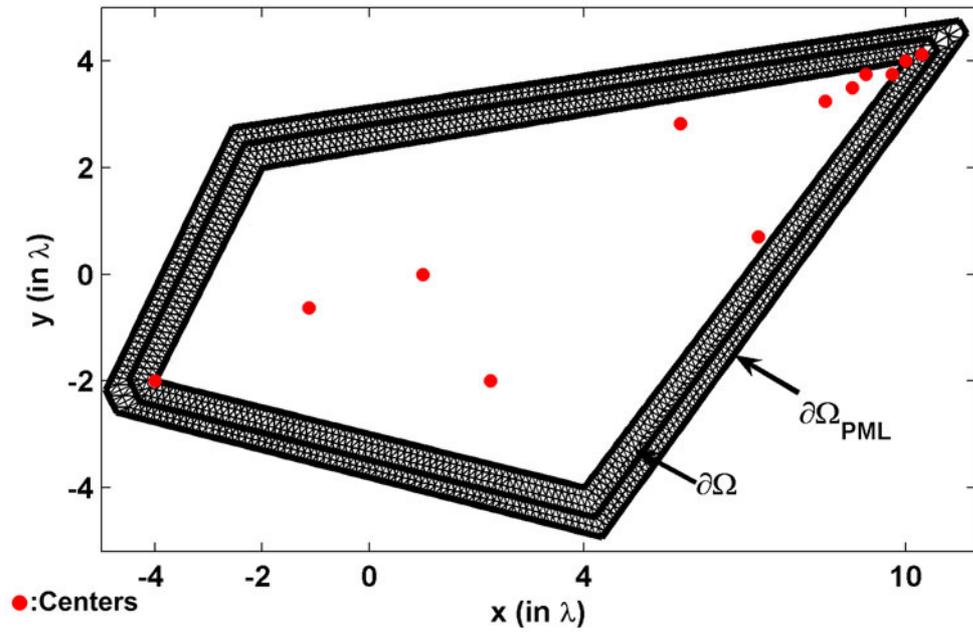


Figure 6.50. Mesh for quadrilateral domain (vertices are shown by tick labels).

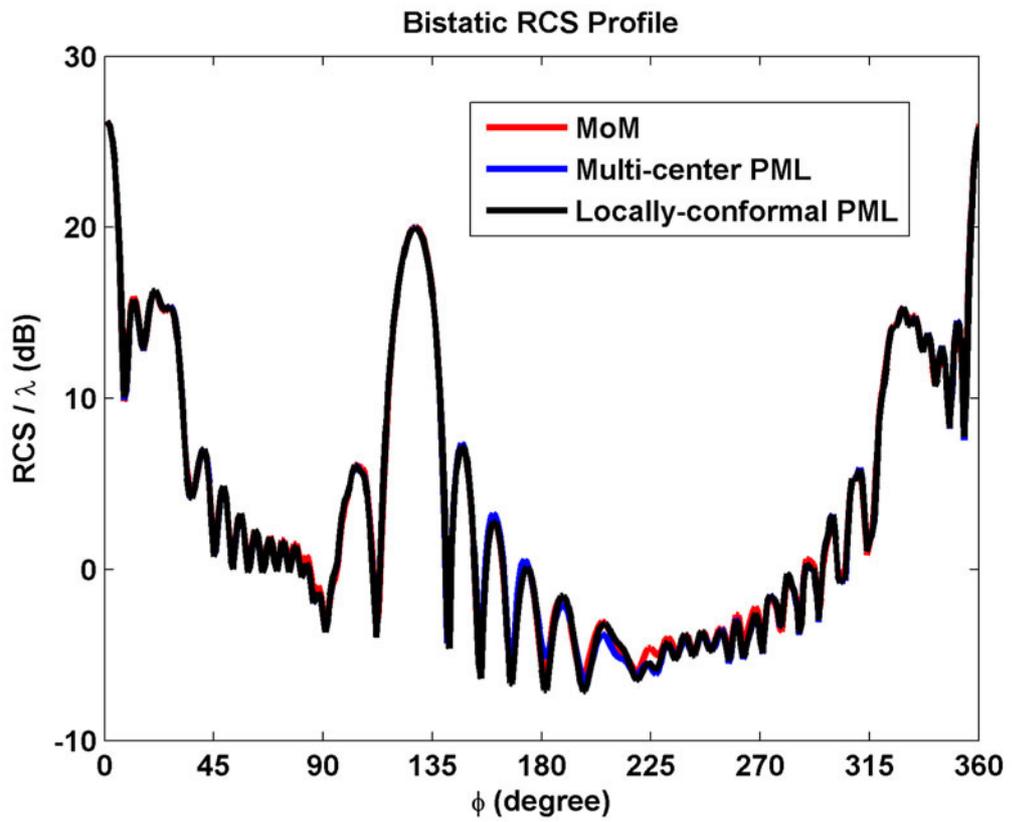


Figure 6.51. RCS profile for scattering from quadrilateral PEC cylinder.

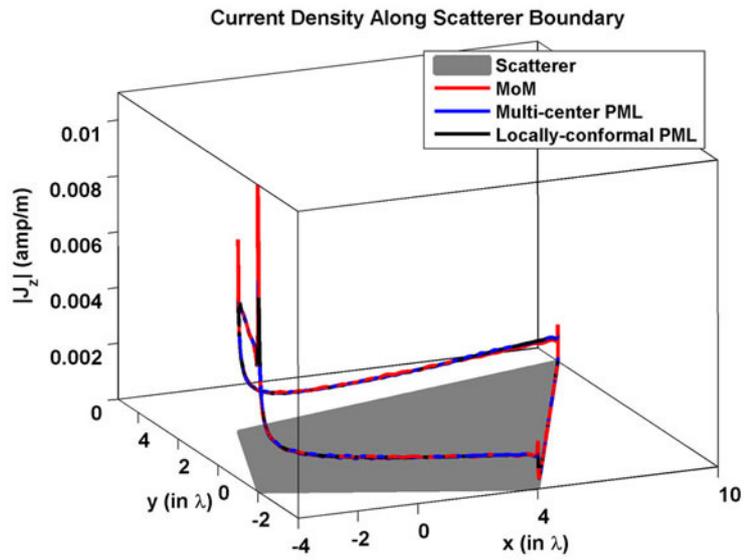


Figure 6.52. Magnitude of current density along quadrilateral PEC cylinder.

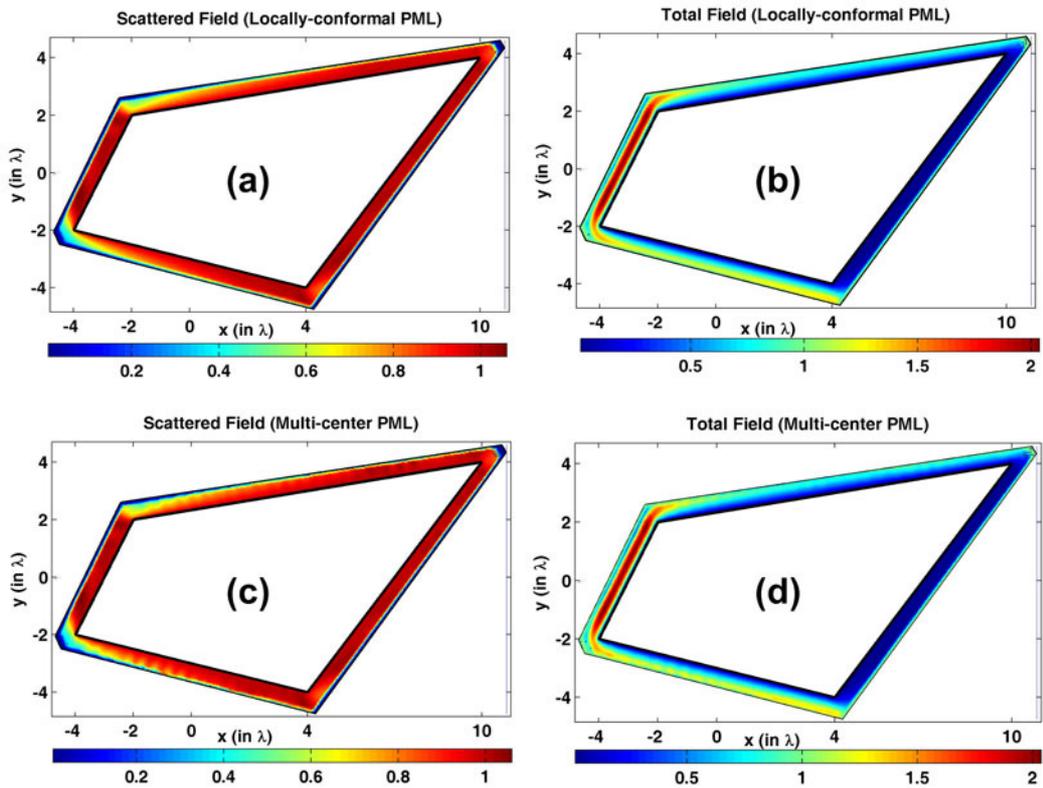


Figure 6.53. Field contours for quadrilateral PEC cylinder (magnitude): (a-b) Scattered and total fields in the locally-conformal PML, (c-d) Scattered and total fields in the multi-center PML.

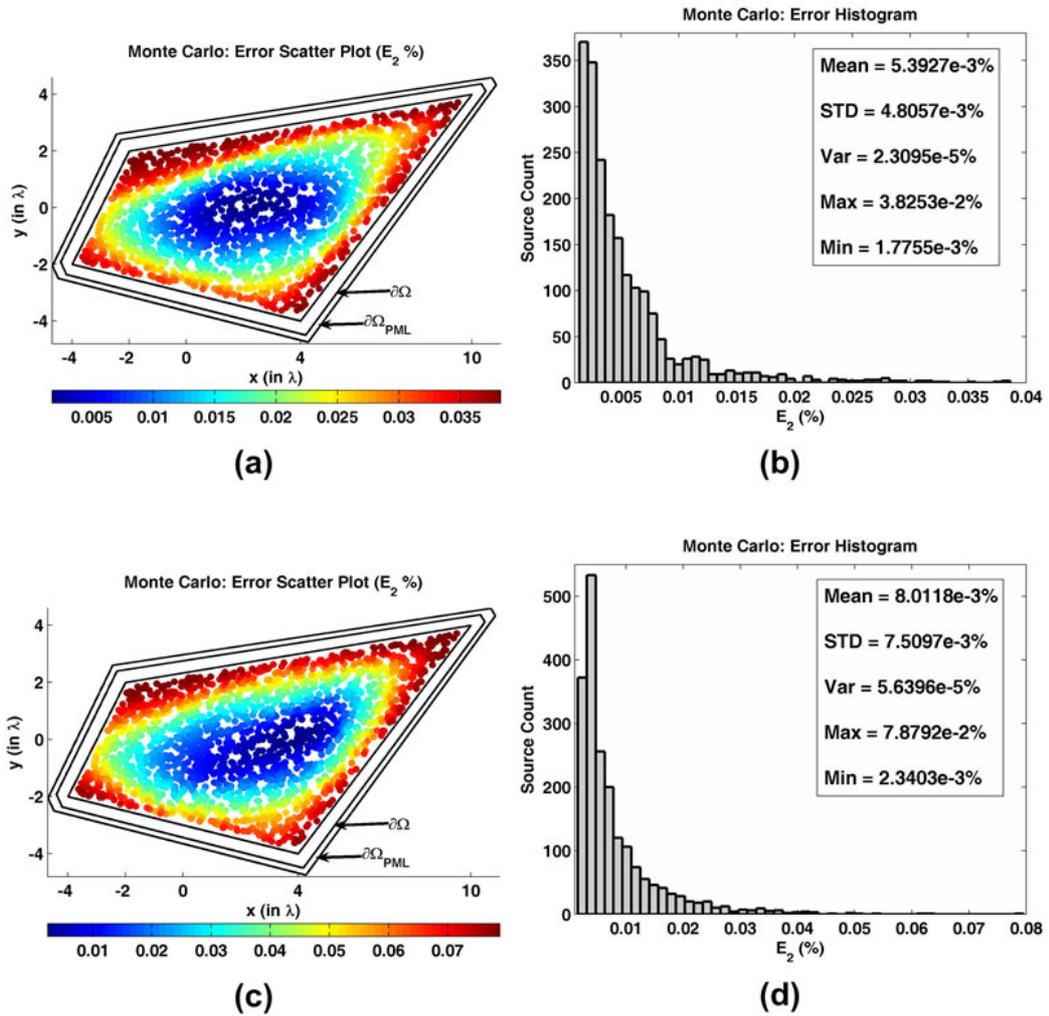


Figure 6.54. Error analysis for E_2 in the Monte Carlo Simulation of the radiation problem in quadrilateral domain: (a) Error scatter plot in the locally-conformal PML, (b) Error histogram and statistics in the locally-conformal PML, (c) Error scatter plot in the multi-center PML, (d) Error histogram and statistics in the multi-center PML.

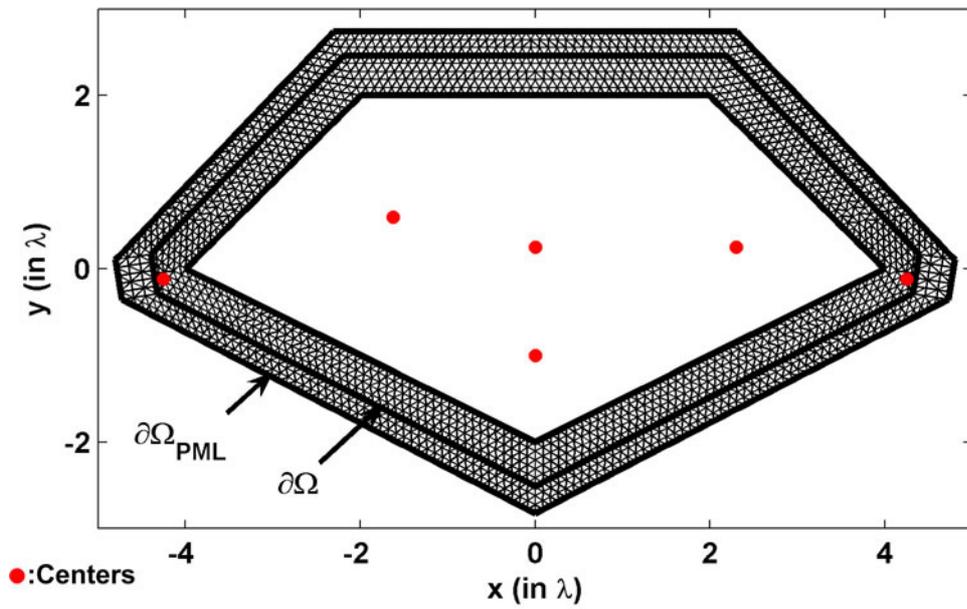


Figure 6.55. Mesh for pentagonal domain (vertices are shown by tick labels).

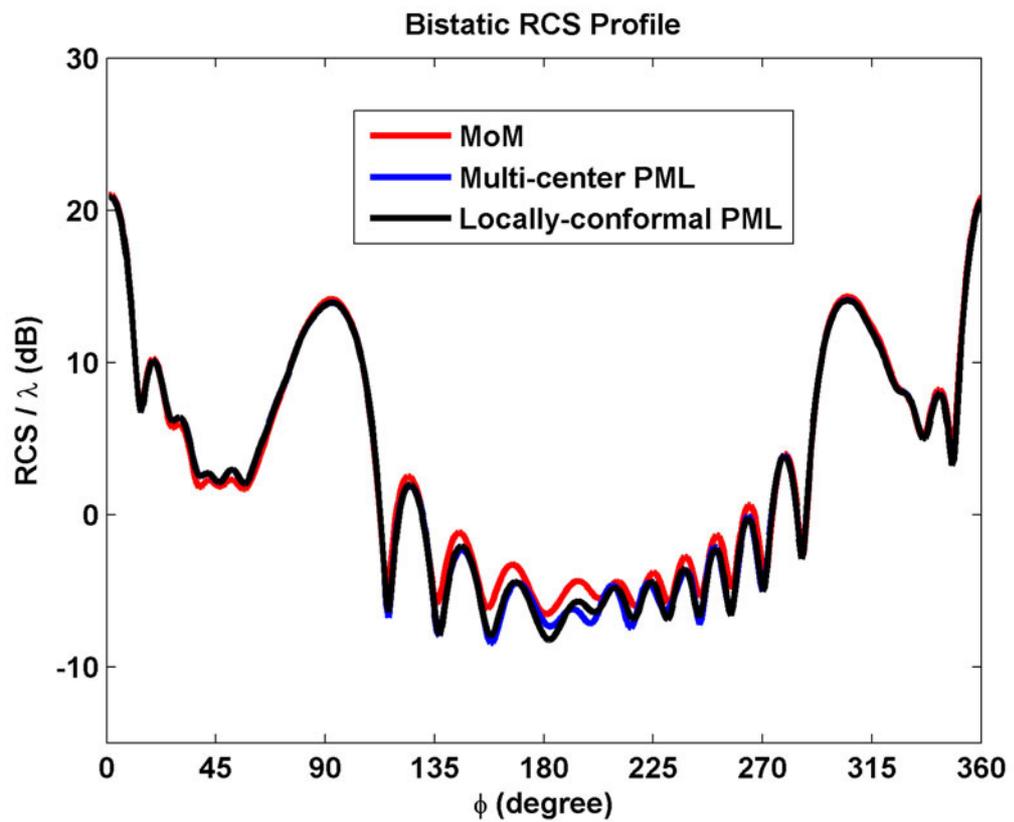


Figure 6.56. RCS profile for scattering from pentagonal PEC cylinder.

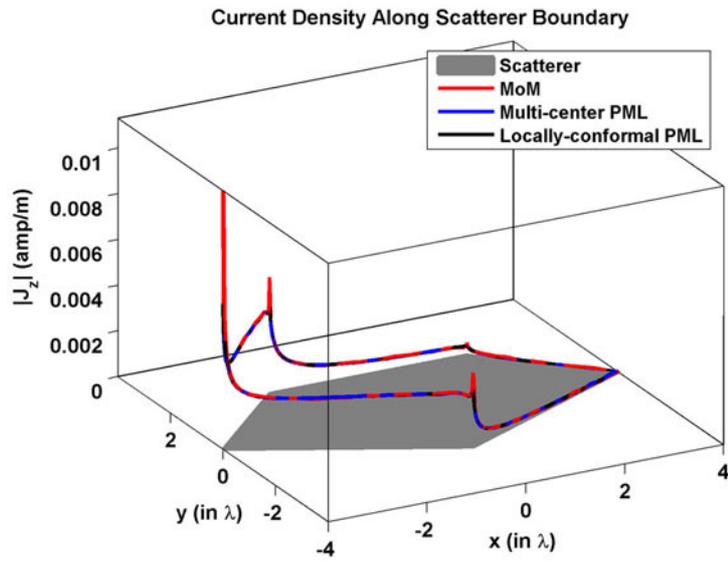


Figure 6.57. Magnitude of current density along pentagonal PEC cylinder.

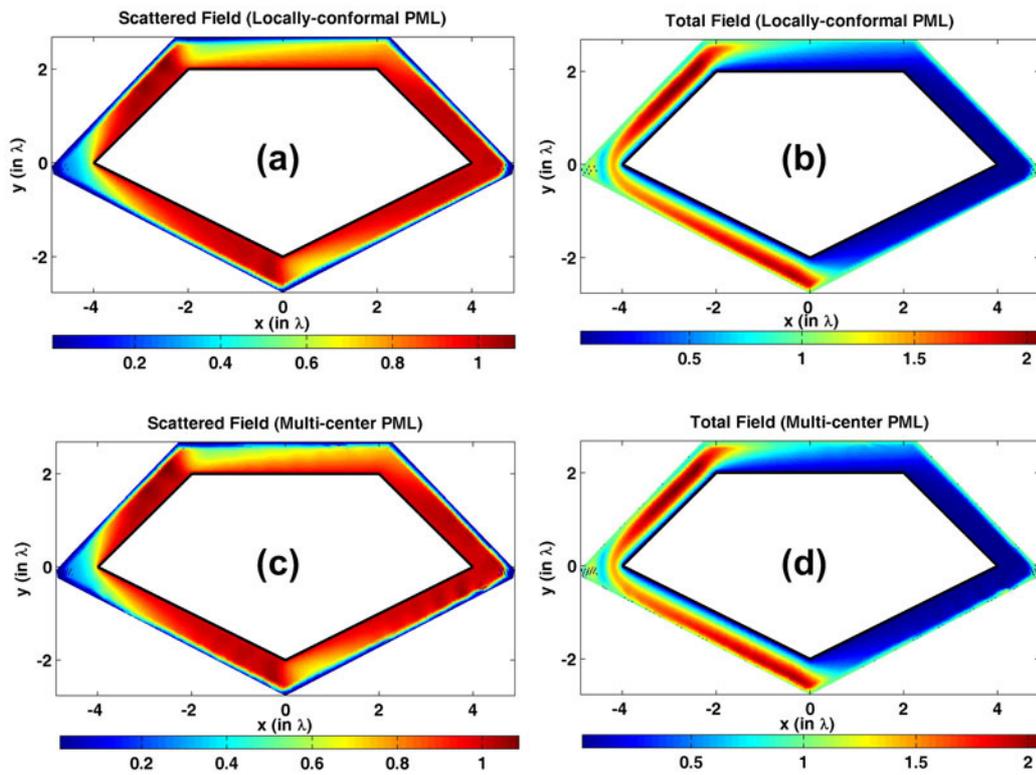


Figure 6.58. Field contours for pentagonal PEC cylinder (magnitude): (a-b) Scattered and total fields in the locally-conformal PML, (c-d) Scattered and total fields in the multi-center PML.

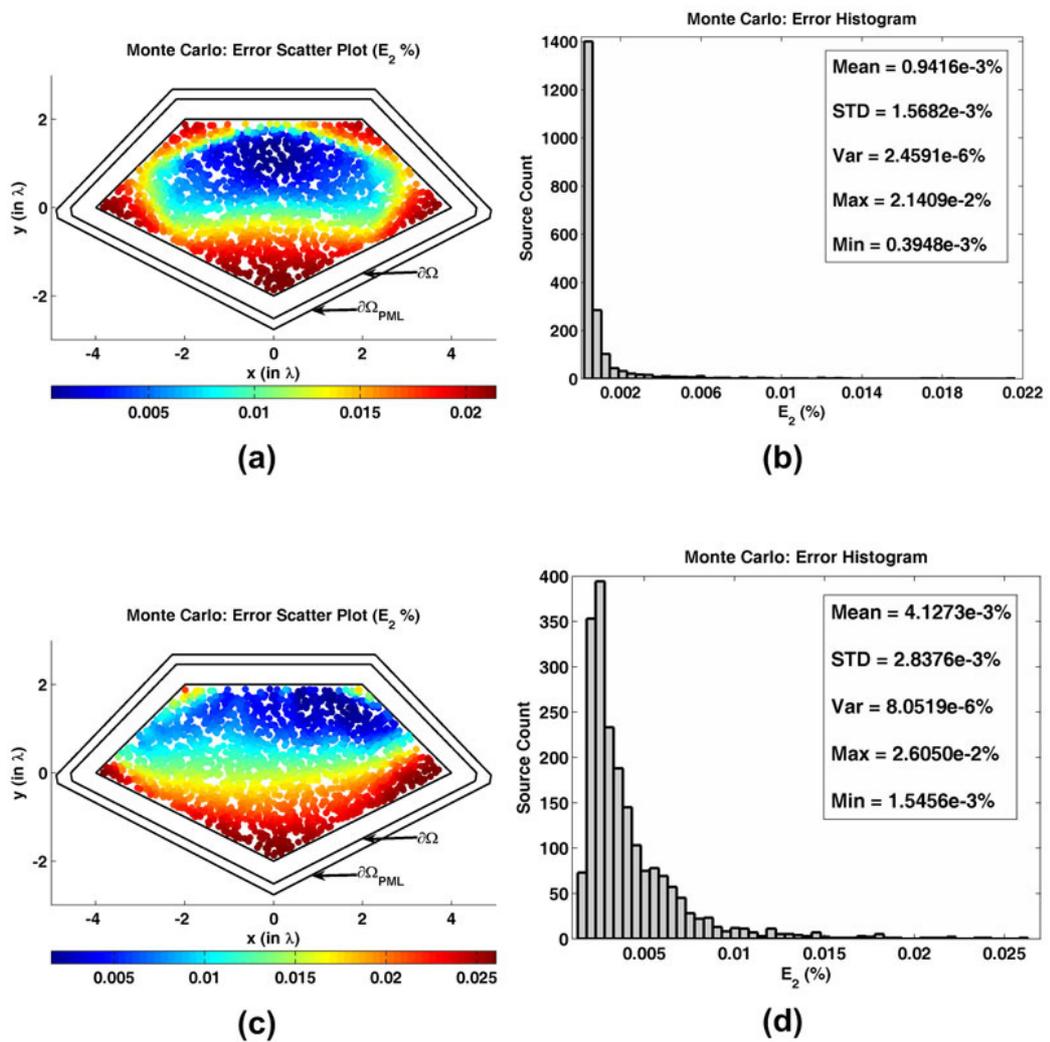


Figure 6.59. Error analysis for E_2 in the Monte Carlo Simulation of the radiation problem in pentagonal domain: (a) Error scatter plot in the locally-conformal PML, (b) Error histogram and statistics in the locally-conformal PML, (c) Error scatter plot in the multi-center PML, (d) Error histogram and statistics in the multi-center PML.

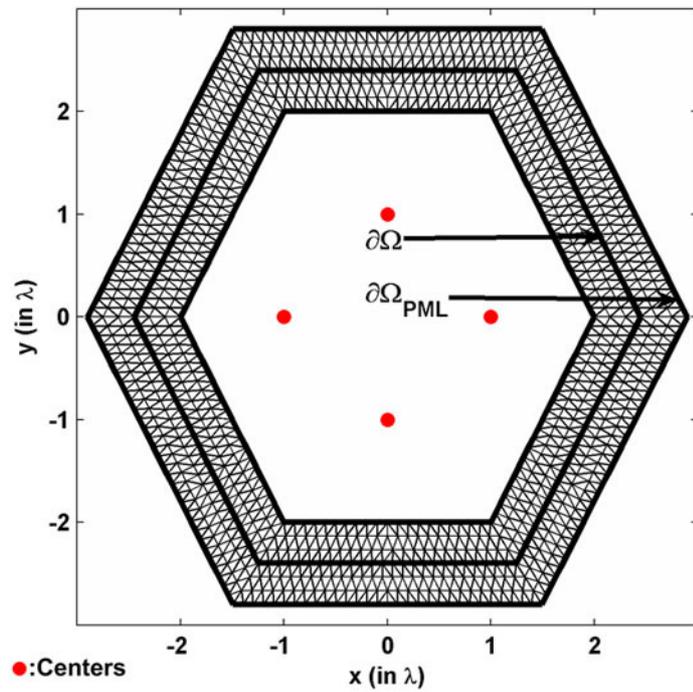


Figure 6.60. Mesh for hexagonal domain (vertices are shown by tick labels).

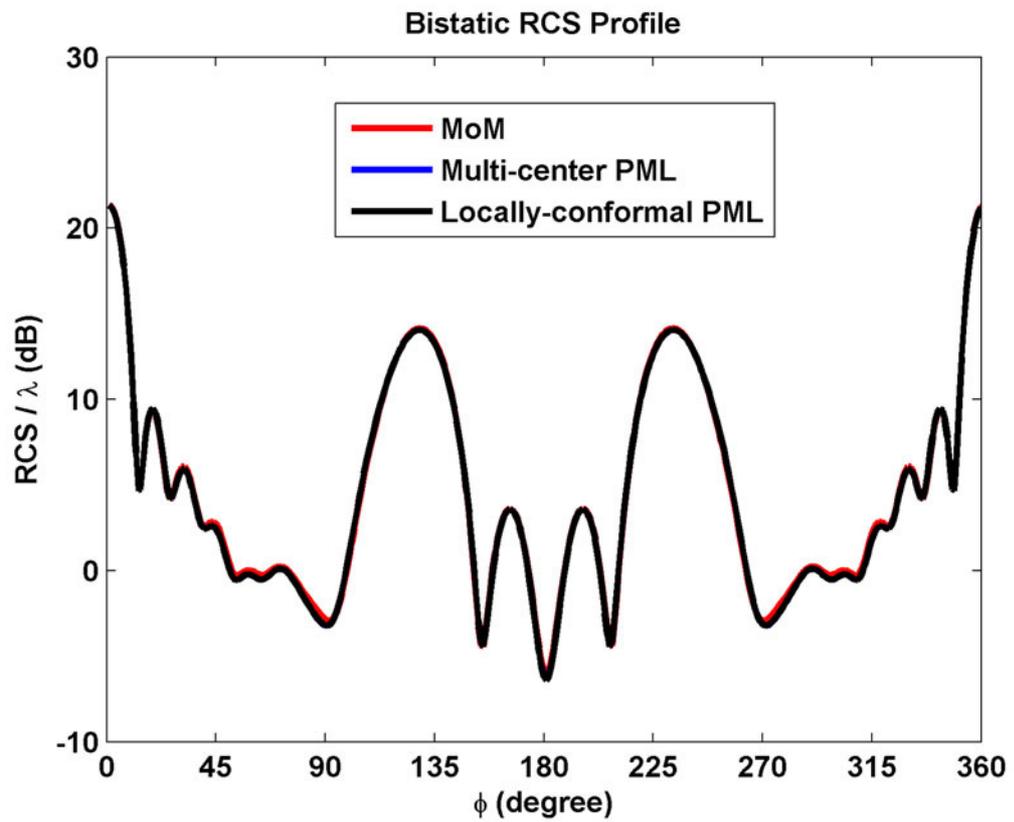


Figure 6.61. RCS profile for scattering from hexagonal PEC cylinder.

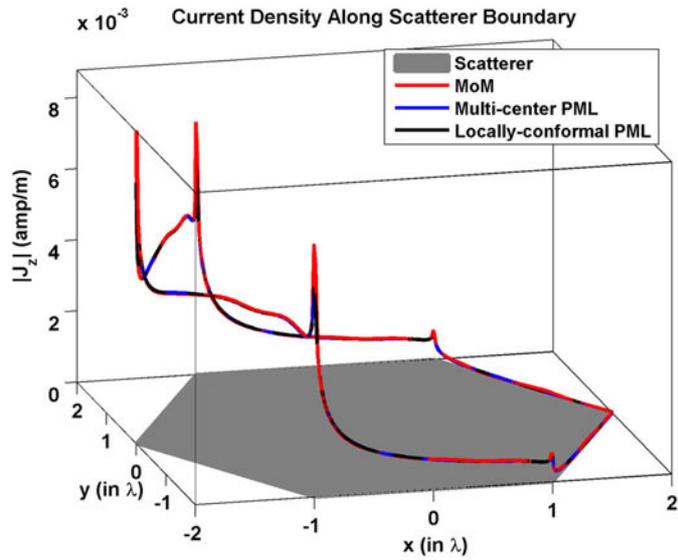


Figure 6.62. Magnitude of current density along hexagonal PEC cylinder.

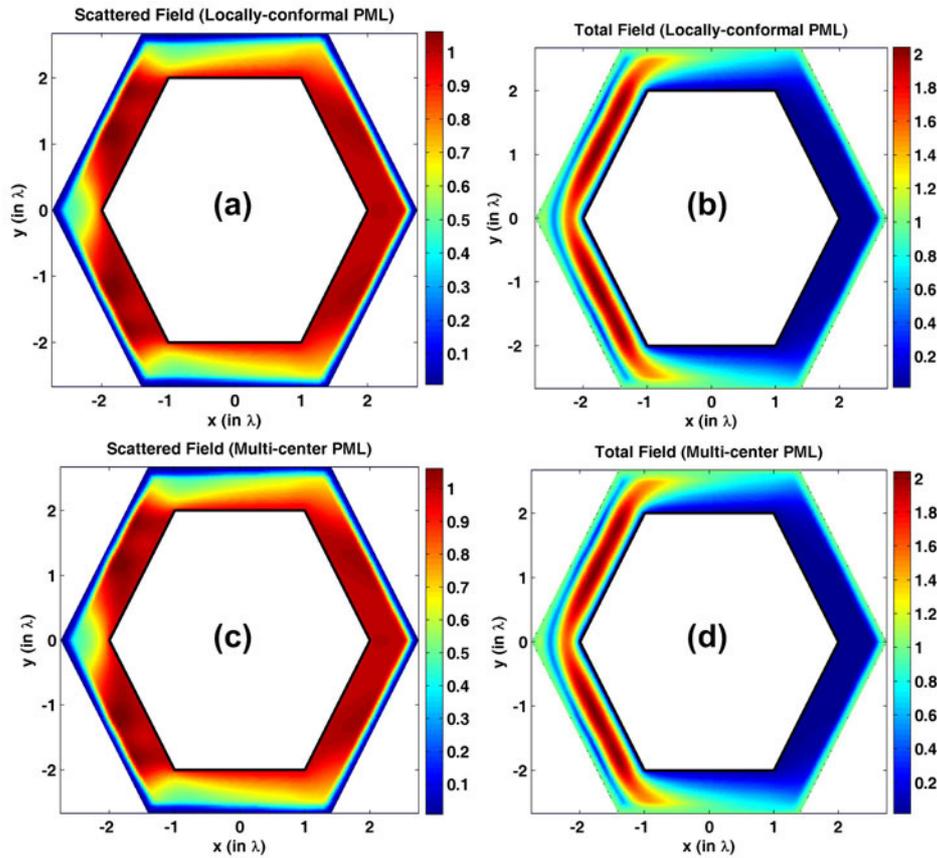


Figure 6.63. Field contours for hexagonal PEC cylinder (magnitude): (a-b) Scattered and total fields in the locally-conformal PML, (c-d) Scattered and total fields in the multi-center PML.

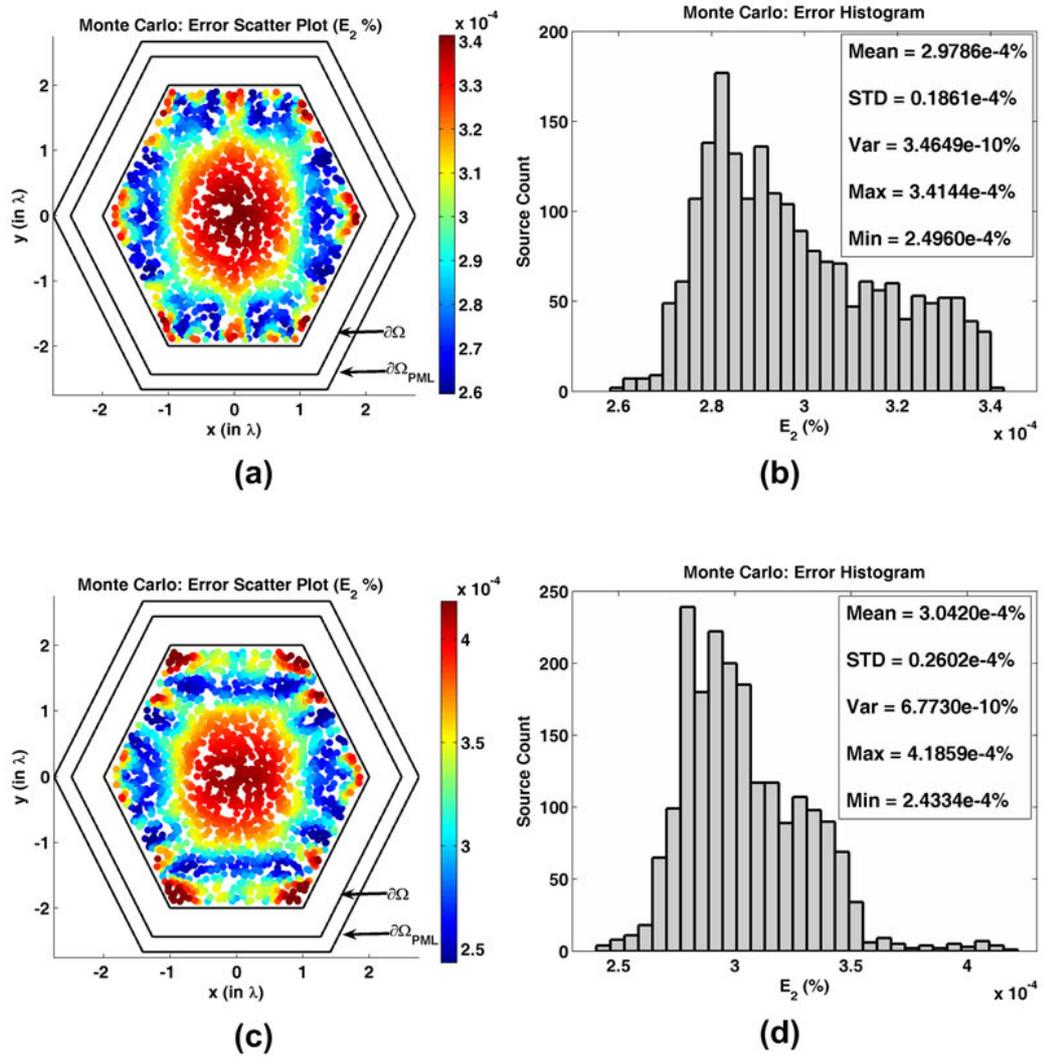


Figure 6.64. Error analysis for E_2 in the Monte Carlo Simulation of the radiation problem in hexagonal domain: (a) Error scatter plot in the locally-conformal PML, (b) Error histogram and statistics in the locally-conformal PML, (c) Error scatter plot in the multi-center PML, (d) Error histogram and statistics in the multi-center PML.

6.3. 3D Simulations

In this section, we test the locally-conformal PML and the multi-center PML techniques via 3D scatterers of various shapes. First, in Section 6.3.1, we report the results of some experiments to compare the two PML methods with each other and with the results in the literature. Then, in Section 6.3.2, we present miscellaneous examples which are implemented by the locally-conformal PML method.

The common parameters in all experiments are chosen as: k is 20π (i.e., the wavelength λ is 0.1 meter), m is 3, and α is $5k$. In addition, the PML thickness is approximately set to $\lambda/5$, and the edge size of each tetrahedral element in the mesh is approximately adjusted to the range between $\lambda/10$ - $\lambda/20$, depending on the size of the problem.

6.3.1. Comparison of the PML Methods

The 1st example is a PEC sphere whose radius is 1λ . We assume that the incident plane wave propagates along the positive z -direction (axial incidence), and the electric field is polarized along the x -direction (i.e., $\vec{E}^{\text{inc}} = \hat{a}_x \exp(-jkz)$). We plot the bistatic RCS profile of the PEC sphere in Figure 6.65. The mesh cross-section of the whole computational domain and the mesh of the scatterer are also shown in this figure. These meshes are just coarse meshes given for the purpose of visualization. The different gray-shades in the mesh cross-section correspond to the scatterer, free-space and PML regions, respectively. A similar approach is followed in the remaining examples to present the results. A single center is chosen at $(0, 0, 0)$ for the multi-center PML method. It is obvious that the multi-

center PML method with a single center at the center-of-mass point of the sphere is equivalent to the locally-conformal PML method.

The 2nd example is a dielectric sphere whose radius is 0.1λ and relative permittivity is 4. Assuming that the incident plane wave is $\vec{E}^{\text{inc}} = \hat{a}_x \exp(-jkz)$, we plot the scattered farfield profile in Figure 6.66. Similar to the first example, a single center is chosen at $(0, 0, 0)$ for the multi-center PML method.

The 3rd example is a PEC cube whose edge length is 0.755λ . We assume that the incident plane wave propagates along the positive z -direction (axial incidence), and the electric field is polarized along the y -direction (i.e., $\vec{E}^{\text{inc}} = \hat{a}_y \exp(-jkz)$). We plot the bistatic RCS profile of the PEC cube in Figure 6.67. A single center is chosen at $(0, 0, 0)$ for the multi-center PML method. We calculate the error E_1 as 0.2591 %.

The 4th example is a dielectric cube whose edge length is 0.2λ and relative permittivity is 4. Assuming that the incident plane wave is $\vec{E}^{\text{inc}} = \hat{a}_x \exp(-jkz)$, we plot the scattered farfield profile in Figure 6.68. Similar to the 3rd example, a single center is selected at $(0, 0, 0)$ for the multi-center PML method. We calculate the error E_1 as 0.4275 %.

The 5th example is a PEC prolate spheroid whose semi-major axis is 0.8λ and axial ratio is 2. We assume that the incident plane wave propagates along the negative z -direction, and the electric field is polarized along the y -direction (i.e., $\vec{E}^{\text{inc}} = \hat{a}_y \exp(jkz)$). We plot the bistatic RCS profile of the PEC spheroid in Figure 6.69. Three centers are chosen at $(0, 0, 0)$, $(0, 0, 0.8\lambda)$ and $(0, 0, -0.8\lambda)$ for the multi-center PML method. We calculate the error E_1 as 0.2403 %.

The 6th example is a PEC conesphere whose half-cone angle is 45° , base radius is 1.6λ and cone-height is 1.6λ . Assuming an axial (nose-on) plane wave incidence, we plot the bistatic RCS profile in Figure 6.70. A single center is selected at $(0, 0, 0)$ for the multi-center PML method. We calculate the error E_1 as 0.0784% ($\theta\theta$ polarization) and 0.0709% ($\phi\phi$ polarization).

The 7th example is a PEC ‘sharp’ conesphere whose half-cone angle is 10° , base radius is 0.45λ and cone-height is 2.65λ . Assuming an axial (nose-on) plane wave incidence, we plot the bistatic RCS profile in Figure 6.71. Since the cone angle is less than 90° , we slightly cut the PML region in order to satisfy the threshold criterion. We choose eight centers shown by the red dots in Figure 6.71 in the multi-center PML method. These centers are: $(0, 0, 0)$, $(0, 0, 0.5\lambda)$, $(0, 0, 1\lambda)$, $(0, 0, 1.5\lambda)$, $(0, 0, 2\lambda)$, $(0, 0, 2.7\lambda)$, $(0, 0, 3.1\lambda)$ and $(0, 0, 3.4\lambda)$. We calculate the error E_1 as 0.0475%.

The 8th example is a PEC finite circular cylinder whose radius is 0.3λ and height is 0.6λ . We plot the backscattering RCS profile in $\phi\phi$ polarization in Figure 6.72. We choose seven centers shown by the red dots in this figure for the multi-center PML method. These centers are: $(0, 0, 0)$, $(0.3\lambda, 0, 0)$, $(-0.3\lambda, 0, 0)$, $(0, 0.3\lambda, 0)$, $(0, -0.3\lambda, 0)$, $(0, 0, 0.3\lambda)$ and $(0, 0, -0.3\lambda)$. We calculate the error E_1 as 0.0486%.

6.3.2. Miscellaneous Examples based on the Locally-Conformal PML Method

The 1st example is a PEC finite cone with circular base, whose half-cone angle is 20° . Assuming an axial (nose-on) plane wave incidence, we plot the monostatic RCS profile of the cone in Figure 6.73 as a function of ka where a is the base radius of the cone.

The 2nd example is a PEC ogive, whose nose angle is 75° . Assuming an axial (nose-on) plane wave incidence, we plot the monostatic RCS profile of the ogive in Figure 6.74 as a function of the maximum diameter of the ogive in wavelengths.

The 3rd example is a PEC cube. We assume that the incident plane wave propagates along the positive z -direction (axial incidence), and the electric field is polarized along the y -direction (i.e., $\vec{E}^{\text{inc}} = \hat{a}_y \exp(-jkz)$). We plot the monostatic RCS profile of the cube in Figure 6.75 as a function of the edge length in wavelengths.

The 4th example is an electrically small PEC octahedron (i.e., two pyramids sharing a square base), whose edge length is 0.07λ , and half-height is 0.05λ . The edge size of each tetrahedral element in the mesh is approximately adjusted to $\lambda/100$ in this example. We assume that the incident plane wave propagates along the negative z -direction, and the electric field is polarized along x -direction (i.e., $\vec{E}^{\text{inc}} = \hat{a}_x \exp(jkz)$). We plot the bistatic RCS profile of the PEC octahedron in Figure 6.76.

The 5th example is a PEC square plate. We assume that the incident plane wave propagates along the positive z -direction (axial incidence), and the electric field is polarized along the x -direction (i.e., $\vec{E}^{\text{inc}} = \hat{a}_x \exp(-jkz)$). We plot the monostatic RCS profile of the plate in Figure 6.77 as a function of the edge length in wavelengths.

The 6th example is a PEC circular plate whose radius is 1λ . We plot the backscattering RCS profile in $\phi\phi$ polarization in Figure 6.78.

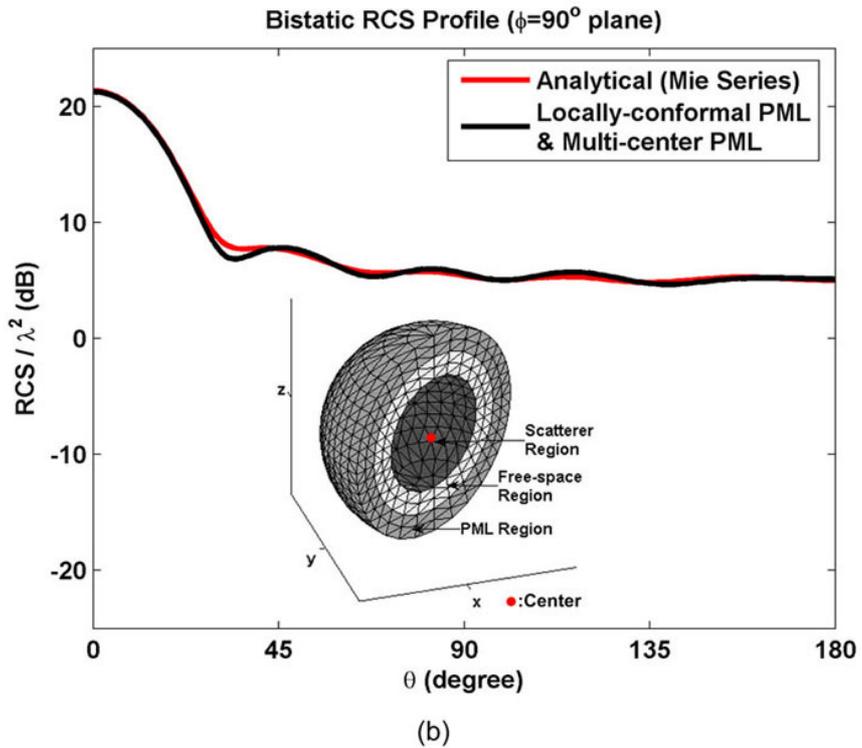
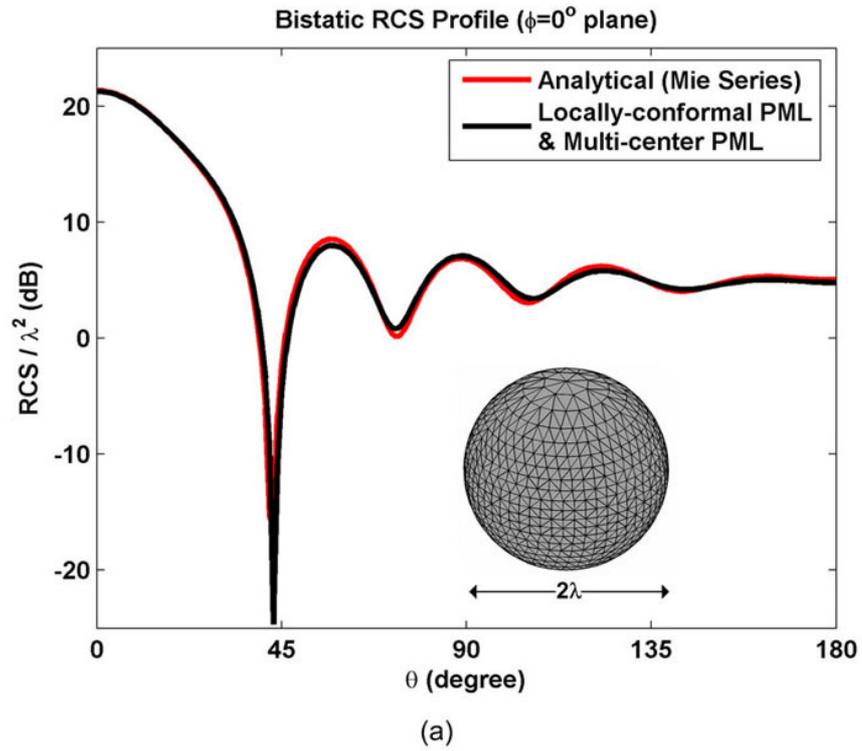
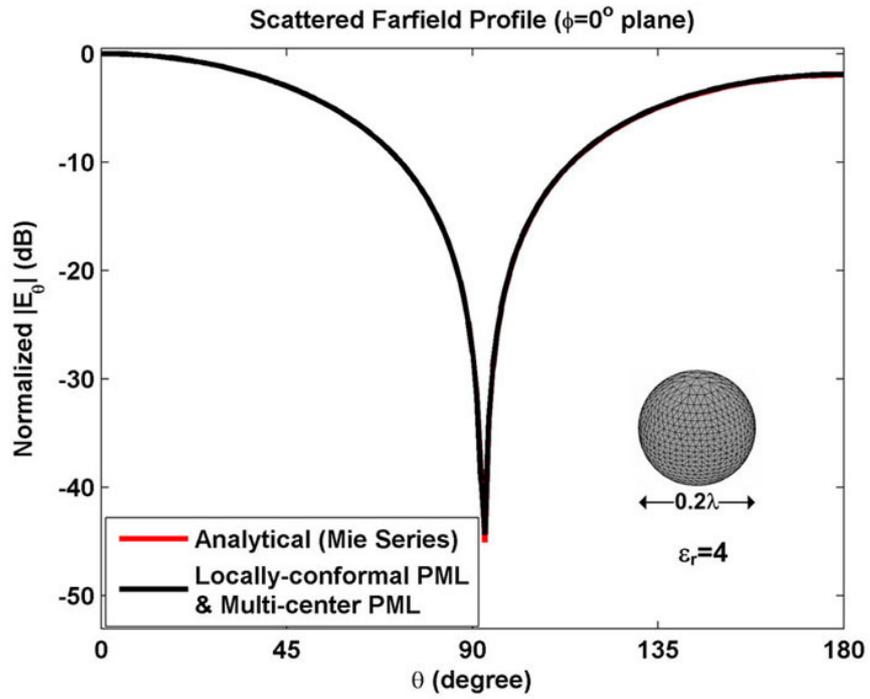
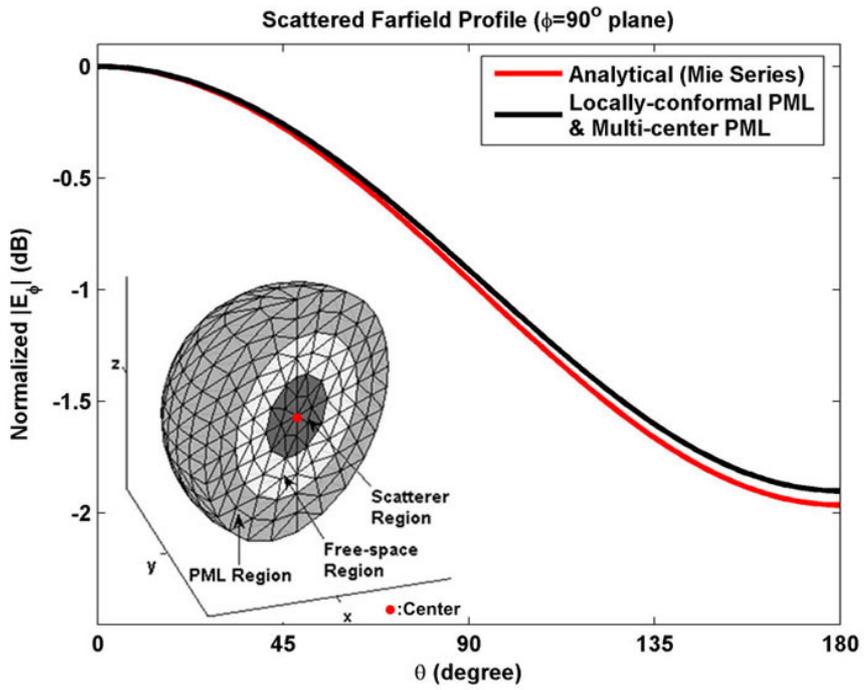


Figure 6.65. Bistatic RCS profile of PEC sphere: (a) $\phi=0^\circ$ plane, (b) $\phi=90^\circ$ plane.



(a)



(b)

Figure 6.66. Scattered farfield profile of dielectric sphere: (a) $\phi=0^\circ$ plane, (b) $\phi=90^\circ$ plane.

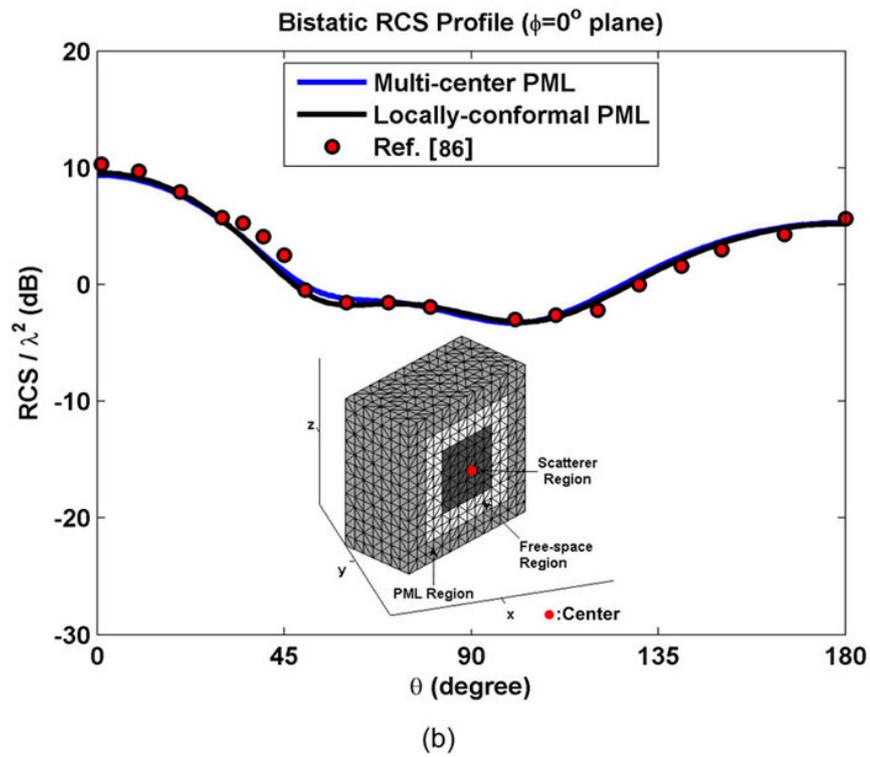
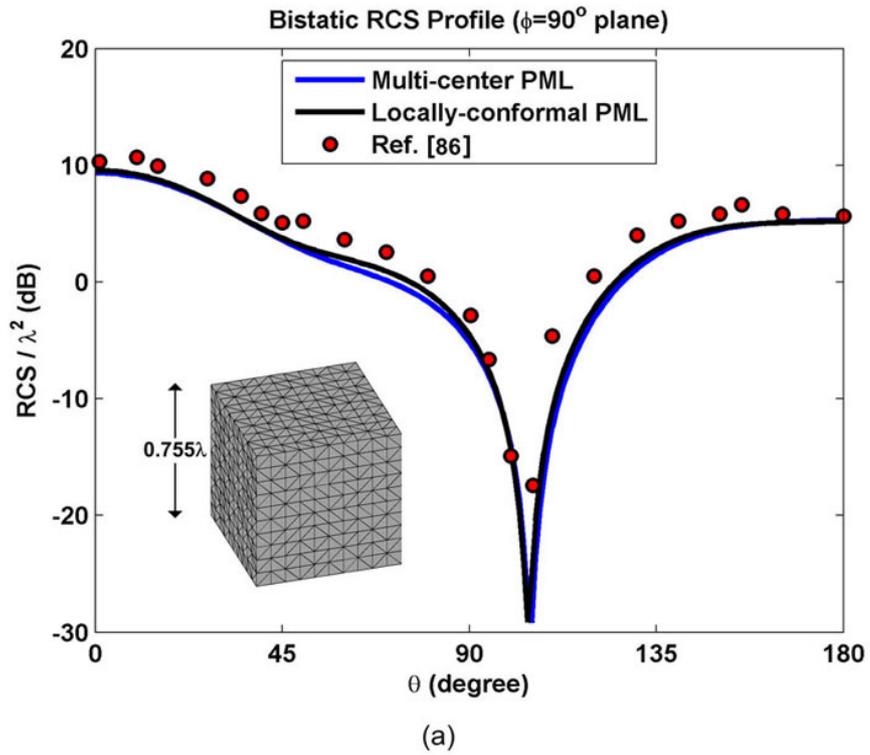


Figure 6.67. Bistatic RCS profile of PEC cube: (a) $\phi=90^\circ$ plane, (b) $\phi=0^\circ$ plane.

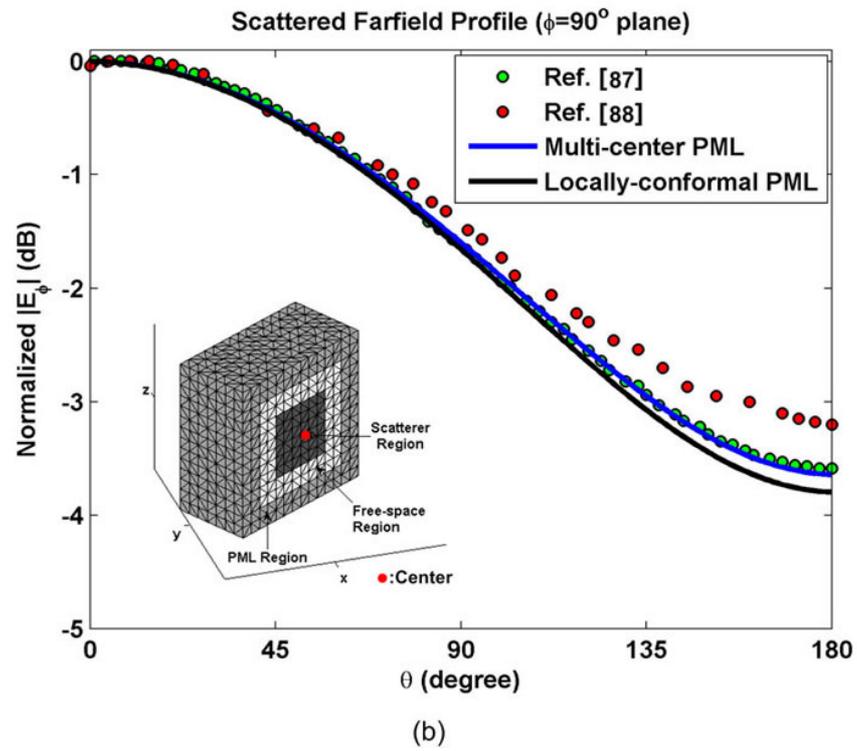
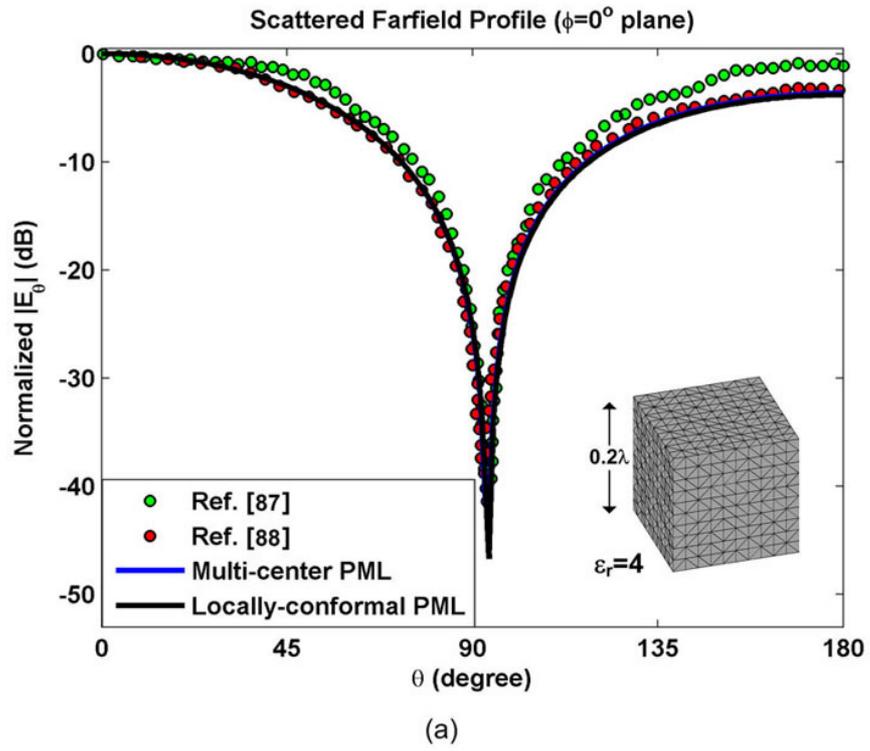


Figure 6.68. Scattered farfield profile of dielectric cube: (a) $\phi=0^\circ$ plane, (b) $\phi=90^\circ$ plane.

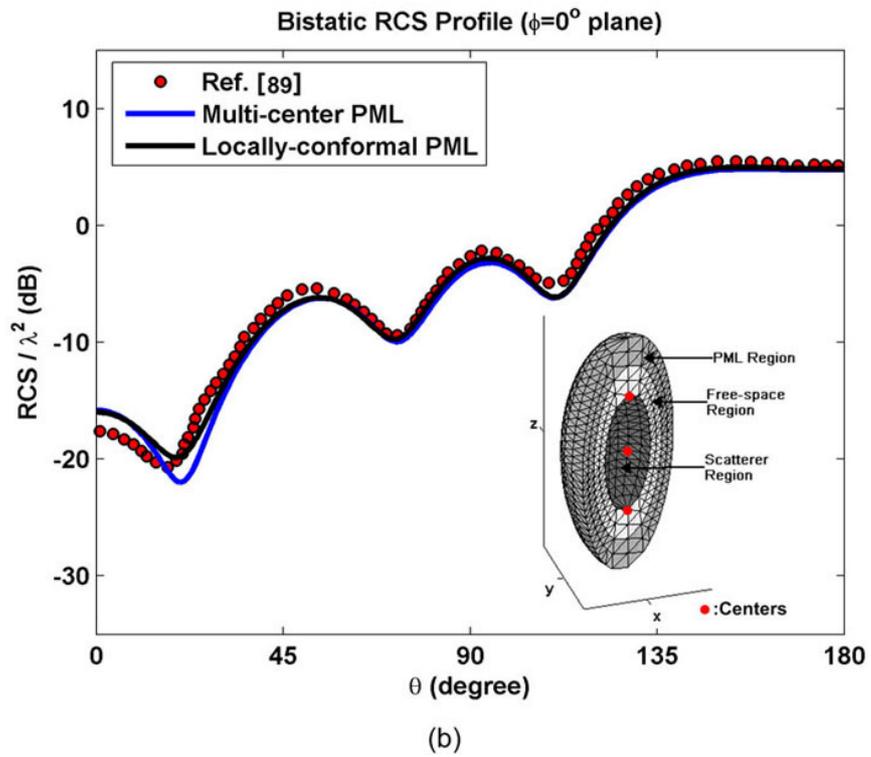
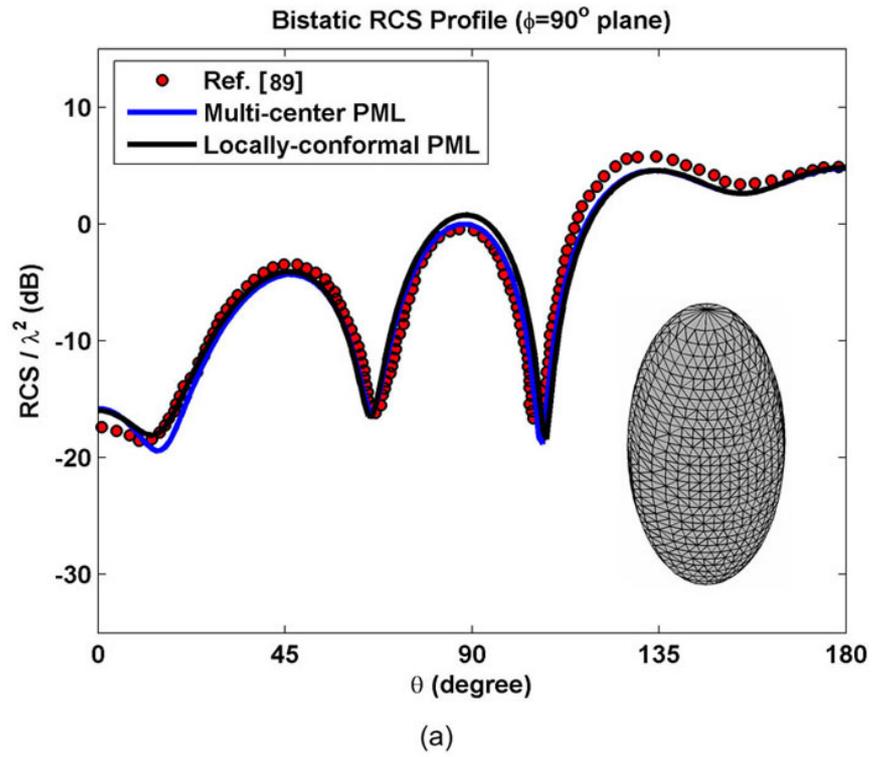


Figure 6.69. Bistatic RCS profile of PEC prolate spheroid (semi-major axis is 0.8λ , axial ratio is 2): (a) $\phi=90^\circ$ plane, (b) $\phi=0^\circ$ plane.

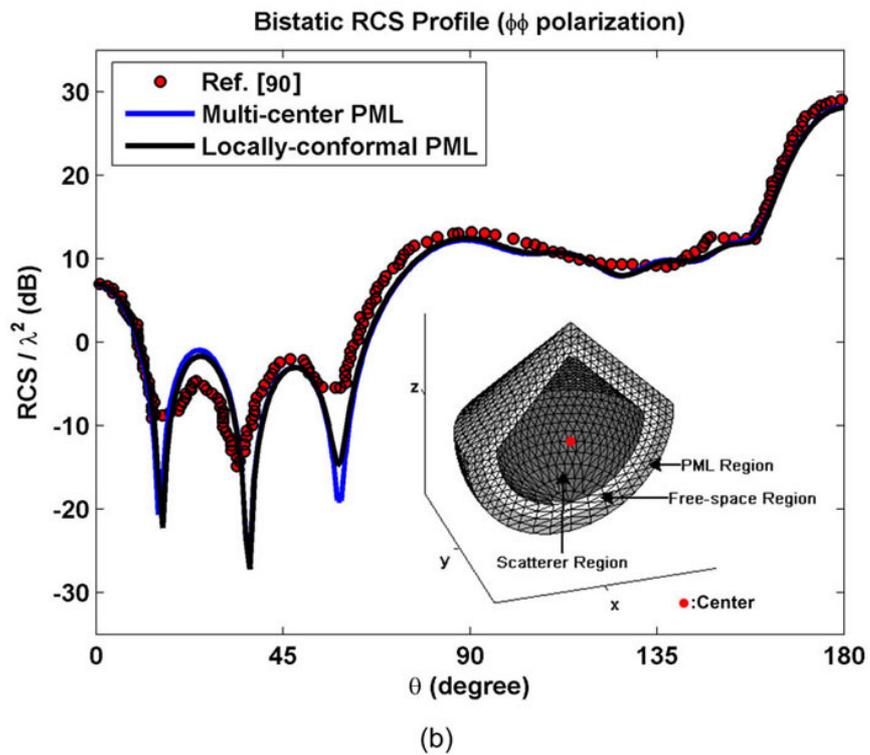
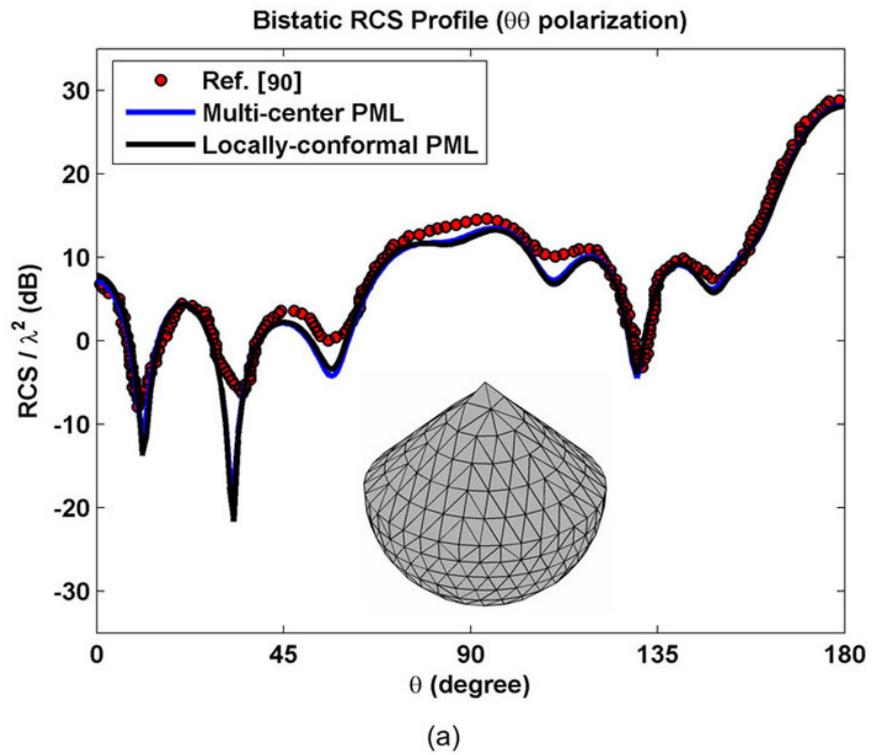


Figure 6.70. Bistatic RCS profile of PEC conesphere (radius is 1.6λ , nose-angle 90°): (a) $\theta\theta$ polarization, (b) $\phi\phi$ polarization.

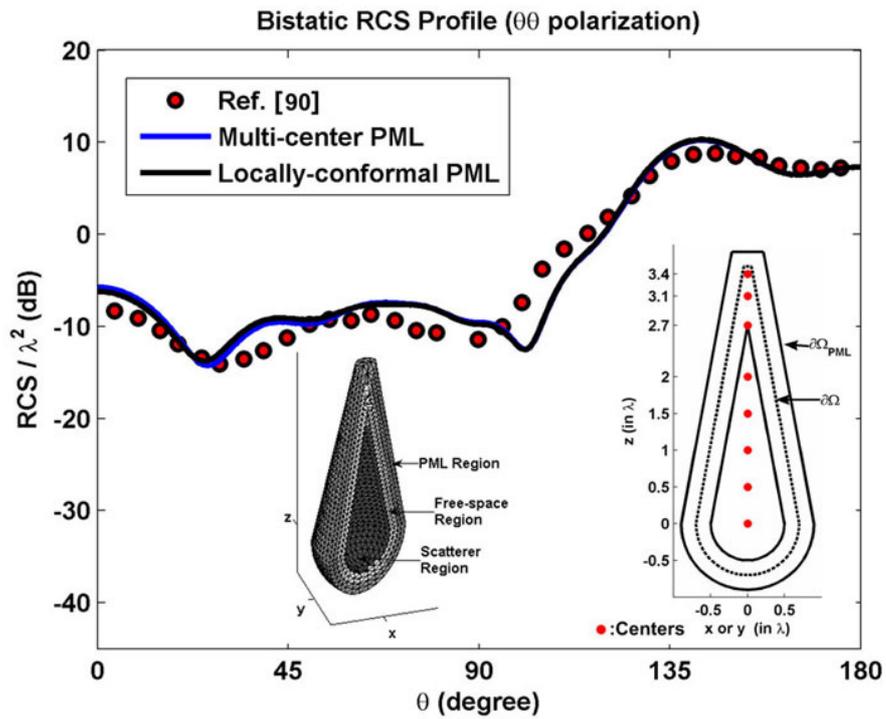


Figure 6.71. Bistatic RCS profile of PEC sharp conesphere ($\theta\theta$ polarization).

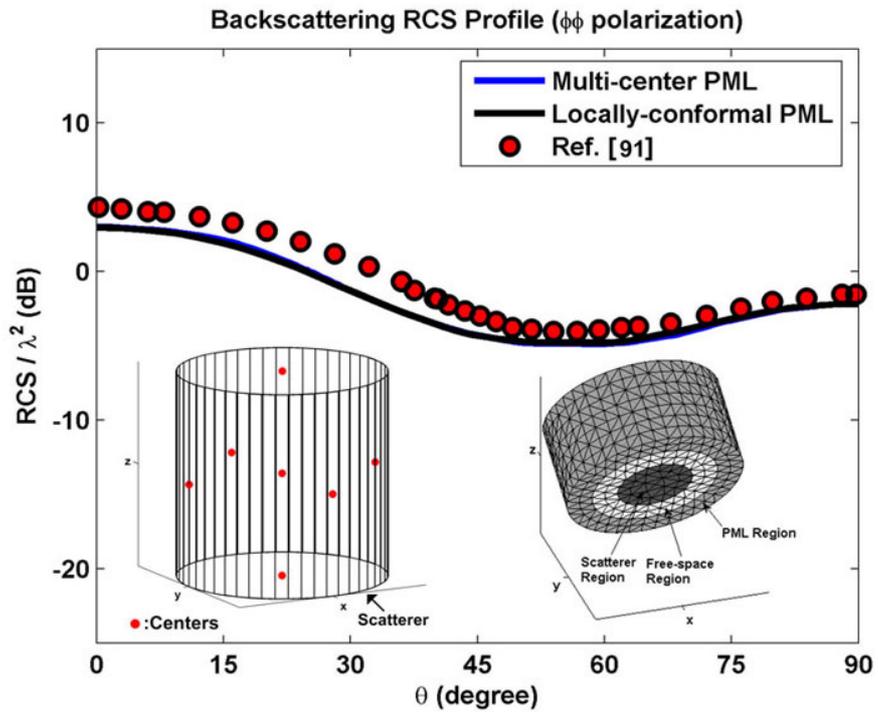


Figure 6.72. Backscattering RCS profile of PEC finite circular cylinder ($\phi\phi$ polarization) (radius 0.3λ , height 0.6λ).

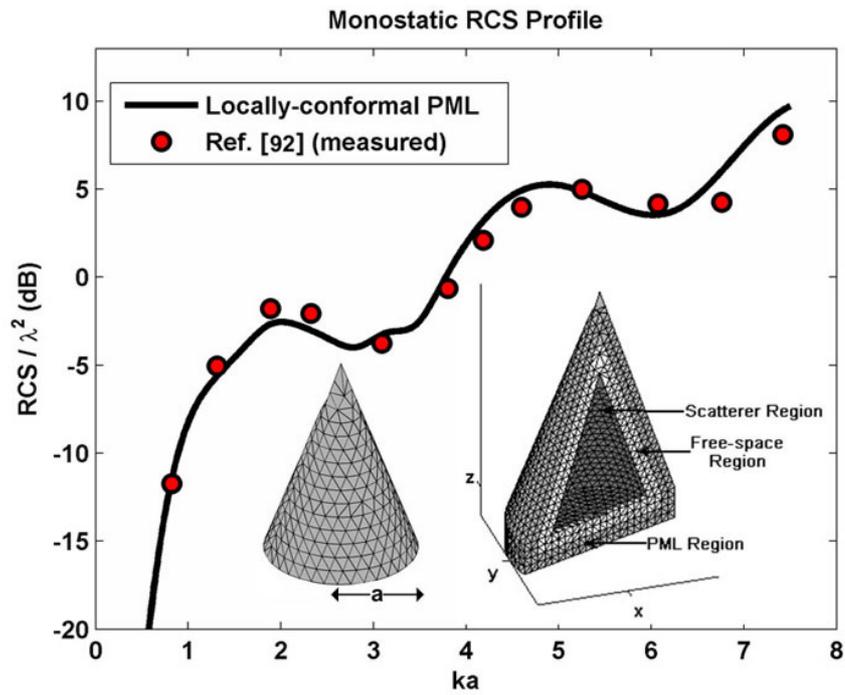


Figure 6.73. Monostatic RCS profile of PEC cone as a function of ka where a is the base radius of the cone (half-cone angle 20°).

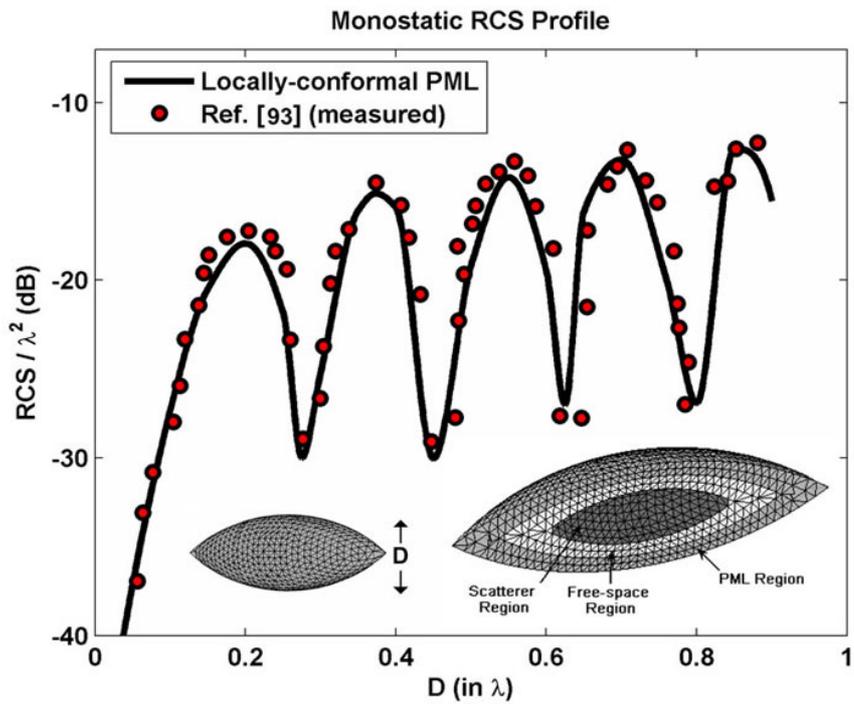


Figure 6.74. Monostatic RCS profile of PEC ogive as a function of the maximum diameter in wavelengths (nose angle 75°).

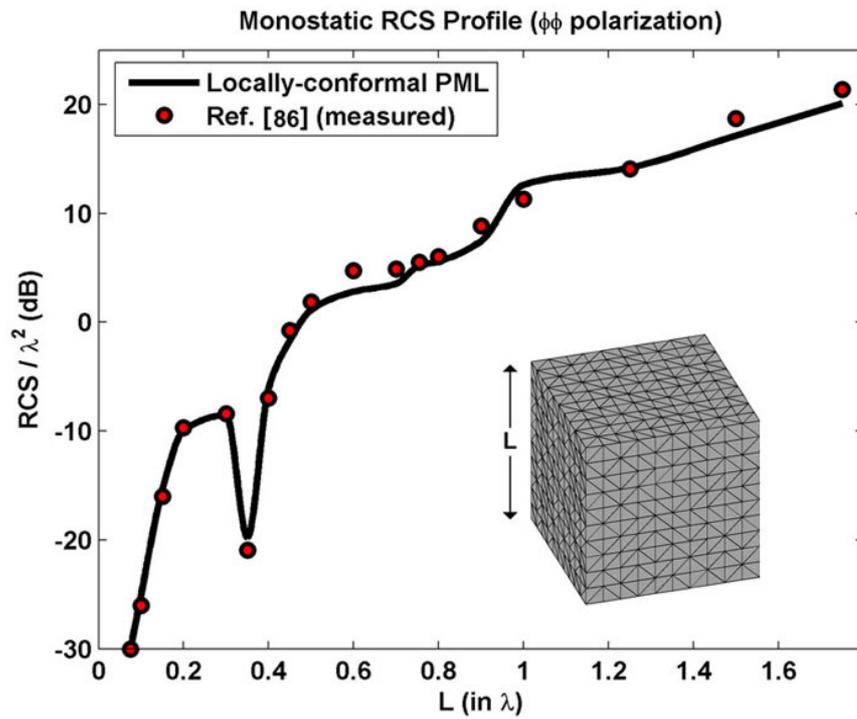


Figure 6.75. Monostatic RCS profile of PEC cube as a function of edge length in wavelengths.

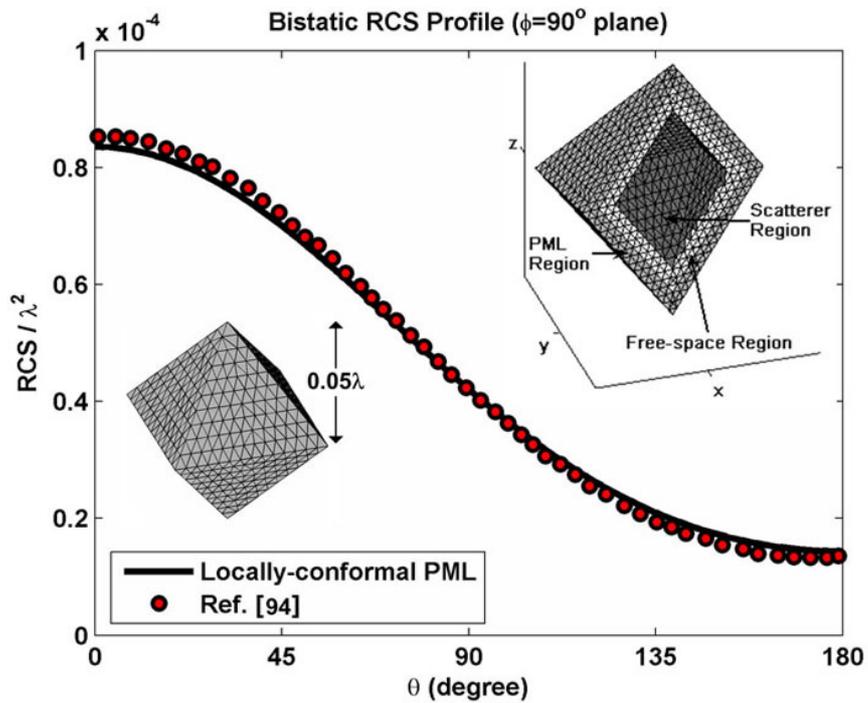


Figure 6.76. Bistatic RCS profile of PEC octahedron in $\phi=90^\circ$ plane.

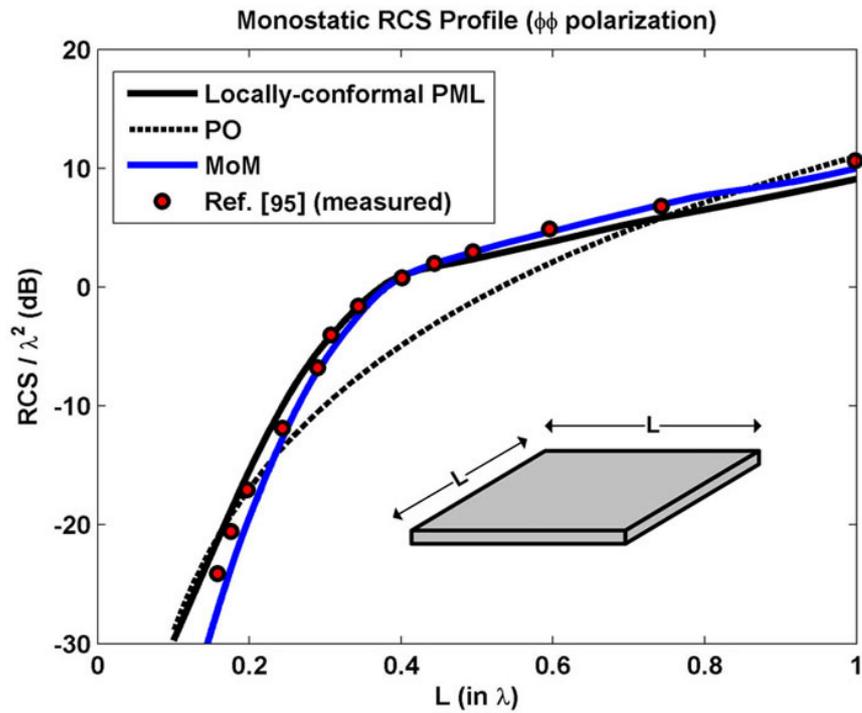


Figure 6.77. Monostatic RCS profile of PEC square plate as a function of edge length in wavelengths.

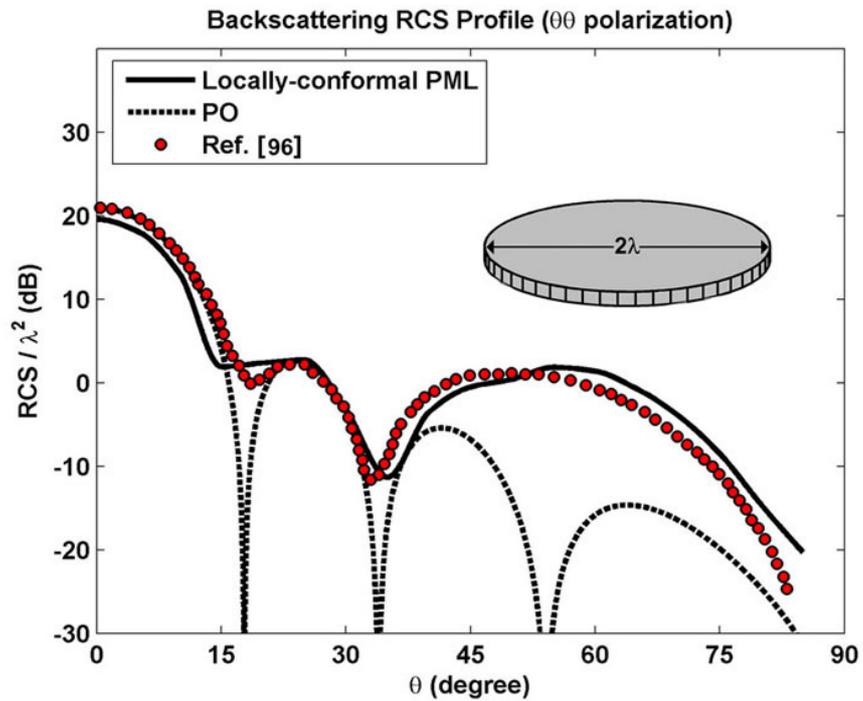


Figure 6.78. Backscattering RCS profile of PEC circular plate ($\theta\theta$ polarization).

CHAPTER 7

NUMERICAL EXPERIMENTS (FB-DDM)

In this chapter, we test the performance of the Forward-Backward DDM (FB-DDM) algorithm by means of some 2D and 3D electromagnetic scattering problems.

In order to measure the performance of the FB-DDM algorithm, we introduce two different kinds of mean-square error criteria. The first error criterion is defined as

$$E_1 = \frac{\sum_{\Omega_c} |\vec{E}^{\text{DDM}} - \vec{E}^{\text{whole}}|^2}{\sum_{\Omega_c} |\vec{E}^{\text{whole}}|^2} \quad (\times 100) \quad (7.1)$$

where \vec{E}^{DDM} and \vec{E}^{whole} are the scattered field values in the original computational domain Ω_c ($\Omega_c = \Omega_{\text{FS}} \cup \Omega_{\text{PML}}$) calculated by the FB-DDM algorithm and the original whole-domain FEM, respectively.

The second error criterion is employed only in the case of multiple forward-backward iterations, and is defined as

$$E_2 = \frac{\sum_{\Omega_c} |\vec{E}^n - \vec{E}^{n-1}|^2}{\sum_{\Omega_c} |\vec{E}^{n-1}|^2} \quad (\times 100) \quad (7.2)$$

where \vec{E}^n and \vec{E}^{n-1} are the scattered field values in the original computational domain Ω_c calculated by the FB-DDM algorithm at the n^{th} and $(n-1)^{\text{th}}$ iterations,

respectively. The second error criterion also provides a proper way of finding out when to stop the algorithm. In other words, when E_2 is less than a certain value (i.e., $E_2 < 10^{-2}$ based on our numerical observations), we assume that convergence is achieved since the scattered field values inside the whole domain do not change significantly for smaller values of this constant. However, if we need to consider only the RCS values, we have observed that $E_2 < 10^{-1}$ is sufficient for the convergence of the algorithm. This is because the far-field calculation which is performed in the RCS calculation has some smoothing effect, and may reflect a reduction in the magnitude of errors present in the near field terms.

The performance of the algorithm is tested in terms of the RCS calculations of objects, and is compared with the results of the whole-domain FEM solution. In addition, we validate the RCS results with those of a standard MoM code, a standard PO code, analytical calculations, or those in the literature, whichever applicable.

7.1. 2D Simulations

In this section, we test the FB-DDM algorithm in electromagnetic scattering problems using infinitely-long PEC cylinders. In all examples, the locally-conformal PML method is implemented using the parameters $\alpha = 7k$ and $m = 3$, and the approximate PML thickness is chosen as $\lambda/4$ (for both existing PML region and additional PML region). The wavenumber k is set to 20π (i.e., the wavelength λ is 0.1 meter). The element size is approximately set to $\lambda/40$. Furthermore, the incident plane wave is assumed to be in the form of $\vec{E}^{inc} = \hat{a}_z \exp\left[jk(x \cos \varphi^{inc} + y \sin \varphi^{inc})\right]$ where φ^{inc} is the angle of incidence with respect to the x -axis. The first four examples deals with some smooth

geometries where the first forward iteration is enough for convergence. The next three examples consider the problem of scattering from challenging cylinders which require some number of forward-backward iterations.

The 1st example is a benchmark scattering problem where a plane wave ($\varphi^{inc} = 180^\circ$) is incident to an infinitely-long PEC cylinder of circular cross-section whose diameter is 30λ . The decomposition of the original (whole) problem into N number of subdomains is illustrated in Figure 7.1. We implement the FB-DDM algorithm in just the first forward iteration, and we tabulate the error values E_1 for different number of subdomains in Table 7.1. We also plot the bistatic RCS profiles of all cases in Figure 7.2 by comparing them with the whole-domain FEM solution and the analytical Mie series solution [84]. In this plot, the results of the whole-domain FEM and the results of all FB-DDM cases coincide almost exactly, and they all have some tiny discrepancies when compared to the results of the Mie series solution. These examples demonstrate that a single forward iteration is sufficient to get reliable results with a suitable partitioning scheme, irrespective of the number of subdomains. However, it is obvious that as the number of subdomains increases, the error also increases due to the effect of the cumulative error as we sweep through the subdomains. In spite of this increase, the error values remain at an acceptable level.

Considering again the problem of scattering from a circular cylinder, we show the importance of the order and the positions of the subdomains by varying the direction of incidence of the plane wave while keeping the positions and numberings of the subdomains fixed in all cases (see Figure 7.3). We decompose the whole domain into two subdomains, and then we run the FB-DDM algorithm in just the first forward iteration. We tabulate the error values E_1 for different cases in Table 7.2. As expected, the case in Figure 7.3(a) where the angle of incidence is 180° shows the optimal choice for the subdomains and provides the best result for the analysis. This example demonstrates the importance of the

problem physics on the performance of the algorithm, and proves our claim that the first subdomain should be placed in the most illuminated part of the original domain in order to get an acceptably accurate result in the first forward iteration. Thus, all cases except the optimal case in Figure 7.3(a) need further forward-backward iterations to refine the solution.

The 2nd example is a scattering problem where a plane wave ($\varphi^{inc} = 180^\circ$) is incident to an infinitely-long PEC cylinder of square cross-section whose edge length is 20λ . The decomposition of the original problem into N number of subdomains is illustrated in Figure 7.4. For the first forward iteration in the FB-DDM algorithm, we tabulate the error values E_1 for different number of subdomains in Table 7.3. We also plot the bistatic RCS profiles of all cases in Figure 7.5. All FB-DDM results and the whole FEM results are almost the same, and have some small deviations from the MoM results. Due to the absence of reflections, a single forward sweep is sufficient to yield an accurate result in this case.

The 3rd example is a scattering problem where a plane wave ($\varphi^{inc} = 0^\circ$) is incident to an infinitely-long PEC cylinder of elliptical cross-section whose major axis is 40λ and axial ratio is 2. The original problem is decomposed into 5 subdomains (see the inset in Figure 7.6). For the first forward iteration in the FB-DDM algorithm, we calculate the error value E_1 as 1.4×10^{-3} %, and we plot the bistatic RCS profile in Figure 7.6.

The 4th example is a scattering problem where a plane wave ($\varphi^{inc} = 0^\circ$) is incident to an infinitely-long $16\lambda \times 4\lambda$ rectangular PEC cylinder. The original problem is decomposed into 4 subdomains (see the inset in Figure 7.7). For the first forward iteration in the FB-DDM algorithm, we calculate the error value E_1 as 3.1×10^{-4} %, and we plot the bistatic RCS profile in Figure 7.7.

The 5th example is a scattering problem where a plane wave ($\varphi^{inc} = 0^\circ$) is incident to a T-shaped cylinder. The decomposition of the original problem into 4 subdomains is illustrated in Figure 7.8. We plot the error values E_1 and E_2 versus the number of iterations (i.e., convergence profile) in Figure 7.9. In this plot, a single iteration denotes a single forward-backward iteration. In terms of the scattered field values inside the whole domain, the algorithm converges at the 15th iteration. In terms of the RCS values, the algorithm converges at the 7th iteration. We also plot the bistatic RCS profiles in Figure 7.10, and the scattered and total field contours corresponding to the first forward iteration, 7th and 15th iterations of the FB-DDM algorithm, as well as the whole-domain FEM in Figure 7.11. These contours serve as an indication of a large amount of field activity in the illuminated part of the whole domain.

The 6th example is a scattering problem where a plane wave ($\varphi^{inc} = 0^\circ$) is incident to a H-shaped cylinder. The decomposition of the original problem into 4 subdomains is illustrated in Figure 7.12. In terms of the scattered field values inside the whole domain, the algorithm converges at the 8th iteration. In terms of the RCS values, the algorithm converges in the first forward iteration. We plot the convergence profile in Figure 7.13. We also plot the bistatic RCS profiles in Figure 7.14, and the scattered and total field contours corresponding to the first forward iteration, 4th and 8th iterations of the FB-DDM algorithm, as well as the whole-domain FEM in Figure 7.15. We can conclude that the algorithm in this example converges faster than the one in the 5th example, as expected (refer to the discussion in Section 5.1.3).

The 7th example is a scattering problem where a plane wave ($\varphi^{inc} = 0^\circ$) is incident to a U-shaped cylinder (see Figure 7.16(a)), which is described in Section 5.1.3 as a special problem. In Case 1, we consider the partitioning scheme in Figure 7.16(b), where the first subdomain includes the inner part of

the cylinder. For the first forward iteration in the FB-DDM algorithm, we calculate the error value E_1 as $1.1 \times 10^{-3} \%$. In Case 2, we consider the partitioning scheme in Figure 7.16(c), where the inner part of the cylinder is decomposed into two subdomains. Only for the inner part of the domain, we plot the convergence profile in Figure 7.17. In this plot, a single iteration denotes a single forward-backward iteration of the inner subdomains Ω_{in1} and Ω_{in2} . After achieving convergence at the 10th iteration (where $E_2=10^{-2}$) inside the cylinder, we implement the algorithm for the outer domains as a single forward iteration, and we calculate the error value E_1 as 1.5812%. Although Case 1 is much more successful than Case 2 as expected, the error value for Case 2 can be considered at an acceptable level. Moreover, we plot the bistatic RCS profiles of Case 1 and Case 2 in Figure 7.18 by comparing with the whole domain FEM and the MoM solutions. In addition, we plot the contours of the scattered field and the total field in Figure 7.19(a) and Figure 7.19(b), respectively, for the whole domain FEM solution. Similarly, for the FB-DDM solution in Case 1, we plot the contours of the scattered field and the total field in Figure 7.19(c) and Figure 7.19(d), respectively. For the FB-DDM solution in Case 2, we plot the contours of the scattered field and the total field in Figure 7.19(e) and Figure 7.19(f), respectively.

The numerical results in this section demonstrate that the results calculated by the FB-DDM algorithm are sufficiently close to the reference results, and are very reliable even in handling challenging geometries. In addition, we conclude that the numerical results are in conformity with the discussions in Sections 5.1.3 and 5.1.4.

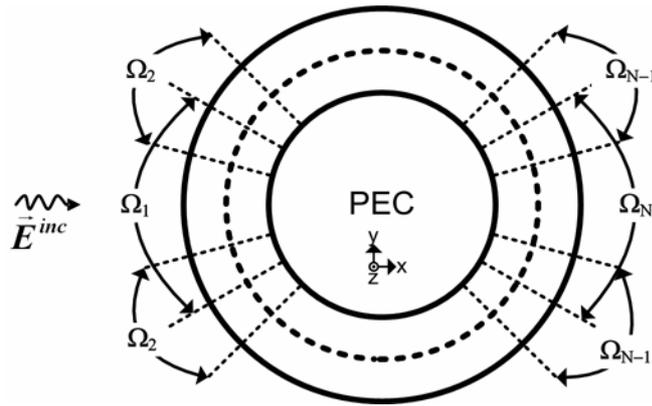


Figure 7.1. Partitioned problem of scattering from circular cylinder (radius 15λ).

Table 7.1. Error values for circular cylinder in the first forward iteration.

E_1 (%)								
$N=2$	$N=3$	$N=4$	$N=5$	$N=6$	$N=7$	$N=8$	$N=9$	$N=10$
0.0045	0.0393	0.0178	0.1088	0.1043	0.1573	0.1776	0.1884	0.2504

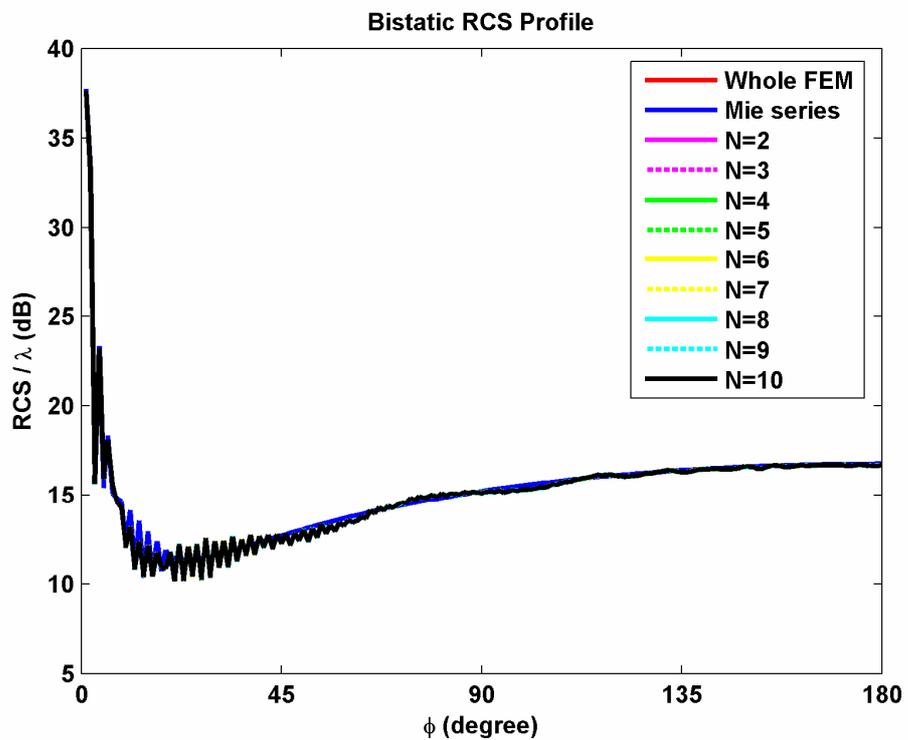


Figure 7.2. Bistatic RCS profile of circular PEC cylinder.

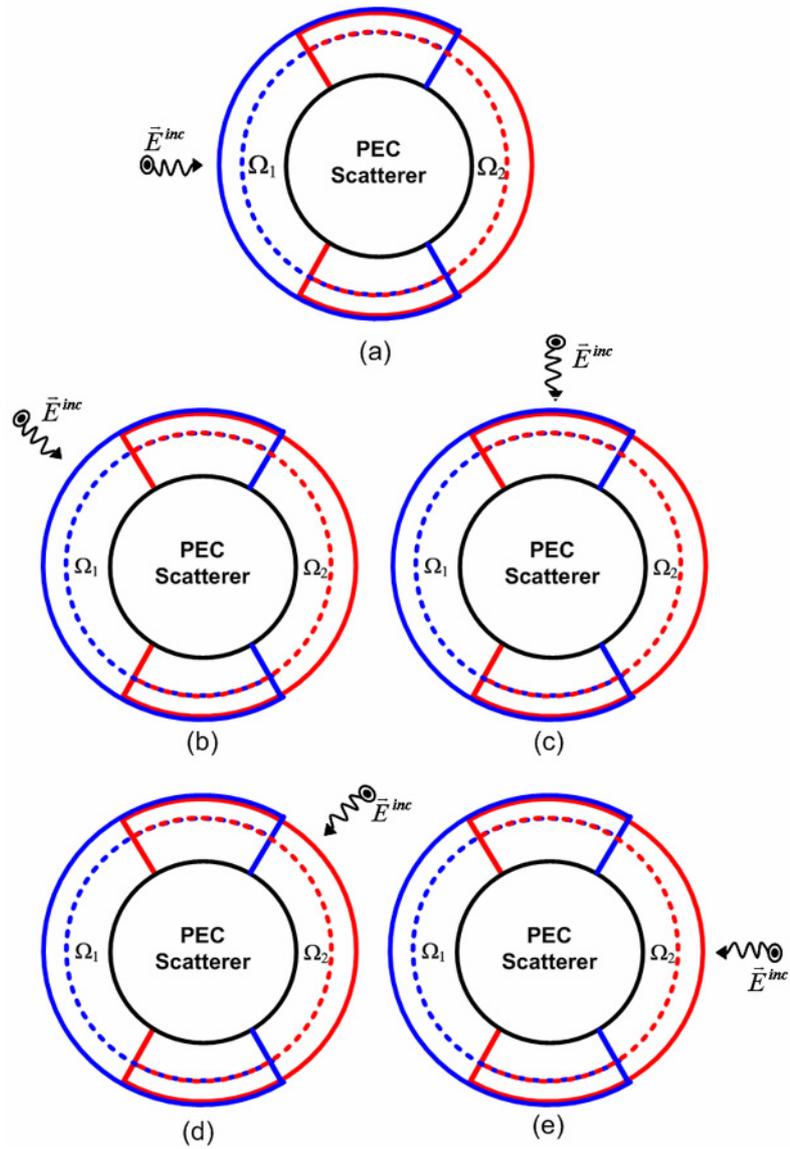


Figure 7.3. Scattering by circular cylinder with different illuminations: (a) $\phi^{\text{inc}}=180^\circ$, (b) $\phi^{\text{inc}}=135^\circ$, (c) $\phi^{\text{inc}}=90^\circ$, (d) $\phi^{\text{inc}}=45^\circ$, (e) $\phi^{\text{inc}}=0^\circ$.

Table 7.2. Error values for circular cylinder with different illuminations in the first forward iteration.

E_1 (%)				
Case (a)	Case (b)	Case (c)	Case (d)	Case (e)
0.0045	4.5686	4.0355	3.0451	2.3146

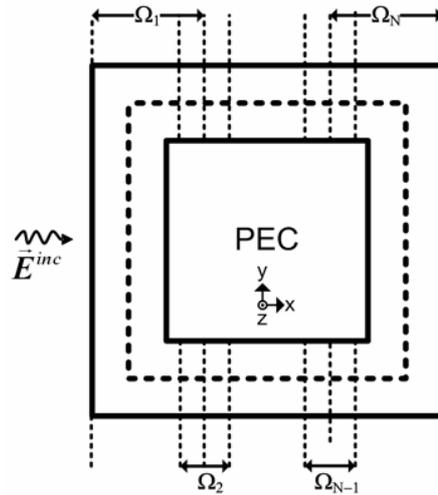


Figure 7.4. Partitioned problem of scattering from square cylinder (edge-length 20λ).

Table 7.3. Error values for square cylinder in the first forward iteration.

E_1 (%)			
$N=2$	$N=3$	$N=4$	$N=5$
0.0111	0.0193	0.0255	0.0321

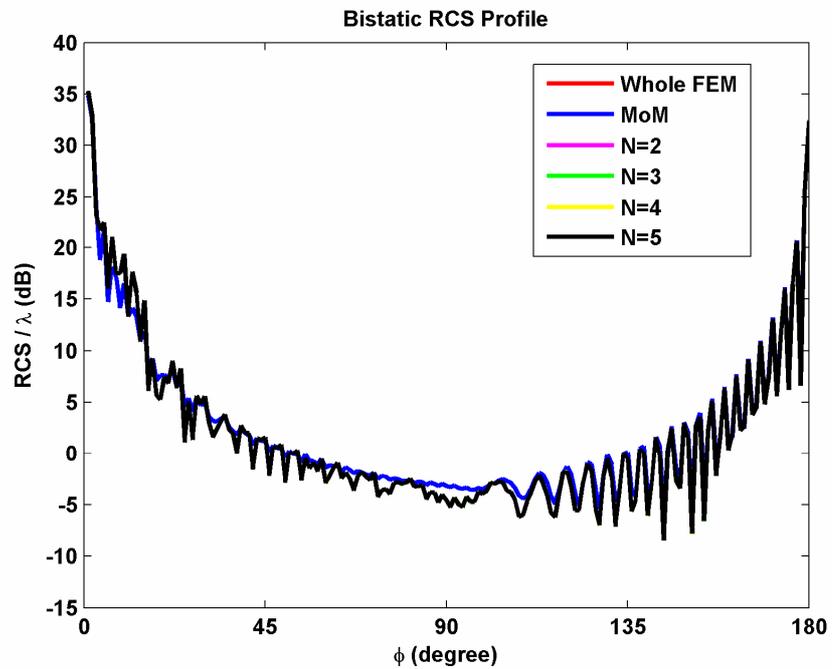


Figure 7.5. Bistatic RCS profile of square PEC cylinder.

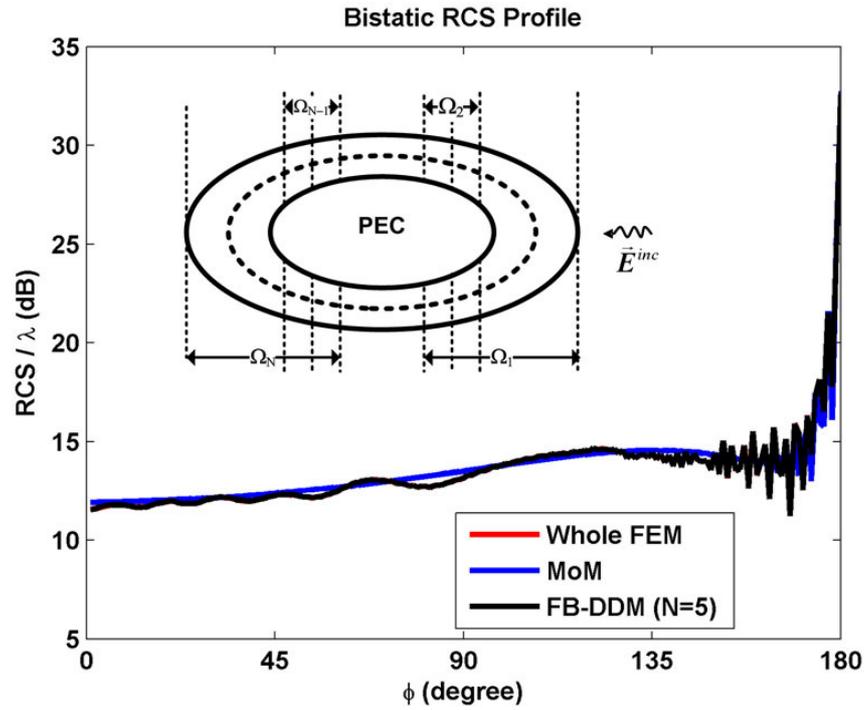


Figure 7.6. Bistatic RCS profile of elliptical cylinder in the 1st forward iteration (major axis is 40λ , axial ratio is 2).

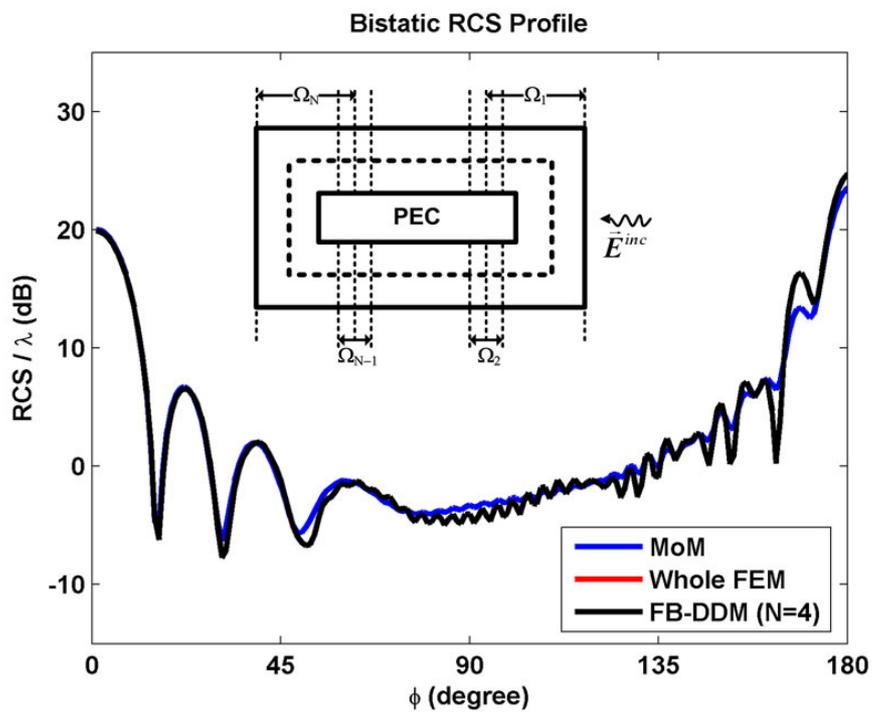


Figure 7.7. Bistatic RCS profile of rectangular cylinder in the 1st forward iteration ($16\lambda \times 4\lambda$).

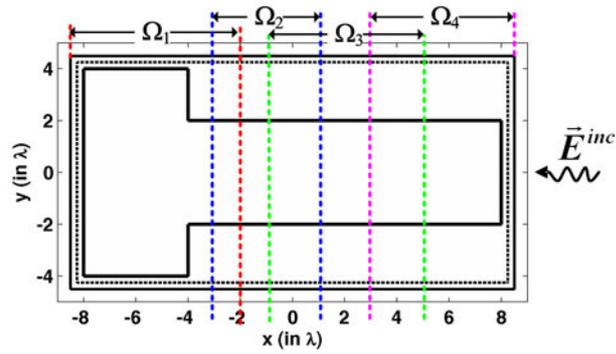


Figure 7.8. Partitioned problem of scattering from T-shaped PEC cylinder.

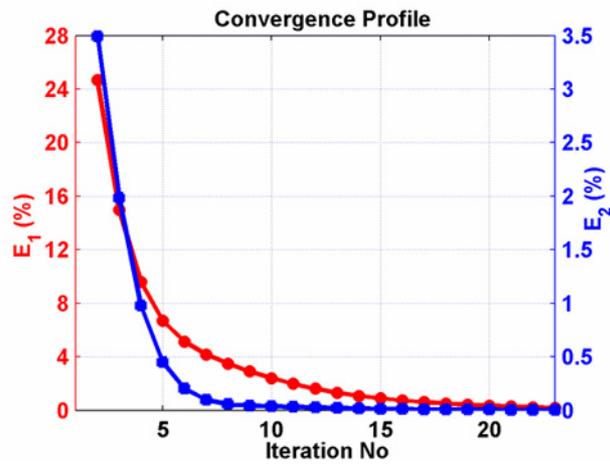


Figure 7.9. Convergence profile of T-shaped PEC cylinder.

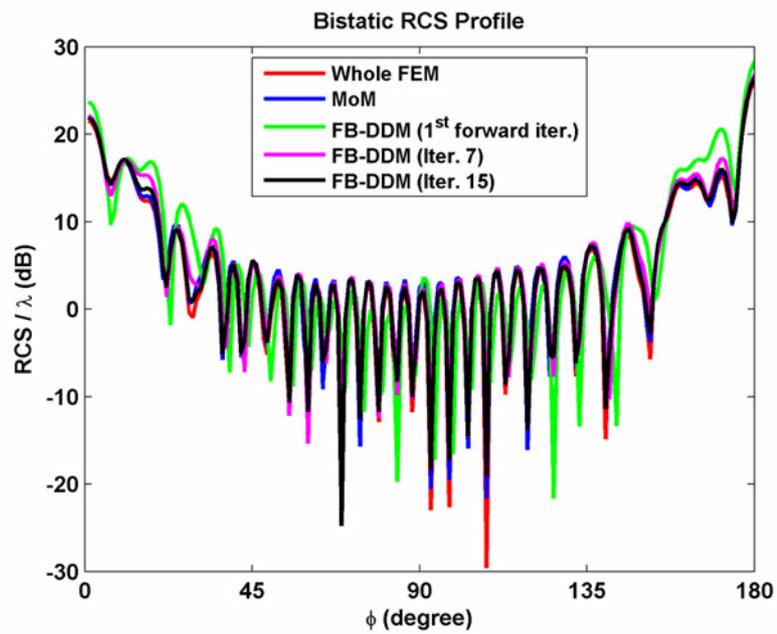


Figure 7.10. Bistatic RCS profile of T-shaped PEC cylinder.

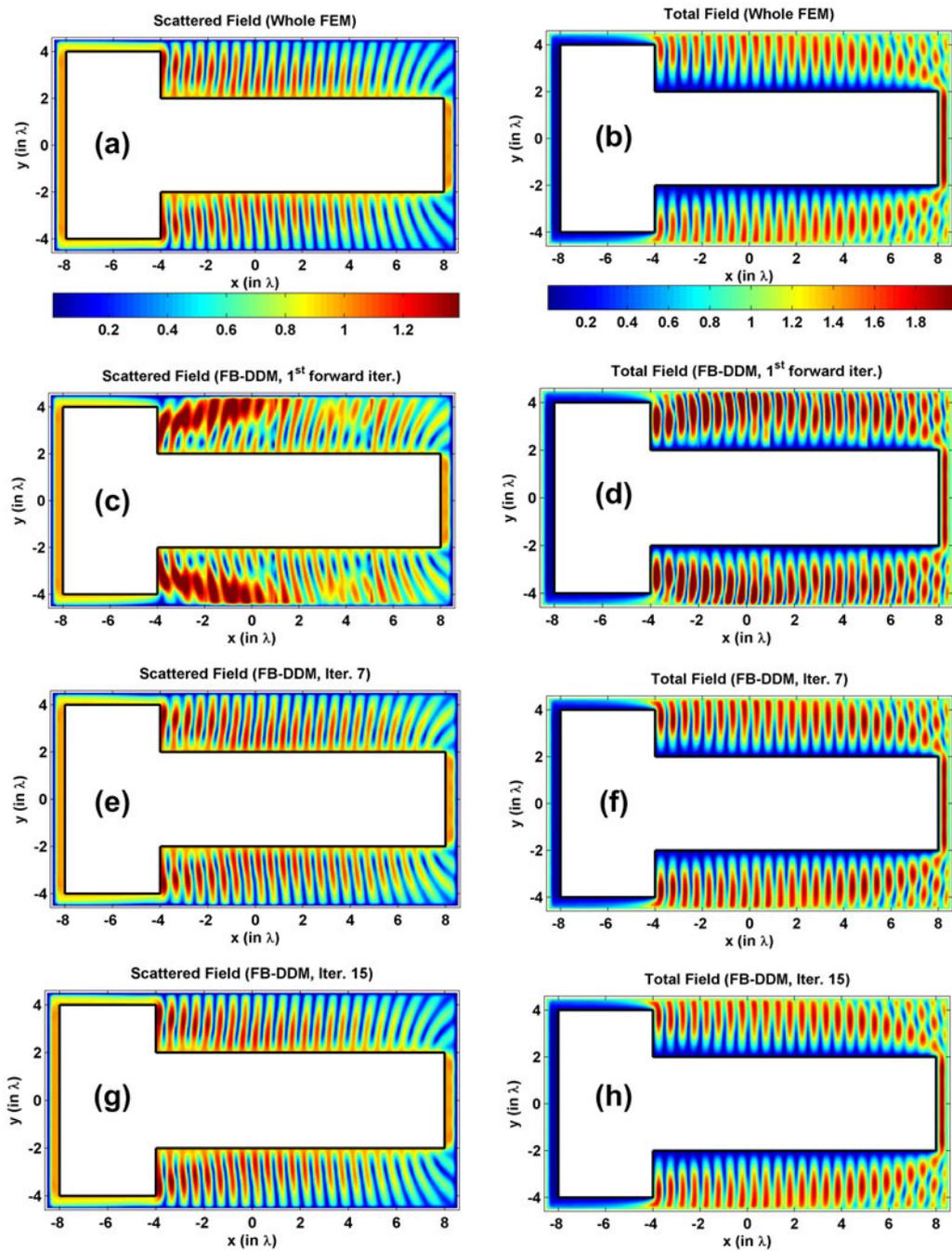


Figure 7.11. Field contours of T-shaped PEC cylinder (magnitude): (a-b) Scattered and total fields of the whole FEM, (c-d) Scattered and total fields in the 1st forward iteration, (e-f) Scattered and total fields in the 7th iteration, (g-h) Scattered and total fields in the 15th iteration.

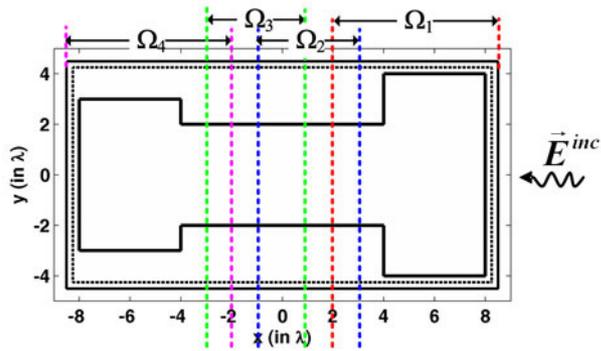


Figure 7.12. Partitioned problem of scattering from H-shaped PEC cylinder.

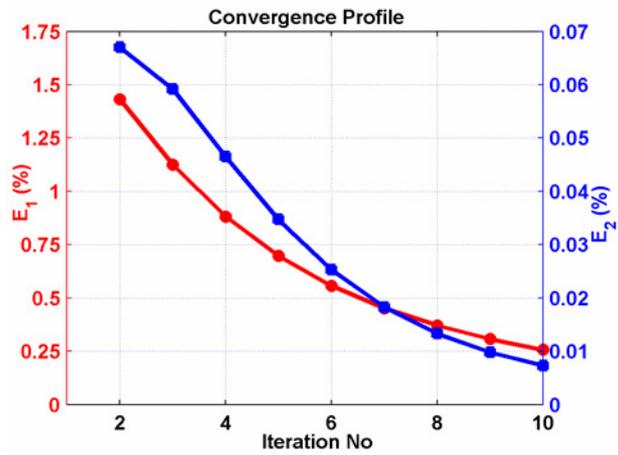


Figure 7.13. Convergence profile of H-shaped PEC cylinder.

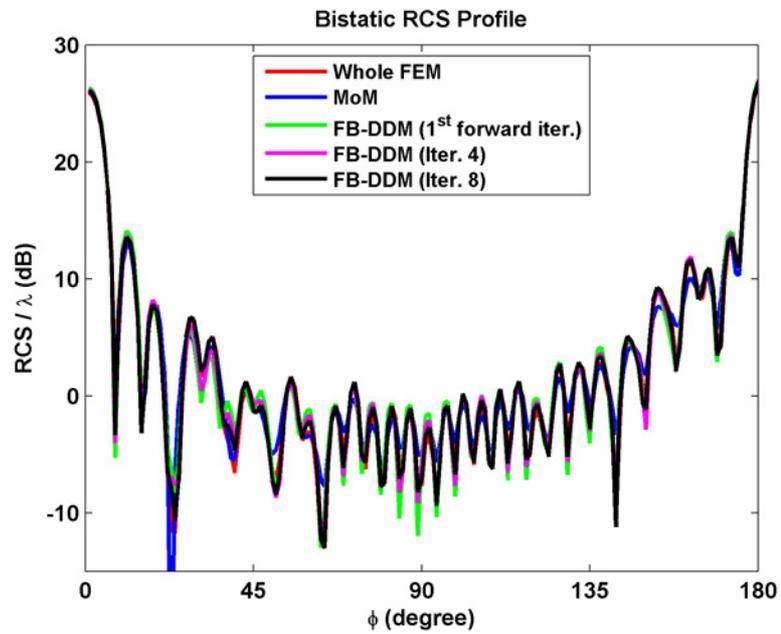


Figure 7.14. Bistatic RCS profile of H-shaped PEC cylinder.

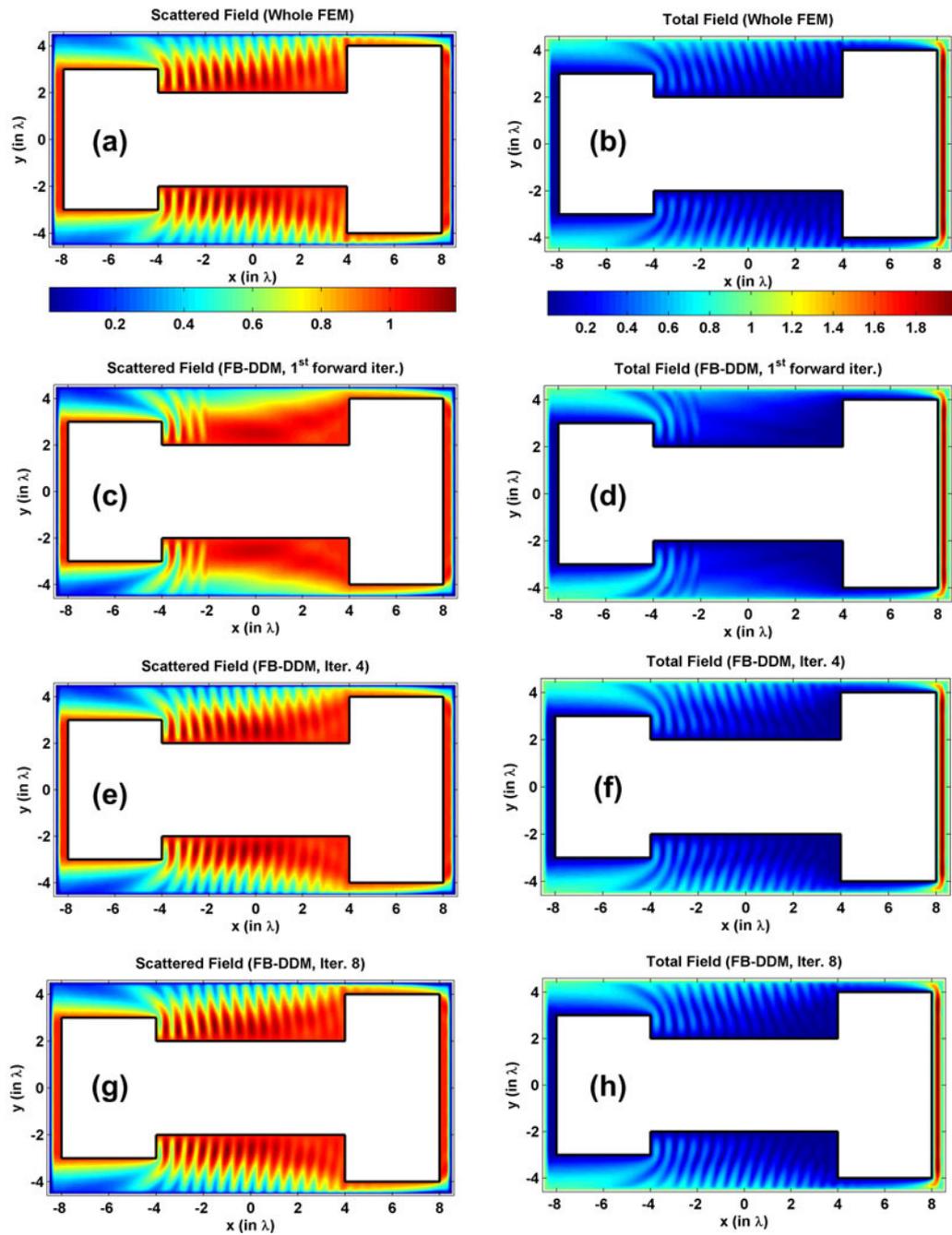


Figure 7.15. Field contours of H-shaped PEC cylinder (magnitude): (a-b) Scattered and total fields of the whole FEM, (c-d) Scattered and total fields in the 1st forward iteration, (e-f) Scattered and total fields in the 4th iteration, (g-h) Scattered and total fields in the 8th iteration.

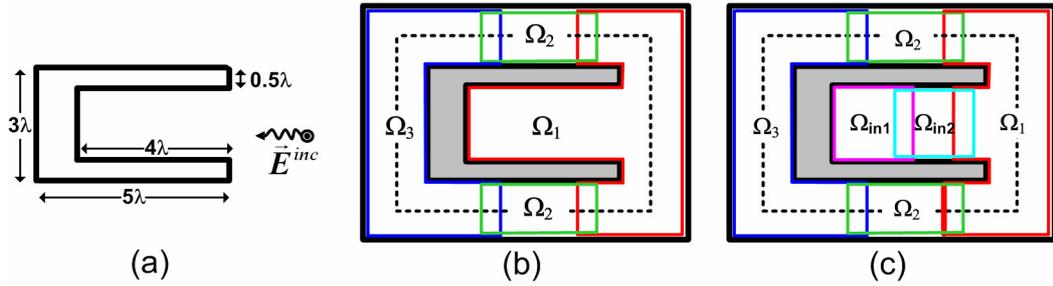


Figure 7.16. Scattering from U-shaped cylinder: (a) Original problem, (b) Partitioned problem (Case 1), (c) Partitioned problem (Case 2).

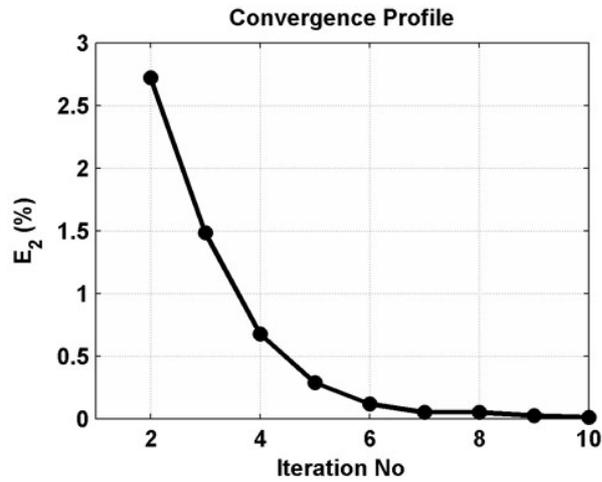


Figure 7.17. Convergence profile in the inner part of U-shaped cylinder (Case2).

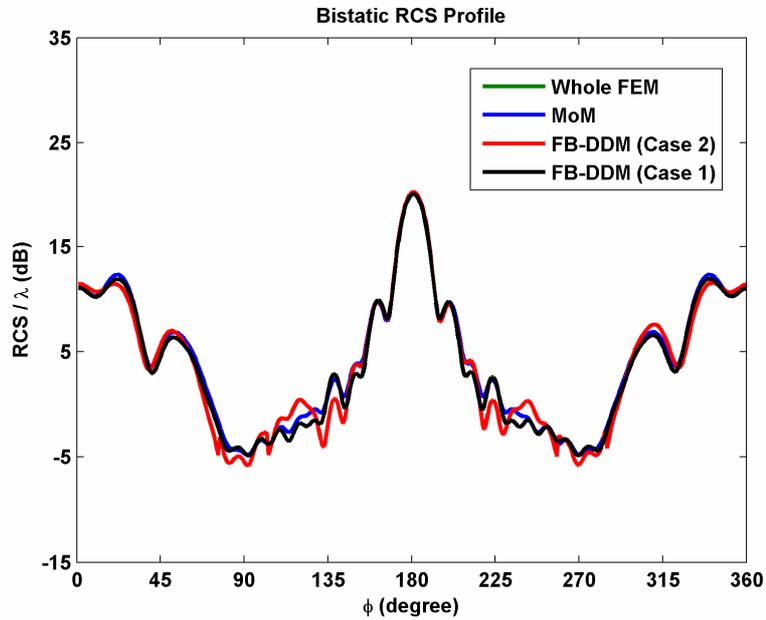


Figure 7.18. Bistatic RCS profile of U-shaped PEC cylinder.

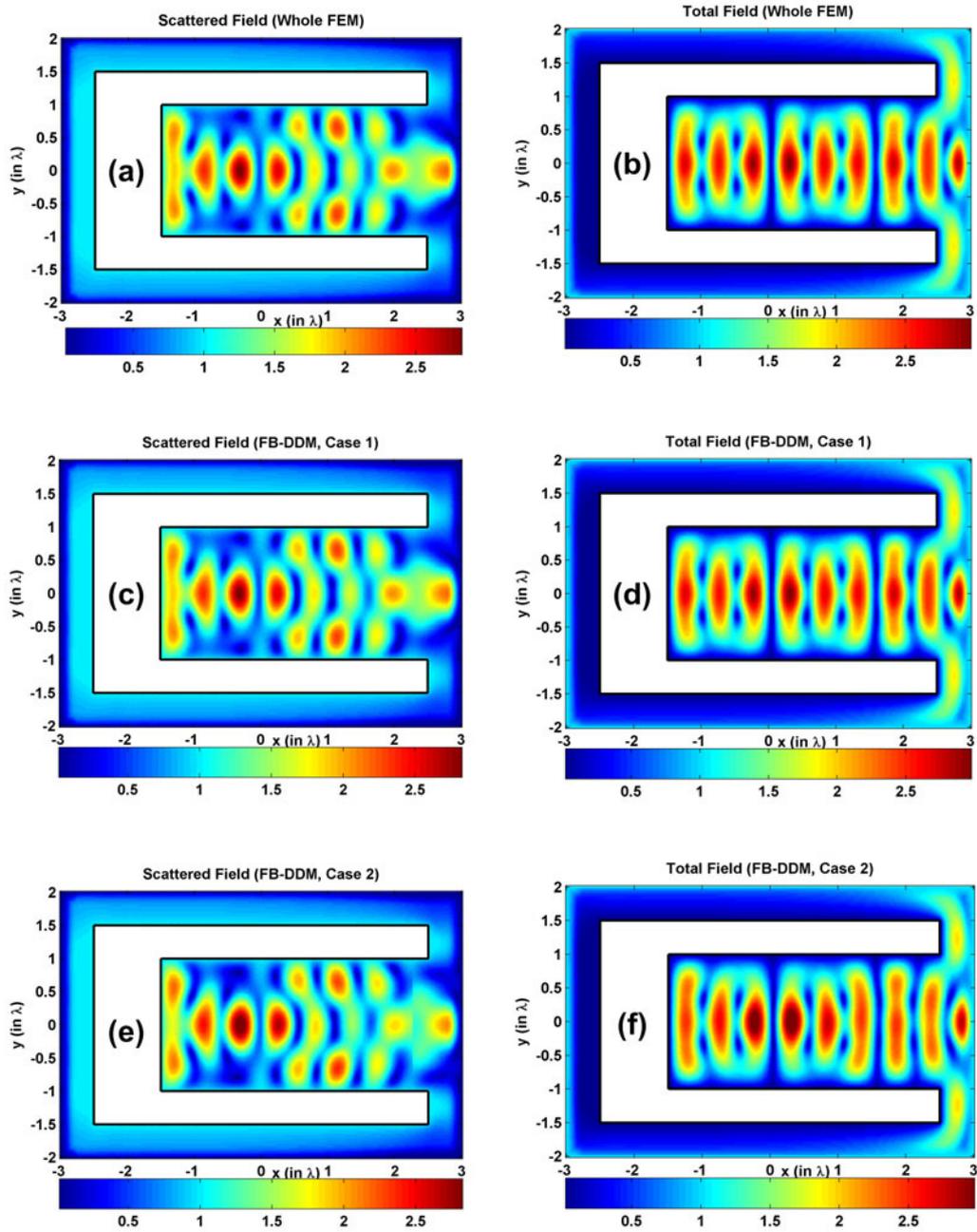


Figure 7.19. Field contours of U-shaped PEC cylinder (magnitude): (a-b) Scattered and total fields of the whole FEM, (c-d) Scattered and total fields in Case 1, (e-f) Scattered and total fields in Case 2.

7.2. 3D Simulations

In this section, we test the FB-DDM algorithm in 3D electromagnetic scattering problems. In all examples, the locally-conformal PML method is implemented using the parameters $\alpha=5k$ and $m=3$, and the approximate PML thickness is chosen as $\lambda/5$ (for both existing PML region and additional PML region). The wavenumber k is set to 20π (i.e., the wavelength λ is 0.1 meter). The element size is approximately set to $\lambda/10$.

The 1st example is a benchmark problem which is a scattering from a PEC sphere whose diameter is 4λ . We assume that the incident plane wave propagates along the positive z -direction (axial incidence), and the electric field is polarized along the x -direction (i.e., $\vec{E}^{inc} = \hat{a}_x \exp(-jkz)$). The number of unknowns (or the number of edges) in this problem is 237,898. The decomposition of the whole problem into $N=9$ number of subdomains is illustrated in Figure 7.20(b) as an inset. We implement the FB-DDM algorithm in just the first forward iteration, and we calculate the error value E_1 as 1.2566%. We also plot the bistatic RCS profiles in Figure 7.20 by comparing them with the whole domain FEM solution and the analytical Mie series solution [84]. Then, we consider larger spheres whose diameters are 6λ , 8λ and 10λ , for the same incident field. The number of unknowns is: 483,920 for 6λ , 684,020 for 8λ , and 1,043,757 for 10λ . The whole problem is decomposed into N subdomains, where $N=14$ for 6λ , $N=19$ for 8λ , and $N=24$ for 10λ . We process the FB-DDM algorithm in just the first forward iteration, and we plot the bistatic RCS profiles in Figure 7.21, Figure 7.22 and Figure 7.23.

The 2nd example considers the problem of scattering from a PEC cube whose edge length is 3λ and 5λ , assuming that the incident plane wave is $\vec{E}^{inc} = \hat{a}_y \exp(-jkz)$. The whole problem is decomposed into N subdomains,

where $N=9$ for 3λ , and $N=24$ for 5λ (see the insets in Figure 7.24(b) or Figure 7.25(b)). The number of unknowns is: 248,556 for 3λ , and 624,675 for 5λ . We implement the FB-DDM algorithm in just the first forward iteration, and we plot the bistatic RCS profiles in Figure 7.24 and Figure 7.25 by comparing them with the results in literature [97].

The 3rd example considers the problem of scattering from a PEC conesphere whose half-cone angle is 45° , base radius is 1.6λ , and cone-height is 1.6λ , assuming an axial (nose-on) plane wave incidence. The whole problem is decomposed into $N=7$ subdomains (see the inset in Figure 7.26(b)), and the number of unknowns is 177,730. We implement the FB-DDM algorithm in just the first forward iteration, and we calculate the error value E_1 as 1.2656%. We plot the bistatic RCS profiles in Figure 7.26 by comparing them with the whole domain FEM solution and the results in literature [90].

The 4th example considers the problem of scattering from a PEC missile whose radome diameter is 1λ and length is 6λ (excluding the radome part). We assume that the incident plane is $\vec{E}^{inc} = \hat{a}_y \exp(jkz)$. The whole problem is decomposed into $N=5$ subdomains (see the inset in Figure 7.27(b)), and the number of unknowns is 106,462. We implement the FB-DDM algorithm in just the first forward iteration, and we calculate the error value E_1 as 0.6085%. We also plot the bistatic RCS profiles in Figure 7.27 by comparing them with the whole domain FEM solution. We also consider a ‘larger’ missile whose radome diameter is 2λ and length is 20λ , for the same incident field. The original domain (the number of unknowns is 563,313) is decomposed into $N=19$ number of subdomains. The bistatic RCS profile is plotted in Figure 7.28.

The 5th example involves the problem of scattering from a PEC inlet whose base is a $1\lambda \times 1\lambda$ square and its depth is 3λ . We assume that the incident plane wave is

$\vec{E}^{inc} = \hat{a}_x \exp(jkz)$. The whole problem (the number of unknowns is 125,484) is decomposed as shown in Figure 5.9 in Section 5.1.4.1. In Case 1, we implement the FB-DDM algorithm as in Figure 5.9(b) where the first subdomain includes the inner part of the inlet. For the first forward iteration, we calculate the error value E_1 as 0.9479%. In Case 2, we process the algorithm as in Figure 5.9(c), where the number of inner subdomains is only 1. Therefore, we solve the first inner subdomain only once, and then solve the outer subdomains in the first forward iteration. Then, we calculate the error value E_1 as 2.781%. In Case 3, we run the algorithm as in Figure 5.9(c), where the number of inner subdomains is 2. Inside the inlet, we solve the inner subdomains iteratively up to the 6th iteration (where $E_2=10^{-1}$). Then we solve the outer subdomains in the first forward iteration, and we calculate the error value E_1 as 2.9935%. We plot the bistatic RCS profiles of all cases in Figure 7.29 with the whole FEM solution.

The 6th example considers the problem of scattering from a PEC square plate which is described in Section 5.1.4.2 as a special problem (see Figure 5.11(a)). We implement the FB-DDM algorithm using the ‘puzzle-like’ partition scheme shown in Figure 5.12, and the x (or y) partition scheme shown in Figure 5.11. We analyze the following 4 different cases about the plates:

- First, we analyze the square plate whose edge length is 4λ . We assume that the incident plane wave is $\vec{E}^{inc} = \hat{a}_y \exp(jkz)$. We decompose the original domain (the number of unknowns is 176,370) into 9 subdomains using the x-partition, and we implement the algorithm in the first forward iteration. We also decompose the original domain into 4×4 subdomains using the puzzle-like partition, and we implement the algorithm in 2 forward-backward iterations. We plot the bistatic RCS profiles in Figure 7.30 by comparing them with the whole domain FEM solution, the MoM solution [81] and the PO solution [82].

- Second, we analyze the square plate whose edge length is 10λ , assuming the same incident field. We decompose the original domain (number of unknowns is 859,128) into 24 subdomains using the x-partition, and we implement the algorithm in the first forward iteration. We also decompose the original domain into 10×10 subdomains using the puzzle-like partition, and we implement the algorithm in the single forward-backward iteration. We plot the bistatic RCS profiles in Figure 7.31 by comparing them with the MoM and PO solutions.
- Third, we analyze the square plate whose edge length is 2λ , assuming that the incident field is $\vec{E}^{inc} = \hat{a}_x \exp[-jk(y \sin \theta^{inc} - z \cos \theta^{inc})]$ where $\theta^{inc} = 30^\circ$ (ϕ -polarization, oblique incidence). We decompose the domain into 4 subdomains using the y-partition, and we implement the algorithm in the first forward iteration. We also decompose the domain into 2×2 subdomains using the puzzle-like partition, and we implement the algorithm in the single forward-backward iteration. We plot the bistatic RCS profiles in Figure 7.32 by comparing them with the whole domain FEM, MoM and PO solutions.
- Finally, we analyze the same square plate whose edge length is 2λ , assuming that the incident field is $\vec{E}^{inc} = [-\hat{a}_x \sin \phi^{inc} + \hat{a}_y \cos \phi^{inc}] \exp[jkz]$ where $\phi^{inc} = 45^\circ$ (ϕ -polarization, normal incidence). We decompose the original domain into 2×2 subdomains using the puzzle-like partition, and we implement the algorithm in 2 forward-backward iterations. We also decompose the domain into 4 subdomains using the x-partition, and we implement the algorithm in the first forward iteration. We plot the bistatic RCS profiles in Figure 7.33 by comparing them with the whole domain FEM, MoM and PO solutions.

Based on the numerical experiments in this section, we conclude that the results are in conformity with the discussions in Sections 5.1.3 and 5.1.4.

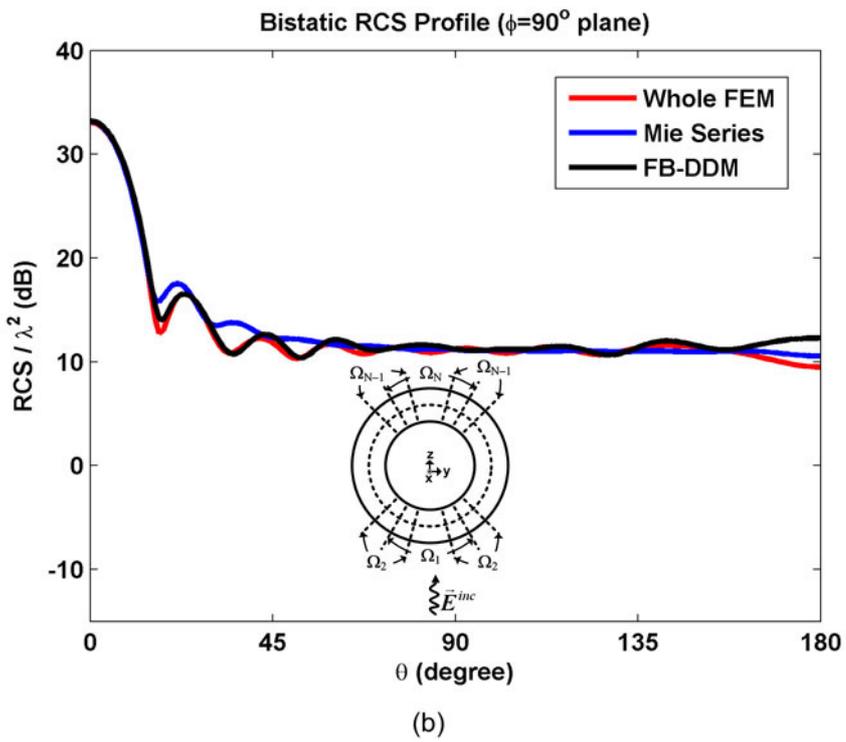
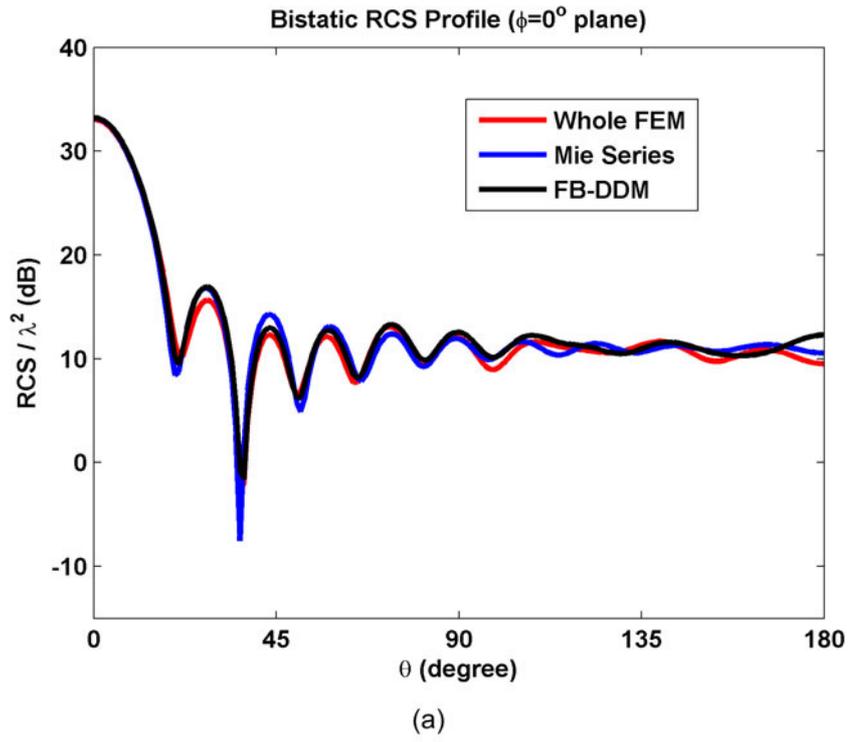
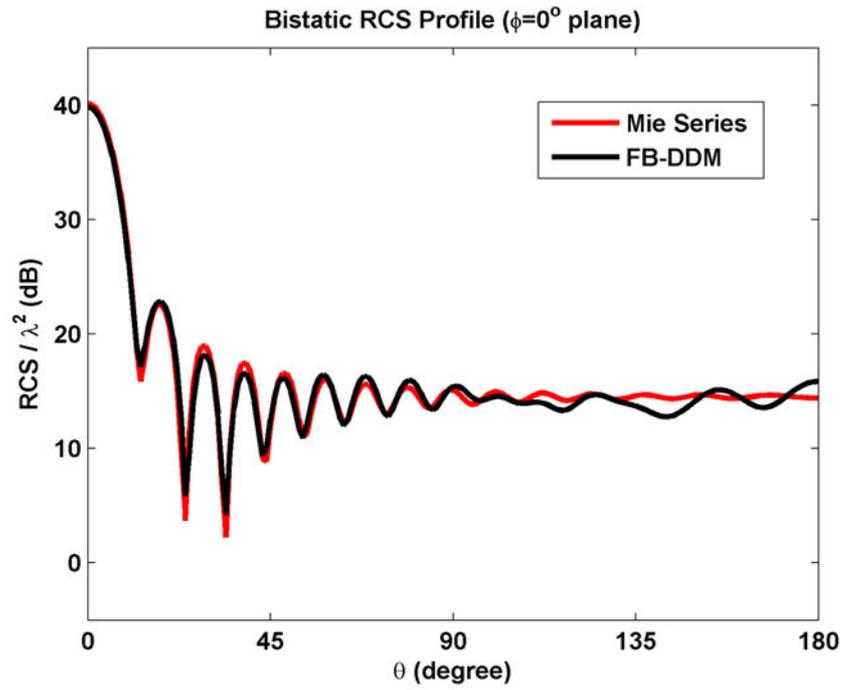
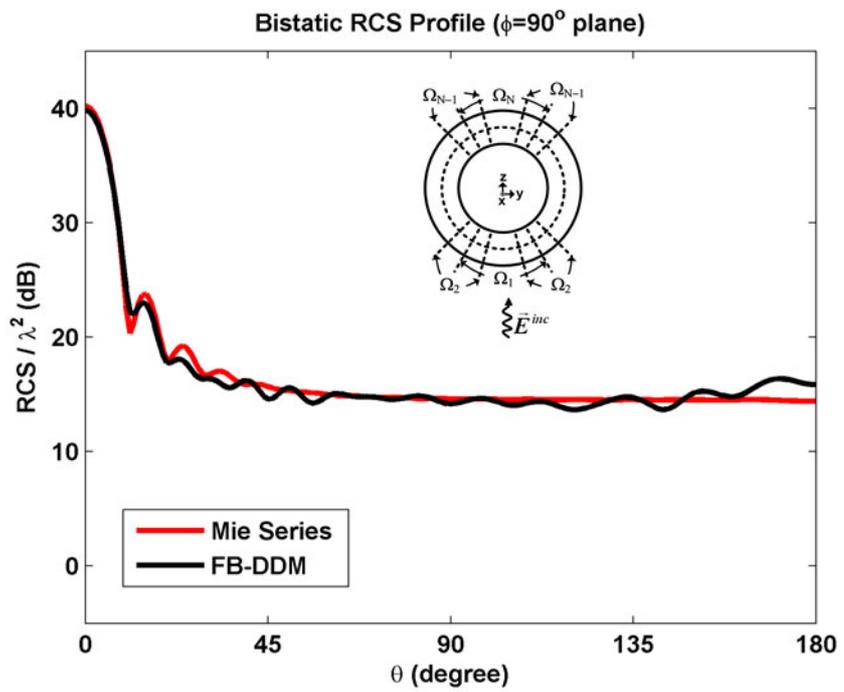


Figure 7.20. Bistatic RCS profile of sphere whose diameter is 4λ : (a) $\phi=0^\circ$ plane, (b) $\phi=90^\circ$ plane.



(a)



(b)

Figure 7.21. Bistatic RCS profile of sphere whose diameter is 6λ : (a) $\phi=0^\circ$ plane, (b) $\phi=90^\circ$ plane.

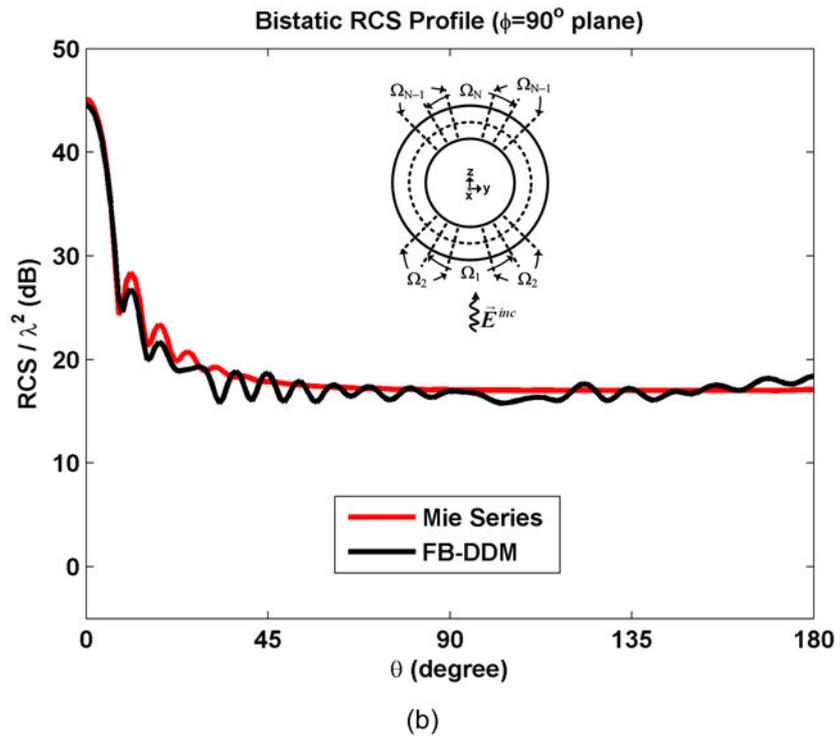
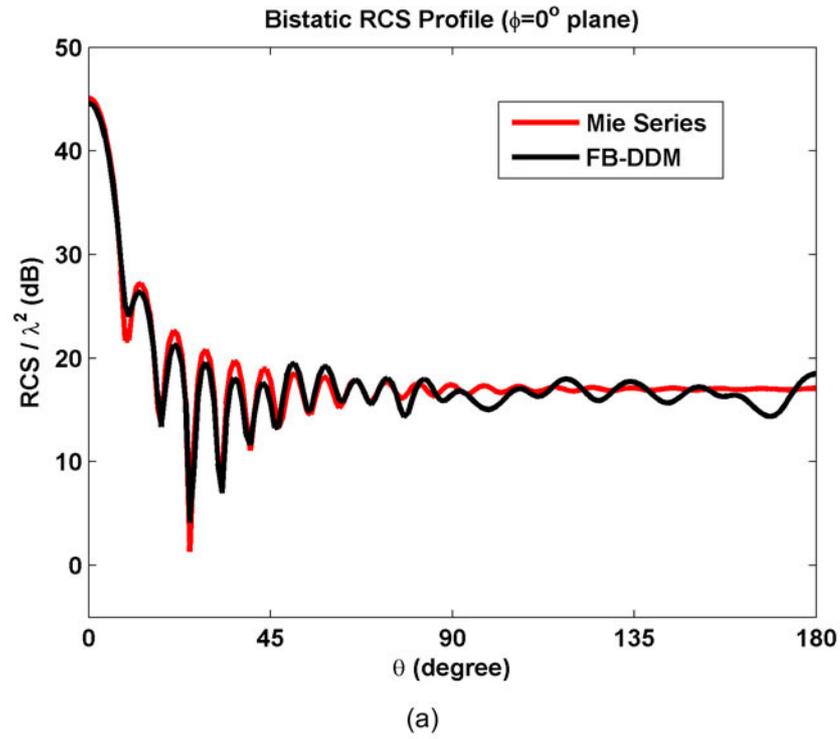
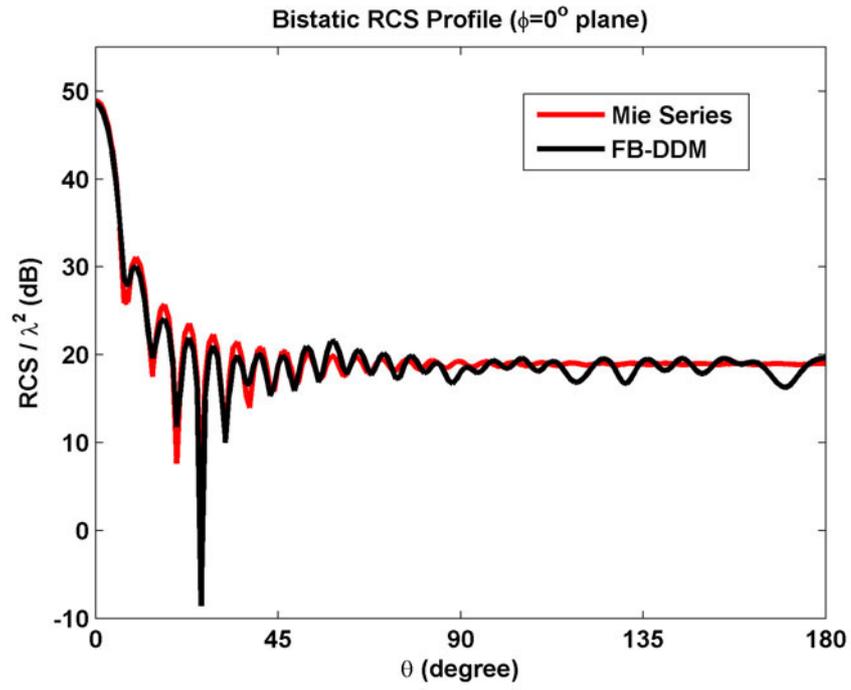
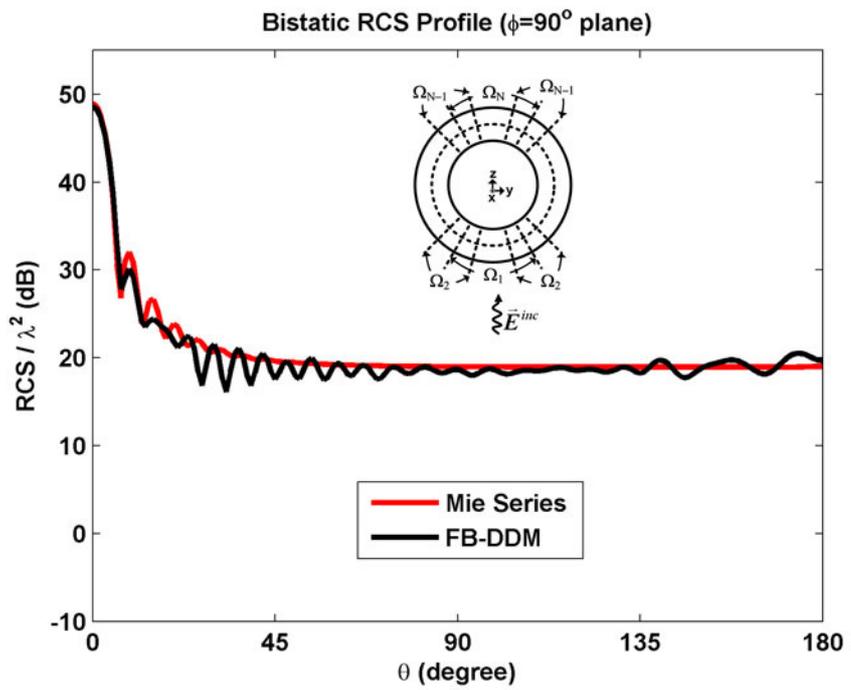


Figure 7.22. Bistatic RCS profile of sphere whose diameter is 8λ : (a) $\phi=0^\circ$ plane, (b) $\phi=90^\circ$ plane.



(a)



(b)

Figure 7.23. Bistatic RCS profile of sphere whose diameter is 10λ : (a) $\phi=0^\circ$ plane, (b) $\phi=90^\circ$ plane.

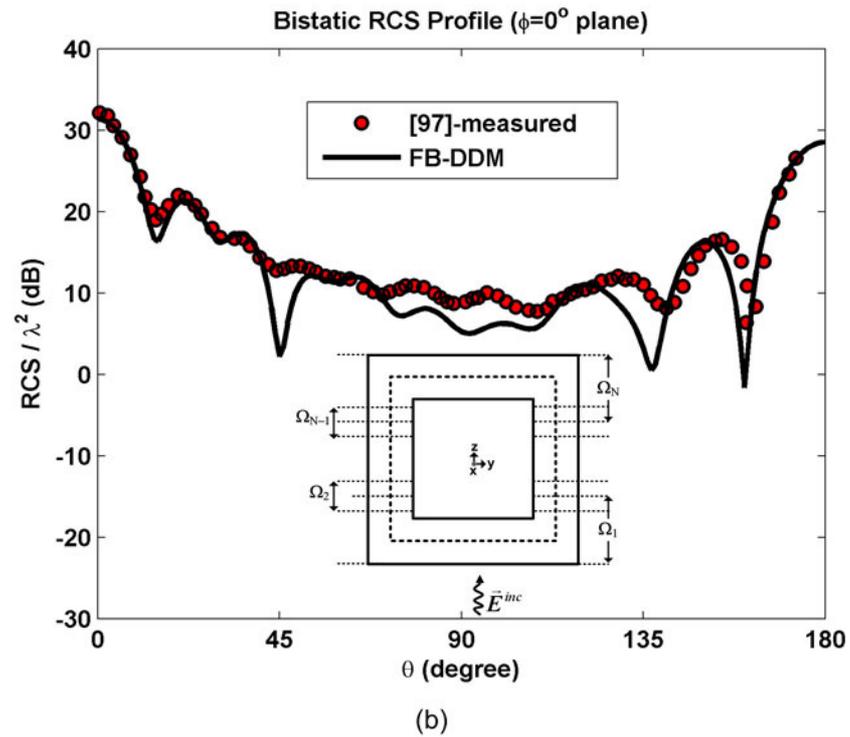
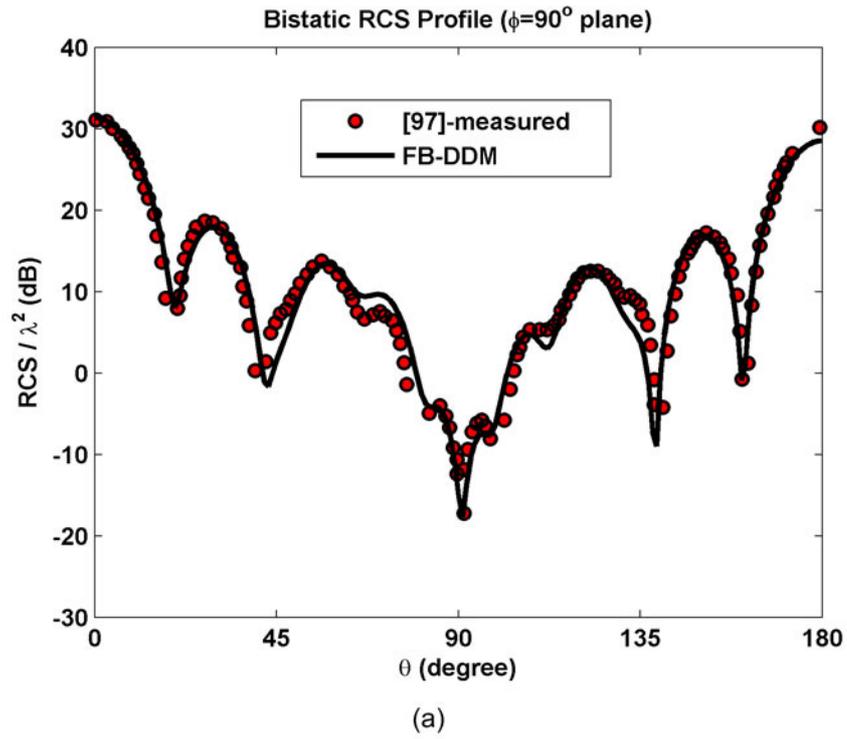
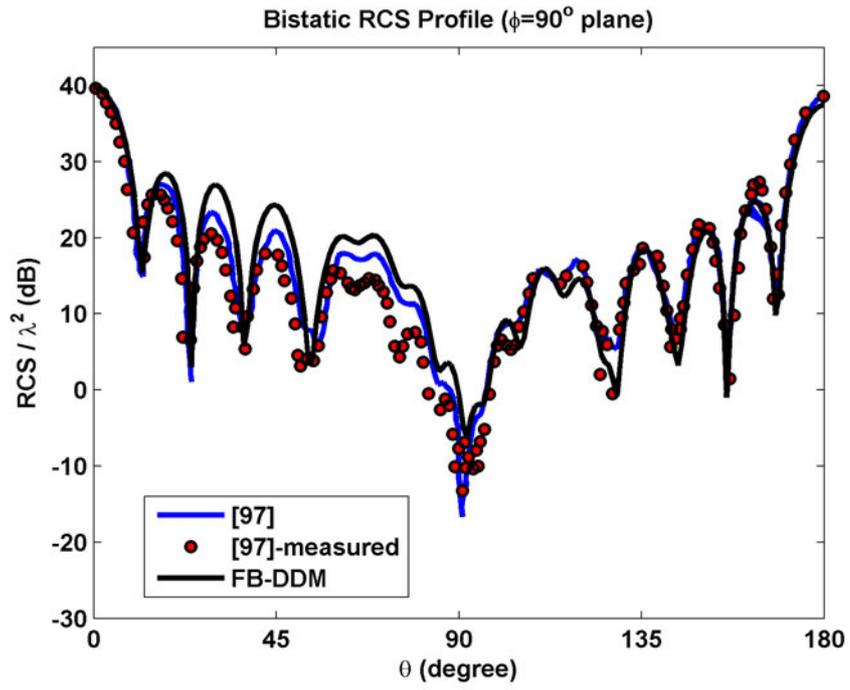
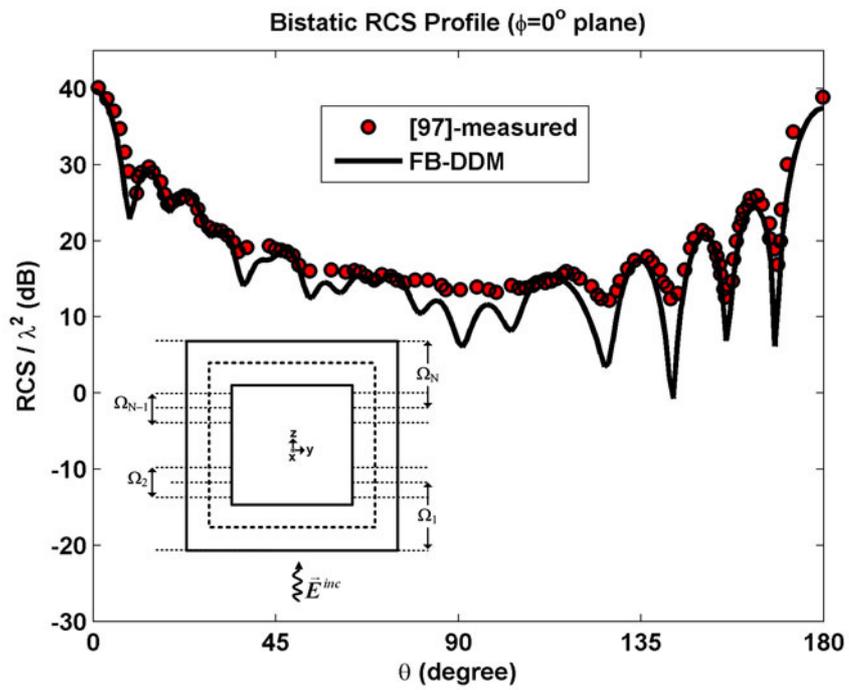


Figure 7.24. Bistatic RCS profile of cube whose edge length is 3λ : (a) $\phi=90^\circ$ plane, (b) $\phi=0^\circ$ plane.

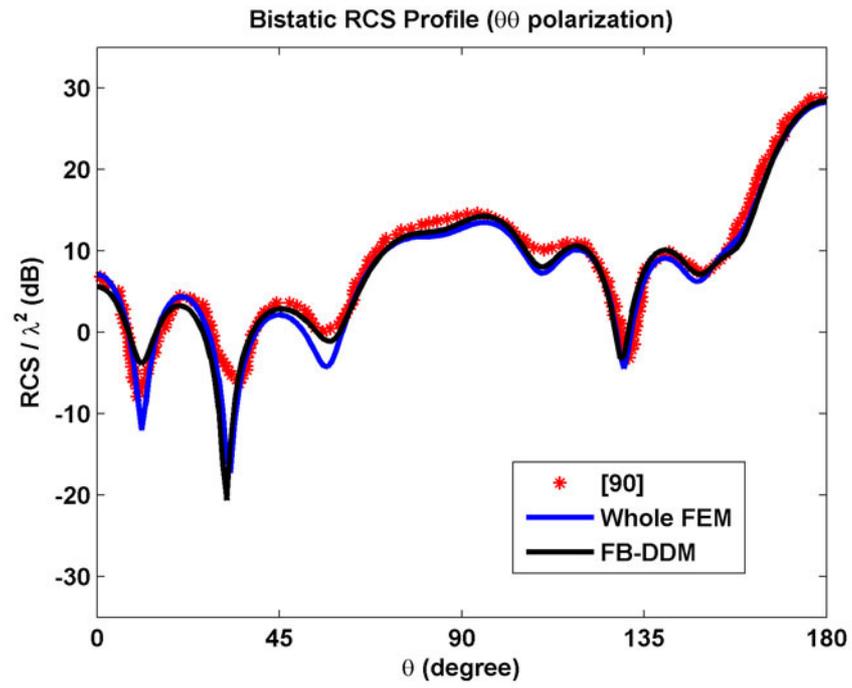


(a)

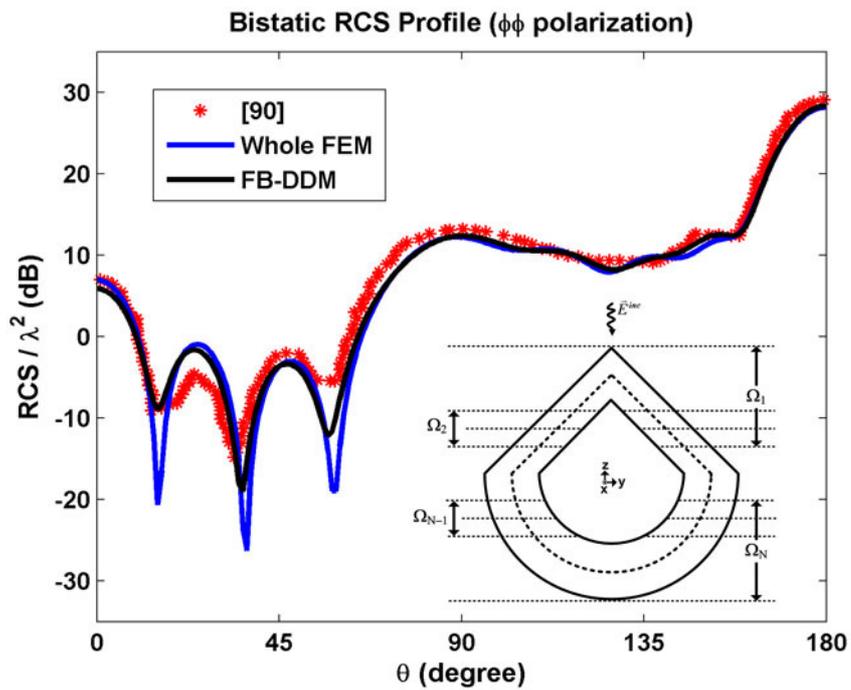


(b)

Figure 7.25. Bistatic RCS profile of cube whose edge length is 5λ : (a) $\phi=90^\circ$ plane, (b) $\phi=0^\circ$ plane.



(a)



(b)

Figure 7.26. Bistatic RCS profile of conisphere (radius 1.6λ , nose angle 90°): (a) $\theta\theta$ polarization, (b) $\phi\phi$ polarization.

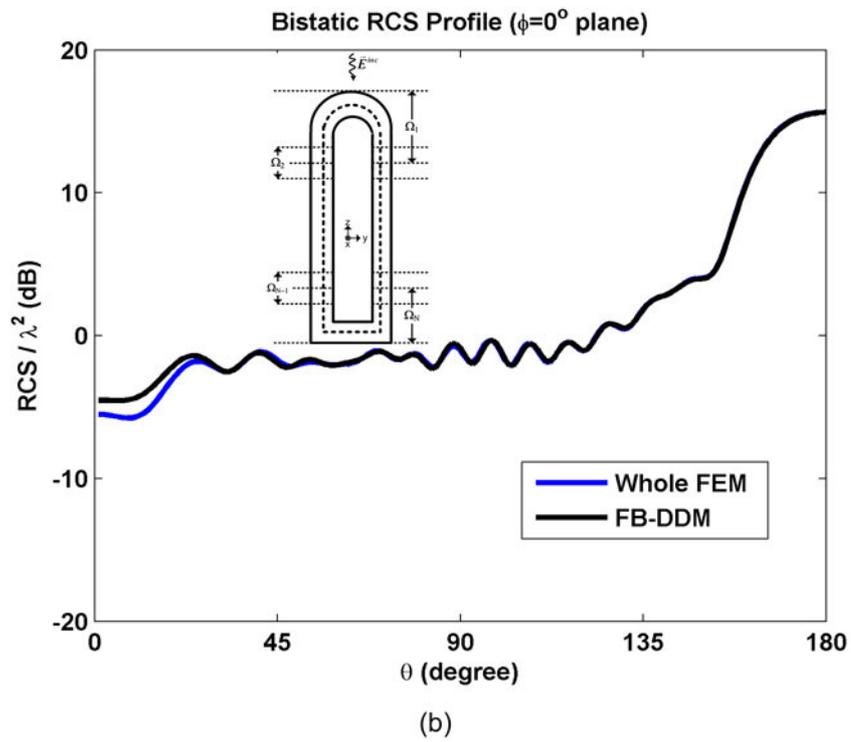
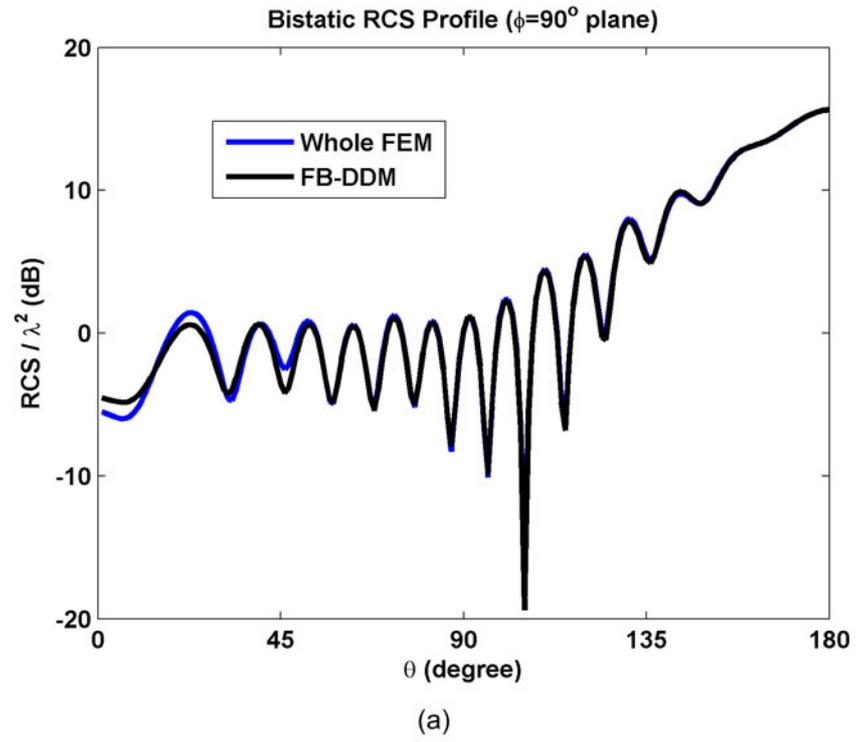


Figure 7.27. Bistatic RCS profile of missile (diameter 1λ , length 6λ): (a) $\phi=90^\circ$ plane, (b) $\phi=0^\circ$ plane.

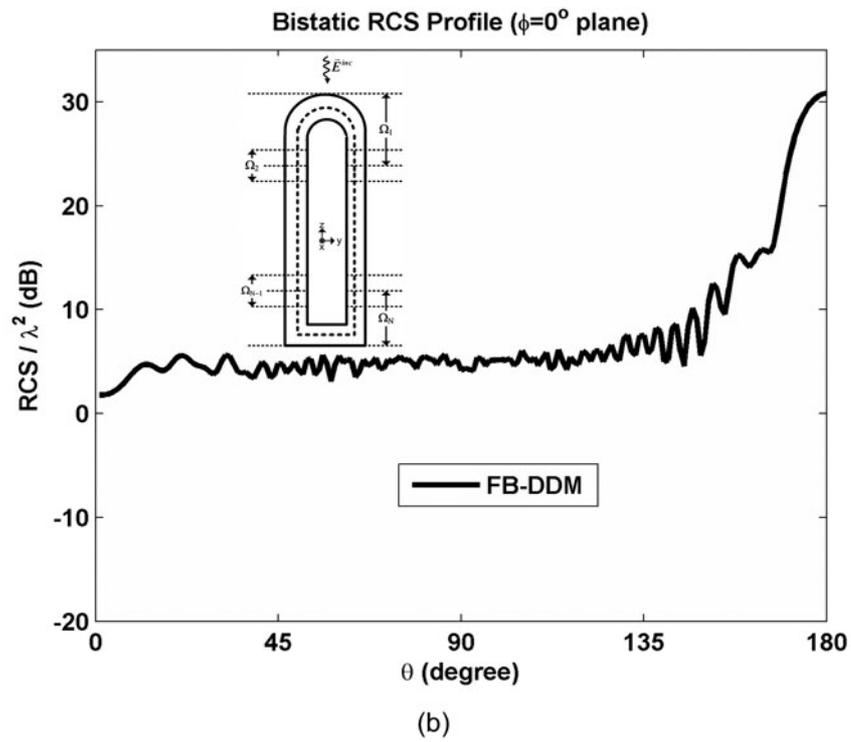
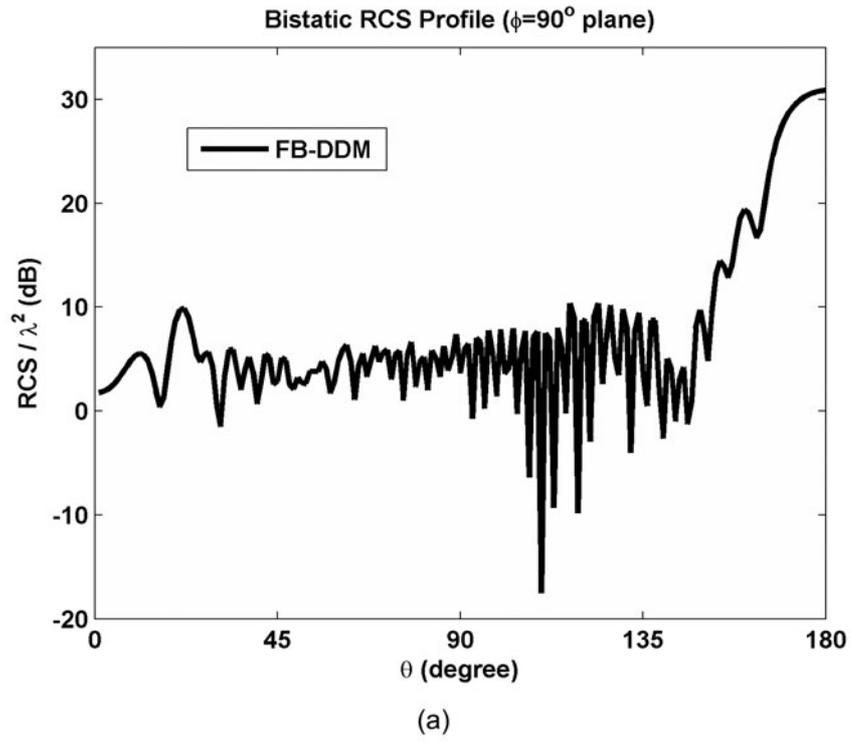


Figure 7.28. Bistatic RCS profile of large missile (diameter 2λ , length 20λ): (a) $\phi=90^\circ$ plane, (b) $\phi=0^\circ$ plane.

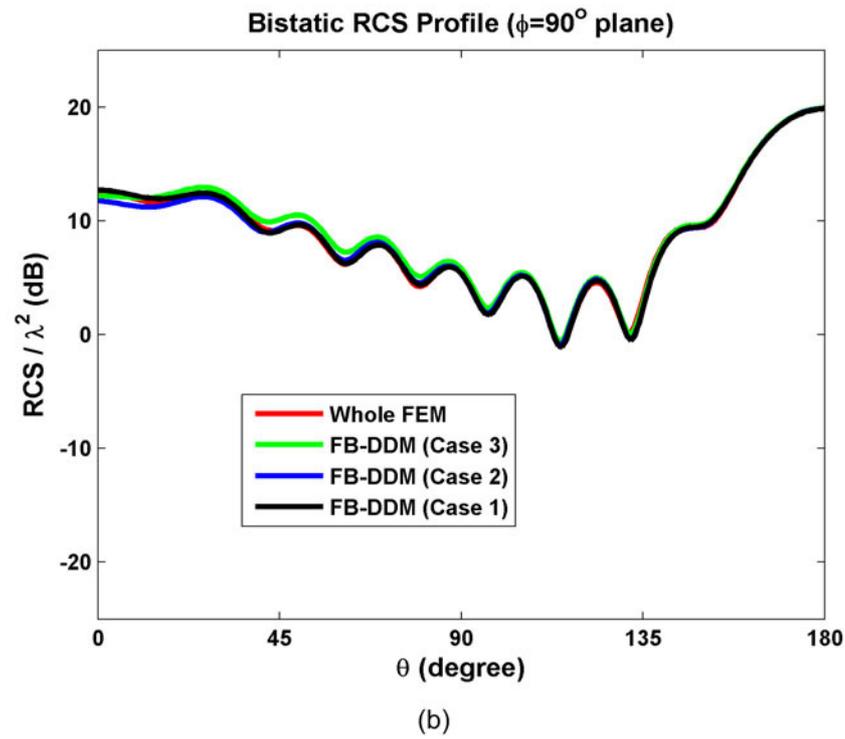
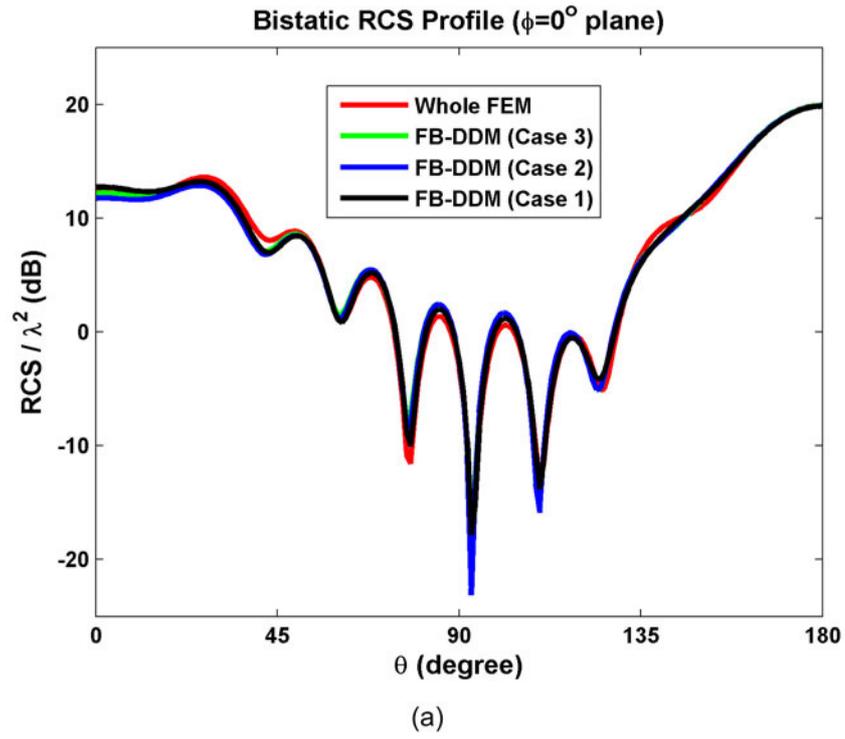


Figure 7.29. Bistatic RCS profile of inlet (base $1\lambda \times 1\lambda$, depth 3λ): (a) $\phi=0^\circ$ plane, (b) $\phi=90^\circ$ plane.

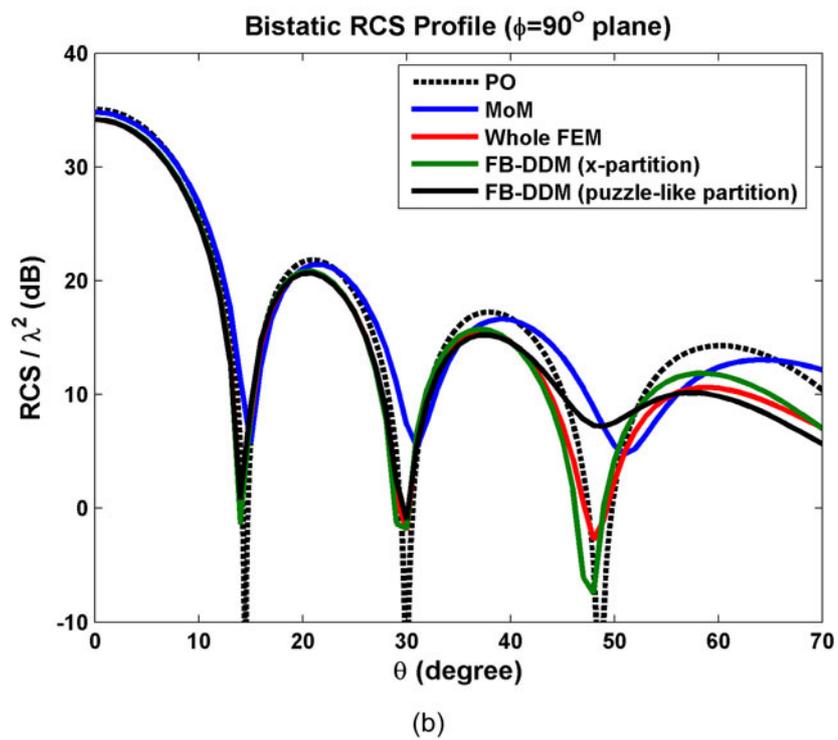
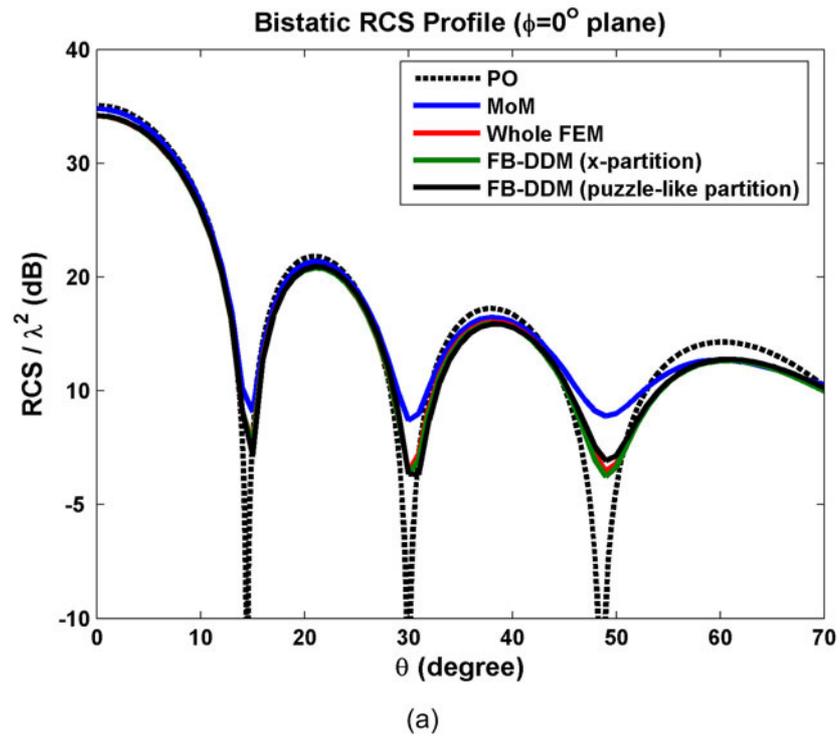


Figure 7.30. Bistatic RCS profile of square plate whose edge length is 4λ : (a) $\phi=0^\circ$ plane, (b) $\phi=90^\circ$ plane.

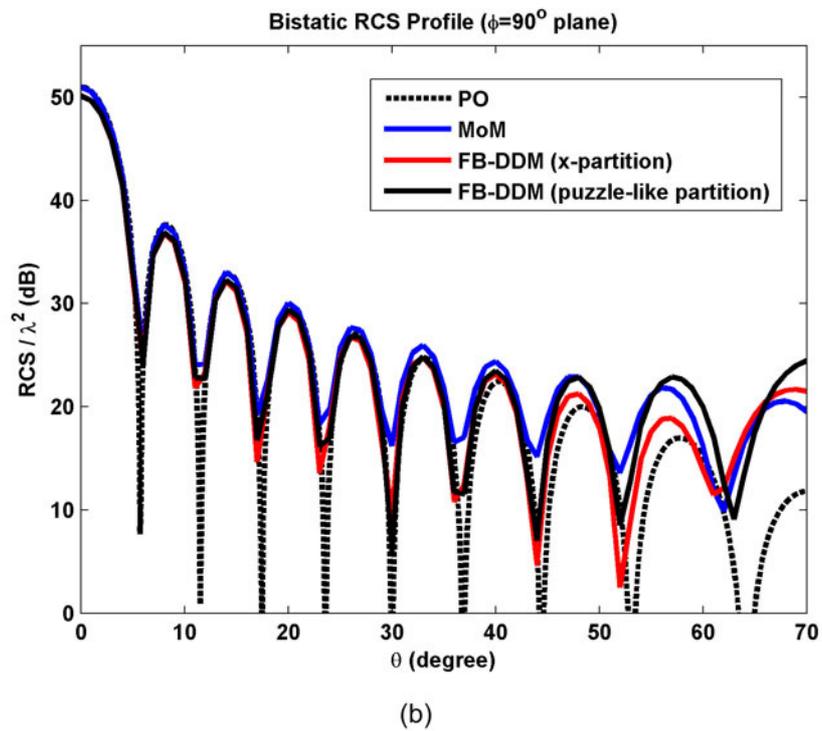
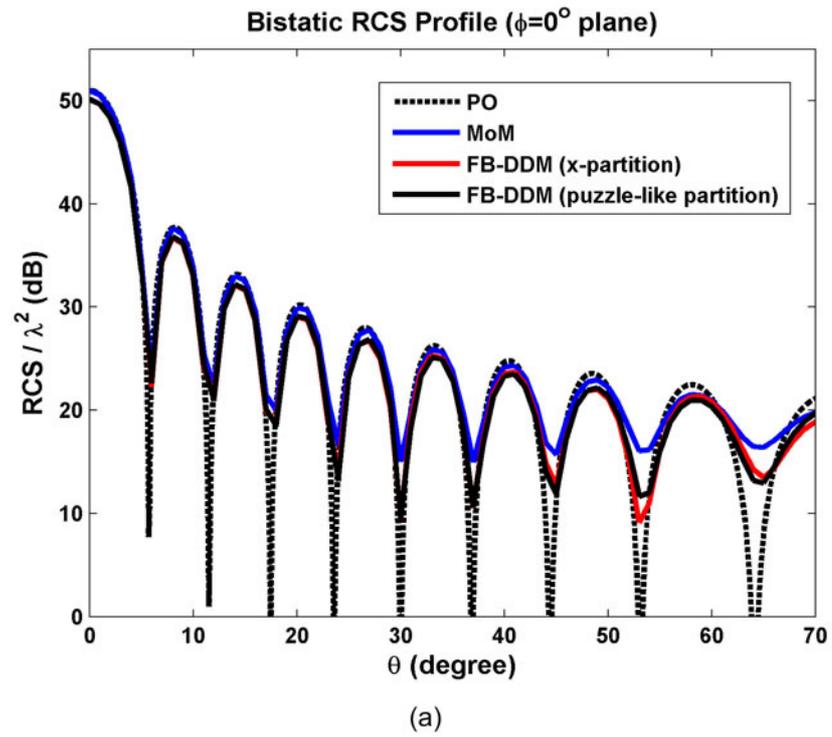


Figure 7.31. Bistatic RCS profile of square plate whose edge length is 10λ : (a) $\phi=0^\circ$ plane, (b) $\phi=90^\circ$ plane.

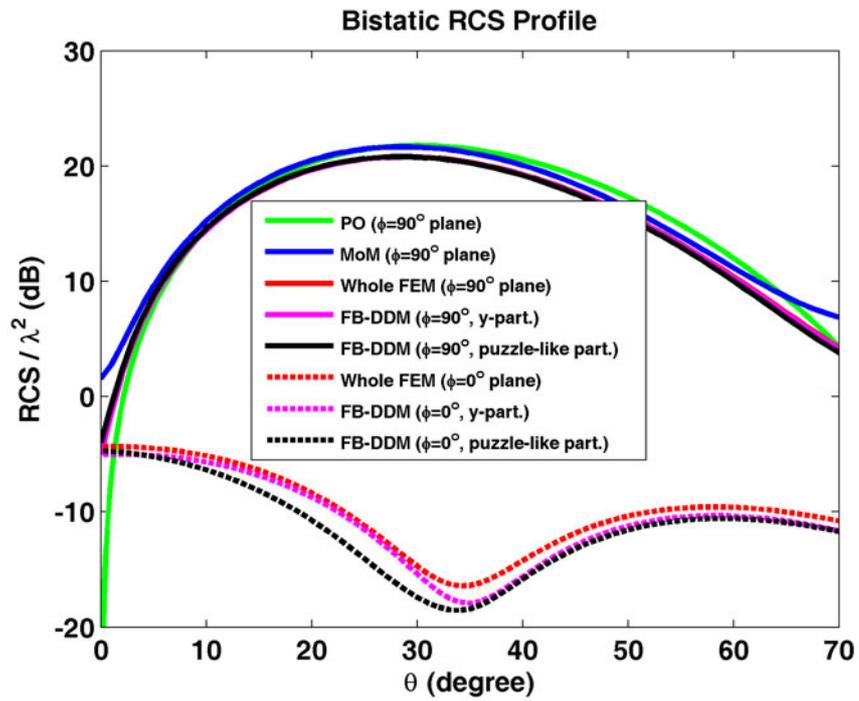


Figure 7.32. Bistatic RCS profile of square plate whose edge length is 2λ for oblique incidence.

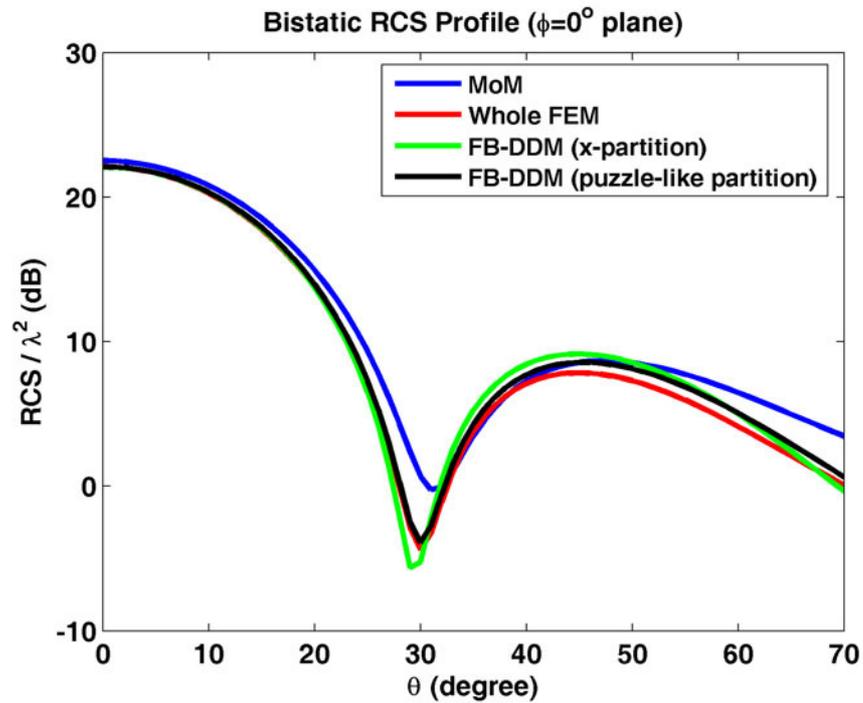


Figure 7.33. Bistatic RCS profile of square plate whose edge length is 2λ for normal incidence.

CHAPTER 8

NUMERICAL EXPERIMENTS (ILF-DDM)

In this chapter, we test the performance of the Iterative Leap-field DDM (ILF-DDM) algorithm via some 2D and 3D electromagnetic scattering problems.

We define a stopping (or threshold) criterion as follows

$$E^n = \max_{i=1,2,\dots,N} \|\vec{J}_i^n\| < \varepsilon \quad (8.1)$$

where \vec{J}_i^n is the equivalent current calculated over the scatterer boundary ($\partial\Omega_{Si}$) for the i^{th} subproblem at the n^{th} iteration. In this expression, $\|\cdot\|$ refers to the L_2 -norm, and the magnitude of the applied incident plane wave is taken as unity in all cases. Among all subproblems, we choose the largest value of the magnitude of the current (the worst case) in order to guarantee that all current values are below a threshold ε at the end of the algorithm. Since the magnitude of the induced current decays monotonically, if E^n is below a certain value, iterations are terminated. Based on our numerical experiments, $\varepsilon = 10^{-2}$ is sufficient for assuming that convergence is achieved since RCS values do not change significantly for smaller values of this constant.

The performance of the algorithm is tested in terms of the RCS calculations, and is compared with the results of the whole-domain FEM, as well as a MoM code in 2D simulations. In addition, in order to measure the near-field performance of the method, we introduce a mean-square error criterion as follows

$$\text{ERR} = \frac{\sum_{\Omega} |\vec{E}^{\text{DDM}} - \vec{E}^{\text{whole}}|^2}{\sum_{\Omega} |\vec{E}^{\text{whole}}|^2} \quad (\times 100) \quad (8.2)$$

where \vec{E}^{DDM} is the scattered field calculated by radiating the total currents along the scatterer boundaries at the n^{th} iteration of the ILF-DDM procedure into Ω (free-space), and \vec{E}^{whole} is the scattered field calculated by the whole-domain FEM. We should emphasize that there is no need to calculate the near-fields inside Ω in the ILF-DDM approach, but we calculate them in order to validate the performance of the ILF-DDM algorithm using the whole-domain FEM.

8.1. 2D Simulations

In this section, we test the ILF-DDM algorithm in electromagnetic scattering problems using infinitely-long PEC cylinders. In all examples, the locally-conformal PML method is implemented using the parameters $\alpha=7k$ and $m=3$, and the approximate PML thickness is chosen as $\lambda/4$. The wavenumber k is set to 20π (i.e., the wavelength λ is 0.1 meter) unless otherwise stated. The element size is approximately set to $\lambda/60$. The incident plane wave is assumed to be $\vec{E}^{\text{inc}} = \hat{a}_z \exp\left[jk(x \cos \varphi^{\text{inc}} + y \sin \varphi^{\text{inc}})\right]$ where φ^{inc} is the angle of incidence.

The 1st example is a scattering problem where a plane wave ($\varphi^{\text{inc}} = 135^\circ$) is incident to an infinitely-long PEC cylinder having a ‘thin’ L-shaped cross-section whose both edge lengths are 8λ and thickness is $\lambda/20$ (see Figure 8.1). The original problem is decomposed into two subproblems, as shown in Figure 5.15. Then, the algorithm converges at the 3rd iteration. We plot; (i) the E^n and ERR values versus the number of iterations (i.e., convergence profile) in Figure 8.2, (ii) the magnitude of the total current density along the scatterer boundary in Figure 8.3, (iii) the bistatic RCS profiles in Figure 8.4, (iv) the contours of the scattered field and the total field for the whole domain FEM solution and the

ILF-DDM algorithm at the end of the 3rd iteration in Figure 8.5. We follow a similar approach in the remaining examples to present the results. Finally, in Figure 8.6, we plot the scattered field contours at the end of first and second iterations and compare them with the GO and UTD approaches. In this figure, we illustrate that the field which leaps in each iteration denotes the field interactions (such as multiple-order of reflections and diffractions) between the corresponding subproblems, depending on the frequency, plane wave illumination, geometry, etc. If a ‘large’ L-shaped cylinder is illuminated as in Figure 8.1, based on the ray optics interpretation, the plane wave reflects from one of the two sections, then reflects from the other one, and finally leaves the scatterer. Therefore, the algorithm is expected to converge after at most two or three iterations. The result of the first iteration of the ILF-DDM algorithm (see Figure 8.6(a)) is similar to the GO method where the first-order reflected field is included (i.e., the ray of incident field reflects from the scatterer only once, see Figure 8.6(b)), or the sum of the first-order reflected field in GO and first-order diffracted fields in UTD approaches (i.e., the ray of incident field diffracts from the two edges, see Figure 8.6(c)). Similarly, we note the resemblance of the second iteration of the ILF-DDM algorithm in Figure 8.6(d), and Figure 8.6(e) where we plot the sum of the first- and second-order reflected fields in GO (i.e., the ray reflects twice) and first- and second-order diffracted fields in UTD approaches (i.e., both the incident ray and the first reflected ray diffract from the two edges). The higher-order terms of the GO and UTD provide a negligible contribution to the results due to their rapid decaying behaviour, as well as the third iteration does *not* contribute to the overall result considerably in the ILF-DDM algorithm.

The 2nd example is a scattering problem where a plane wave ($\varphi^{inc} = 180^\circ$) is incident to a U-shaped cylinder (see Figure 8.7). We set λ as 0.12 meter. The original problem is decomposed into three subproblems, as shown in Figure 5.16. Then, the algorithm converges at the 10th iteration. We plot; (i) the

convergence profile in Figure 8.8, (ii) the magnitude of the total current density along the scatterer boundary in Figure 8.9, (iii) the bistatic RCS profiles in Figure 8.10, (iv) the contours of the scattered and the total fields in Figure 8.11. In this example, the algorithm requires an increased number of iterations in order to satisfy the convergence criterion, due to the strong field interactions inside the cylinder. Even some resonance behavior is observed in some parts of the open cavity structure. This application yields a sequence of slowly-converging iterates due to the multiple reflections occurring in the cavity, but it is still possible to obtain an approximate solution close to the whole domain FEM solution.

The 3rd example considers again a U-shaped cylinder, but we increase the wavelength (λ) slightly and decrease the depth of the cylinder (see Figure 8.12). We set λ as 0.15 meter. The aim of this problem is to show that as the field interactions between different portions of the scatterer decrease, the algorithm converges faster. Then, the algorithm converges at the 4th iteration. We plot; (i) the convergence profile in Figure 8.13, (ii) the magnitude of the total current density in Figure 8.14, (iii) the bistatic RCS profiles in Figure 8.15, (iv) the contours of the scattered and the total fields in Figure 8.16. We conclude that the convergence rate is better than the second example due to less field interactions.

The 4th example is a scattering problem where a plane wave ($\varphi^{inc} = 135^\circ$) is incident to two infinitely-long cylinders having rectangular cross-sections (see Figure 8.17). The original problem is decomposed into two subproblems, similar to Figure 5.17. Then, the algorithm converges at the 3rd iteration. We plot; (i) the convergence profile in Figure 8.18, (ii) the bistatic RCS profiles in Figure 8.19, (iii) the contours of the scattered field and the total field in Figure 8.20.

The 5th example is a scattering problem where a plane wave ($\varphi^{inc} = 135^\circ$) is incident to three infinitely-long PEC cylinders having rectangular cross-sections (see Figure 8.21). We set λ as 0.12 meter. The algorithm converges at the 7th

iteration. We plot; (i) the convergence profile in Figure 8.22, (ii) the bistatic RCS profiles in Figure 8.23, (iii) the contours of the scattered field and the total field in Figure 8.24.

The 6th example is a scattering problem where a plane wave ($\varphi^{inc} = 90^\circ$) is incident to two circular PEC cylinders (see Figure 8.25). The algorithm converges at the 4th iteration. We plot; (i) the convergence profile in Figure 8.26, (ii) the bistatic RCS profiles in Figure 8.27, (iii) the contours of the scattered field and the total field in Figure 8.28.

The 7th example is a scattering problem where a plane wave ($\varphi^{inc} = 30^\circ$) is incident to a rectangular and a circular PEC cylinders (see Figure 8.29). The algorithm converges at the 5th iteration. We plot; (i) the convergence profile in Figure 8.30, (ii) the bistatic RCS profiles in Figure 8.31, (iii) the contours of the scattered field and the total field in Figure 8.32.

The 8th example is a scattering problem where a plane wave ($\varphi^{inc} = 45^\circ$) is incident to an elliptical and a circular PEC cylinders (see Figure 8.33). The algorithm converges at the 5th iteration. We plot; (i) the convergence profile in Figure 8.34, (ii) the bistatic RCS profiles in Figure 8.35, (iii) the contours of the scattered field and the total field in Figure 8.36.

The 9th example is a scattering problem where a plane wave ($\varphi^{inc} = 90^\circ$) is incident to three circular PEC cylinders (see Figure 8.37). The algorithm converges at the 7th iteration. We plot; (i) the convergence profile in Figure 8.38, (ii) the bistatic RCS profiles in Figure 8.39, (iii) the contours of the scattered field and the total field in Figure 8.40. Finally, we plot the convergence profiles of all examples in Figure 8.41 for the purpose of comparison. We present some conclusions about the numerical results in Section 8.3.

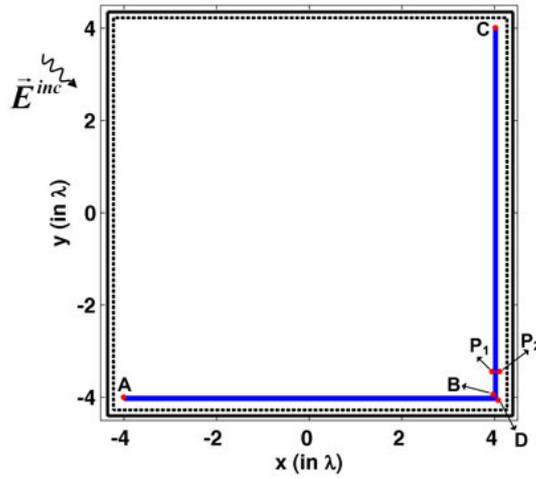


Figure 8.1. Original problem of example 1 (thin L-shaped cylinder, length 8λ).

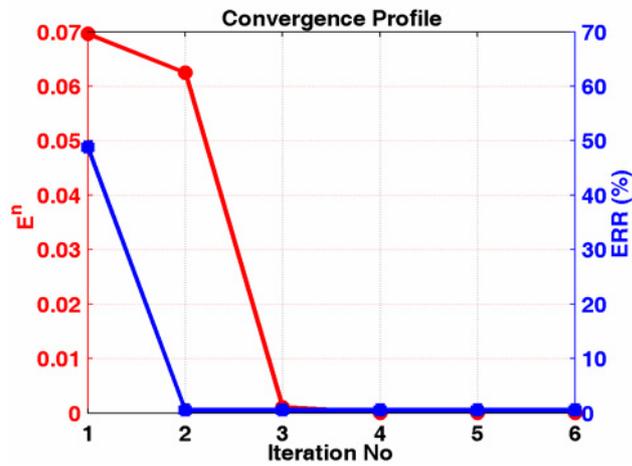


Figure 8.2. Convergence profile of example 1.

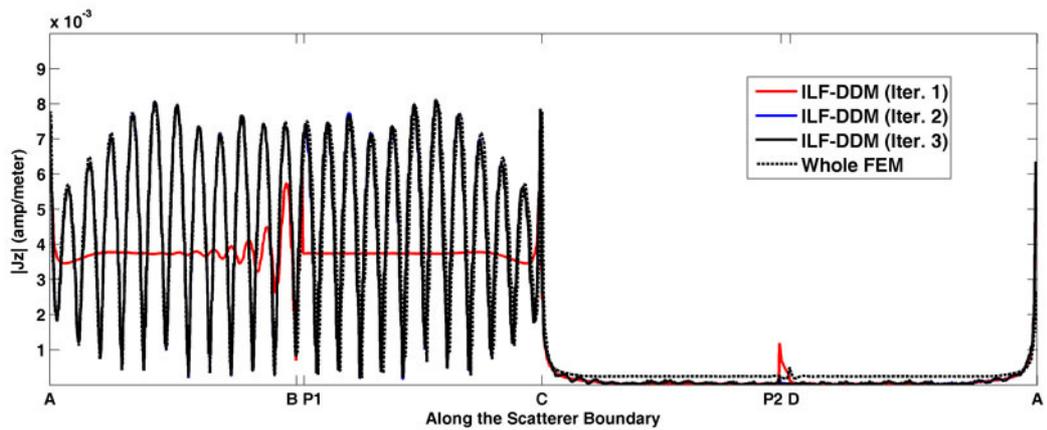


Figure 8.3. Magnitude of the total current density along the boundary of the L-shaped cylinder (example 1).

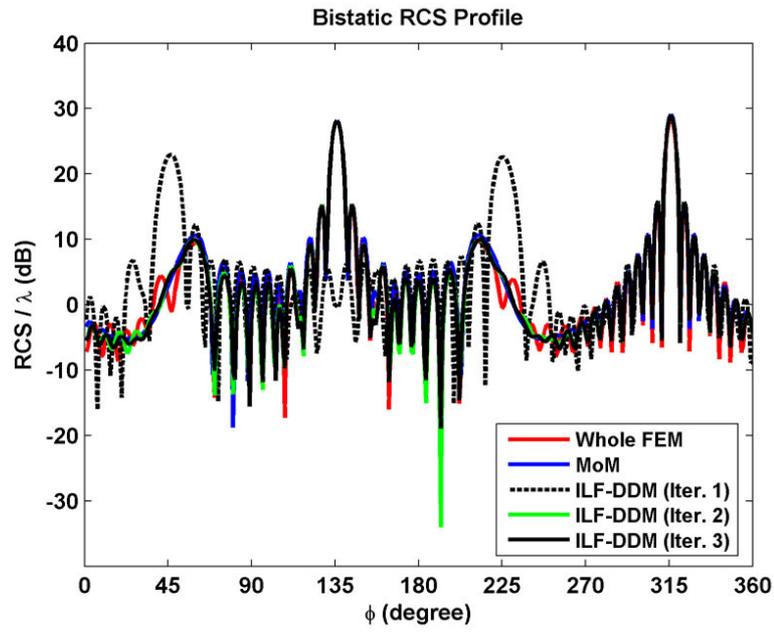


Figure 8.4. Bistatic RCS profile of the L-shaped cylinder (example 1).

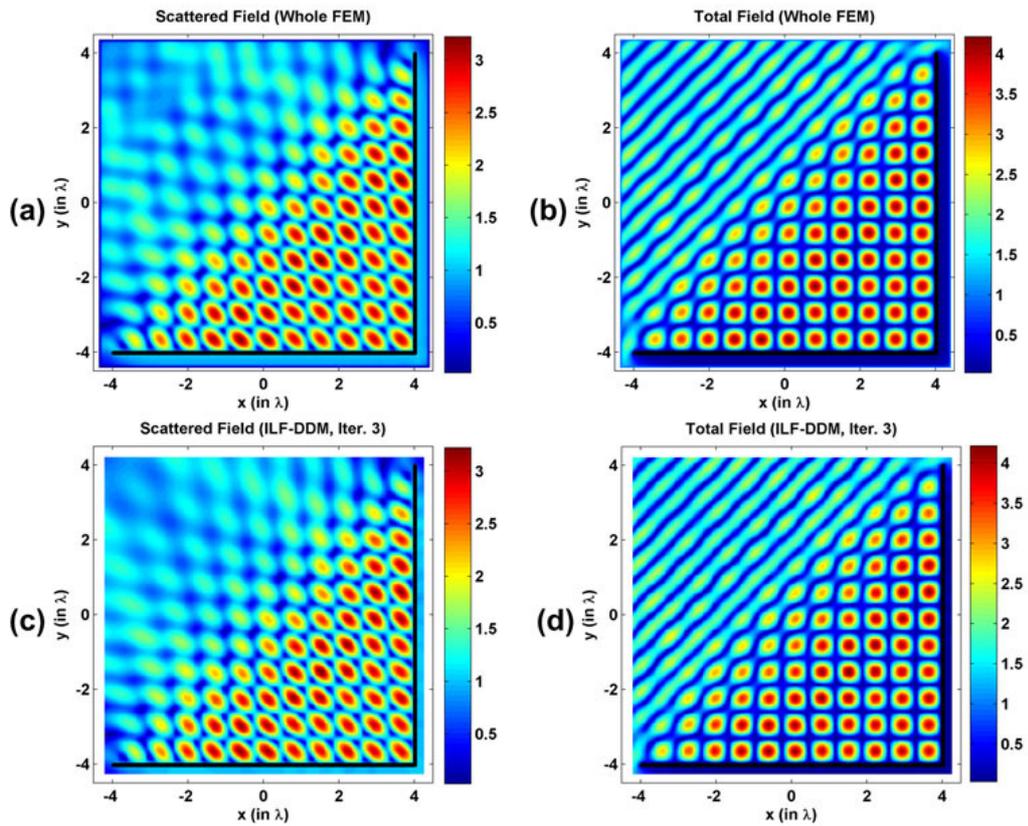


Figure 8.5. Field contours in example 1 (magnitude): (a-b) Scattered and total fields of the whole FEM, (c-d) Scattered and total fields in the 3rd iteration.

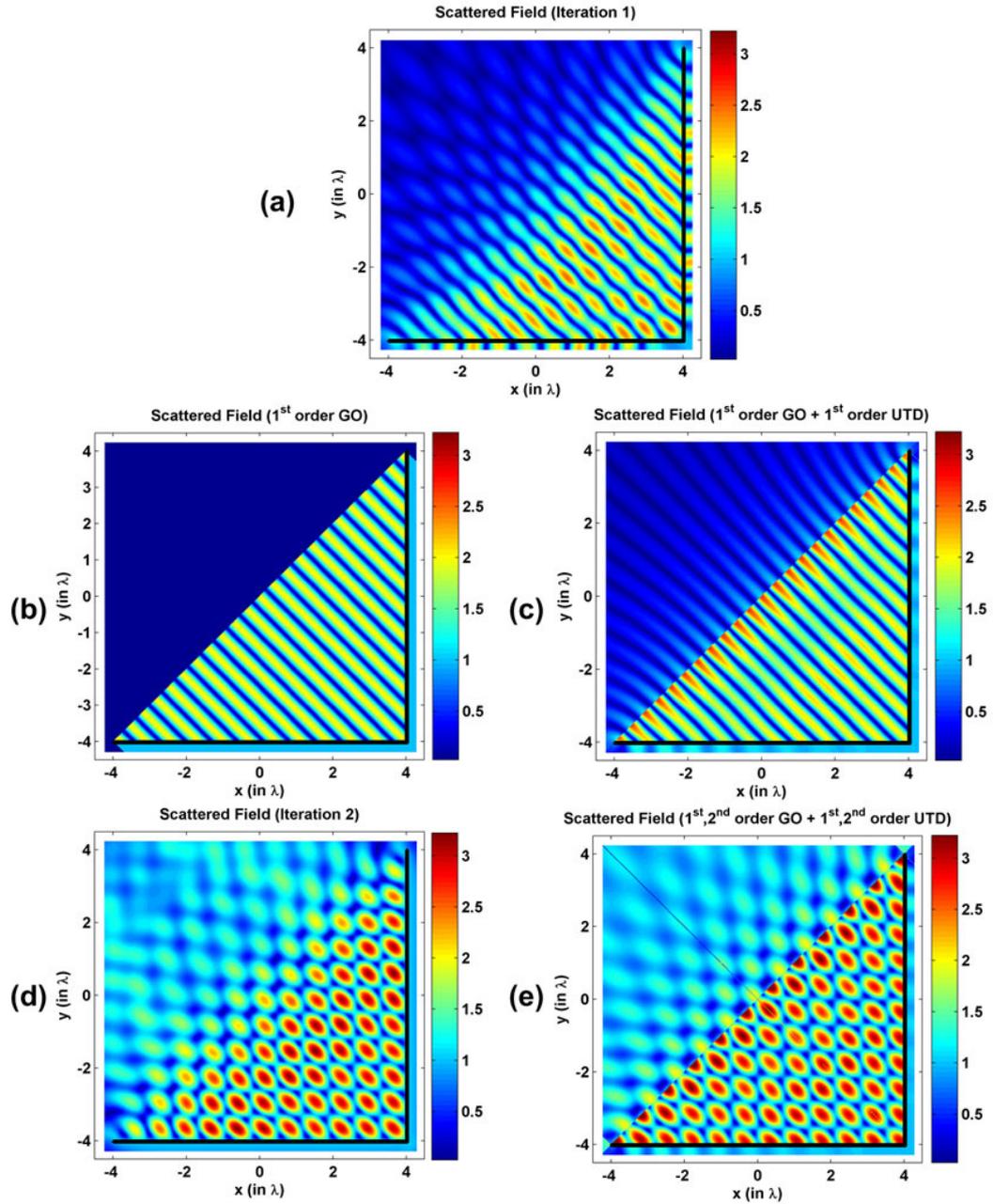


Figure 8.6. Scattered field contours in example 1 compared to GO and UTD results (magnitude): (a) 1st iteration of the ILF-DDM, (b) GO contribution with only 1st order reflected field, (c) Sum of GO contribution with only 1st order reflected field and UTD contribution with only 1st order diffracted field, (d) 2nd iteration of the ILF-DDM, (e) Sum of GO contributions with 1st and 2nd order reflected fields, and UTD contributions with 1st and 2nd order diffracted fields.

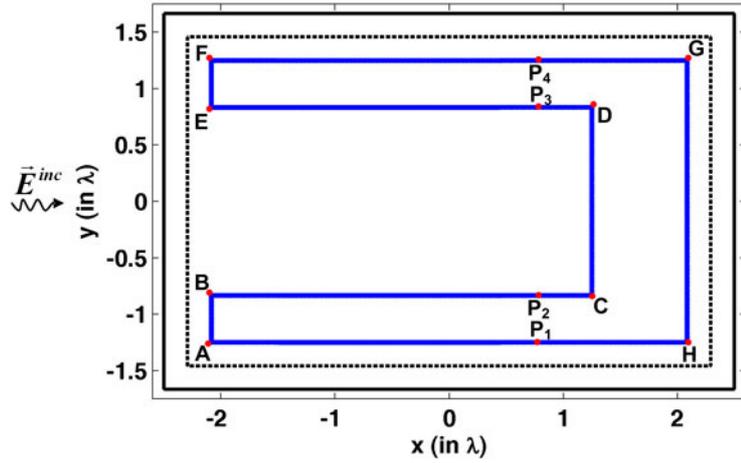


Figure 8.7. Original problem of example 2 (U-shaped cylinder).

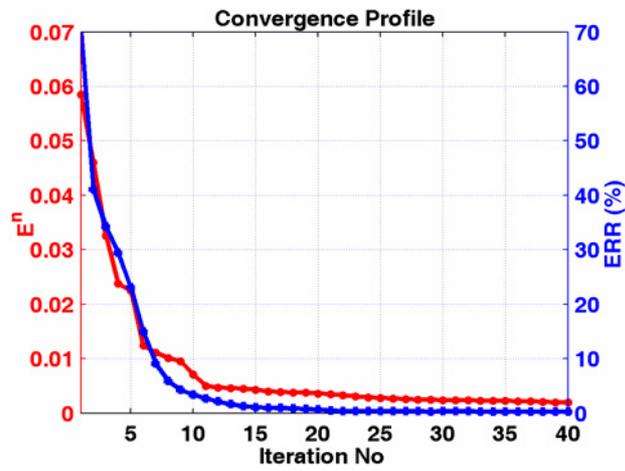


Figure 8.8. Convergence profile of example 2.

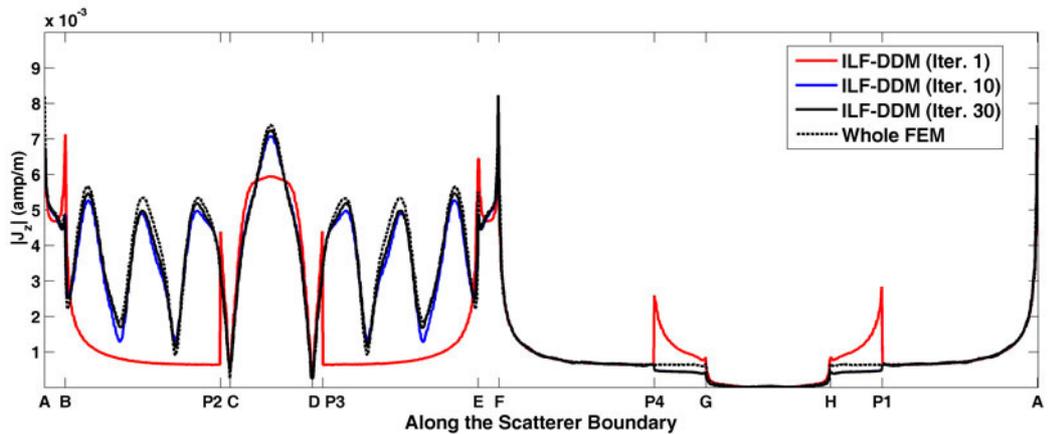


Figure 8.9. Magnitude of the total current density along the boundary of the U-shaped cylinder (example 2).

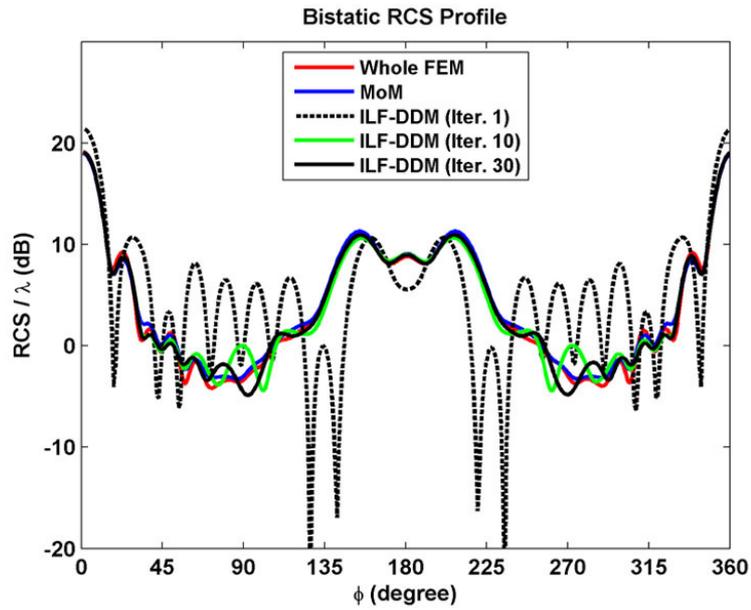


Figure 8.10. Bistatic RCS profile of the U-shaped cylinder (example 2).

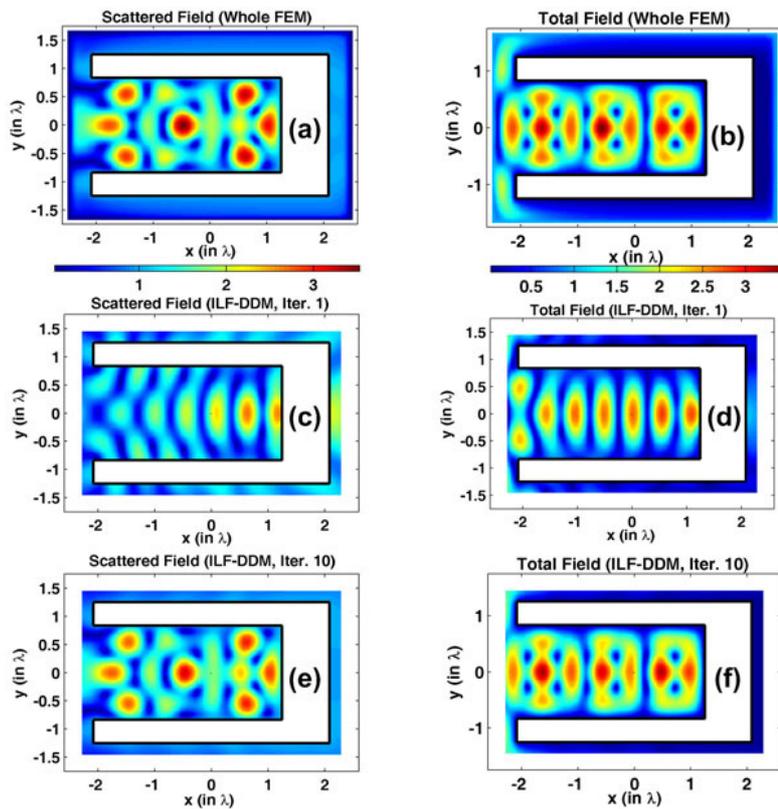


Figure 8.11. Field contours in example 2 (magnitude): (a-b) Scattered and total fields of the whole FEM, (c-d) Scattered and total fields in the 1st iteration, (e-f) Scattered and total fields in the 10th iteration.

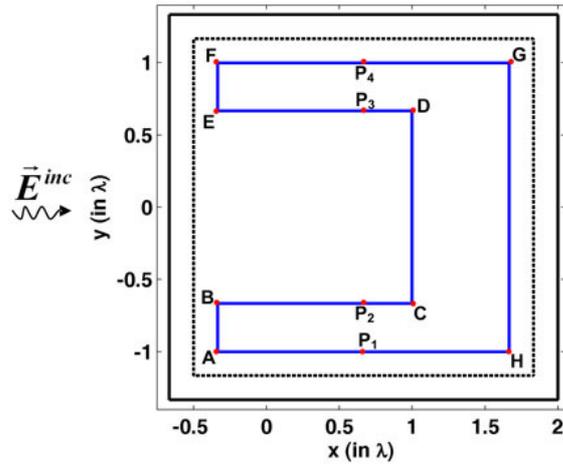


Figure 8.12. Original problem of example 3 (short U-shaped cylinder).

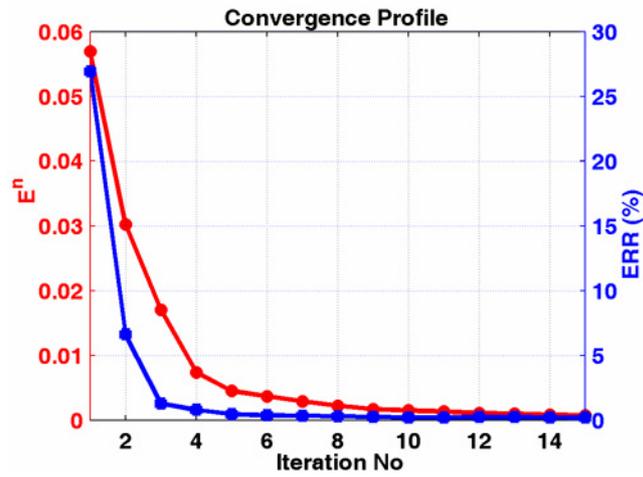


Figure 8.13. Convergence profile of example 3.

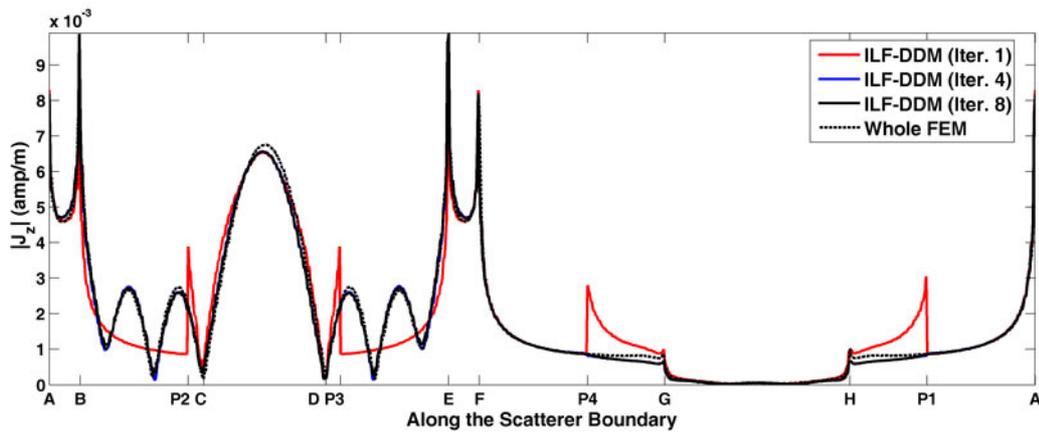


Figure 8.14. Magnitude of the total current density along the boundary of the short U-shaped cylinder (example 3).

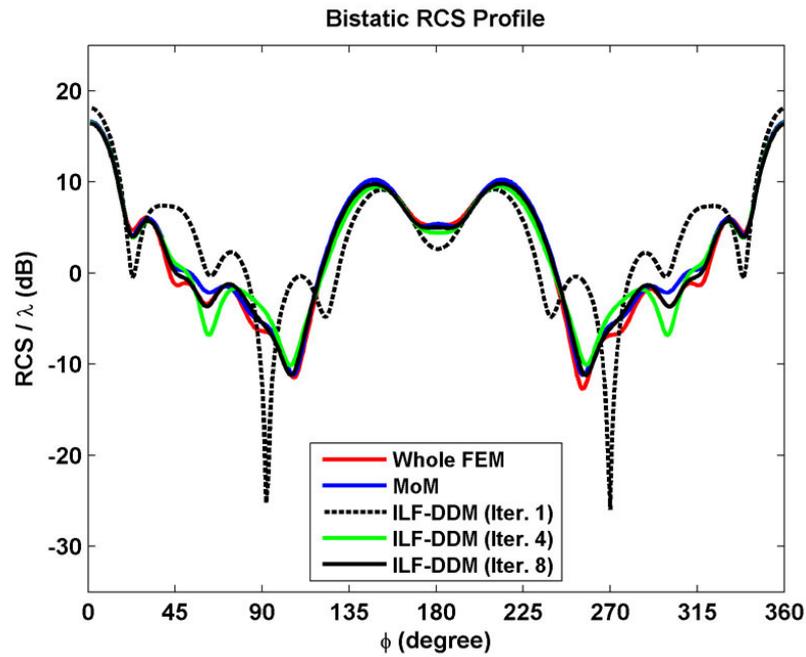


Figure 8.15. Bistatic RCS profile of the short U-shaped cylinder (example 3).

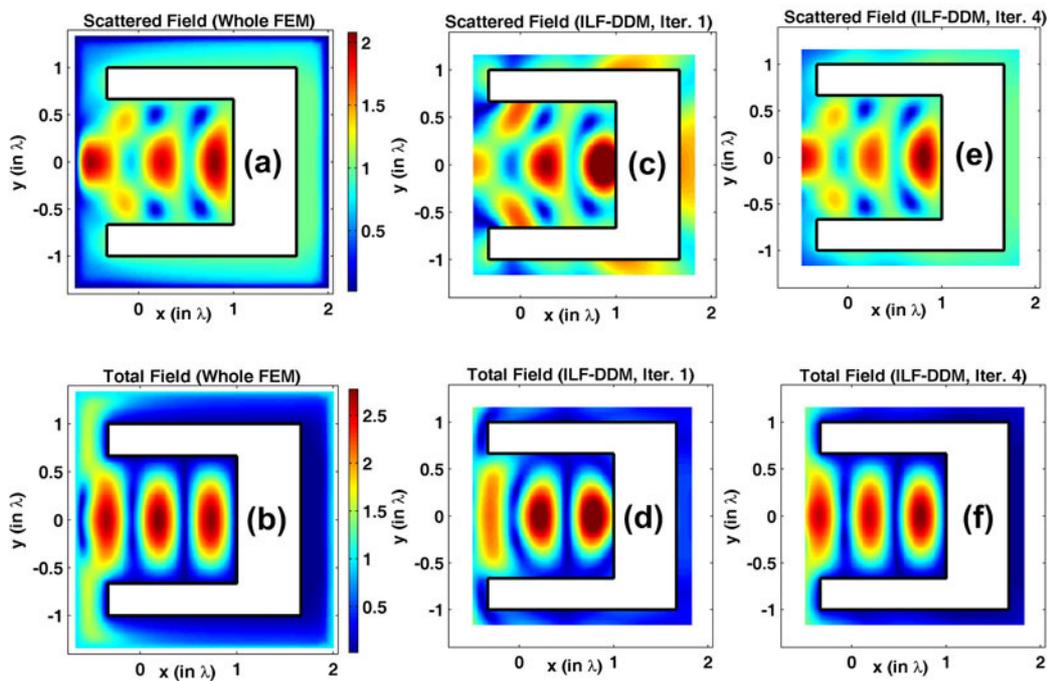


Figure 8.16. Field contours in example 3 (magnitude): (a-b) Scattered and total fields of the whole FEM, (c-d) Scattered and total fields in the 1st iteration, (e-f) Scattered and total fields in the 4th iteration.

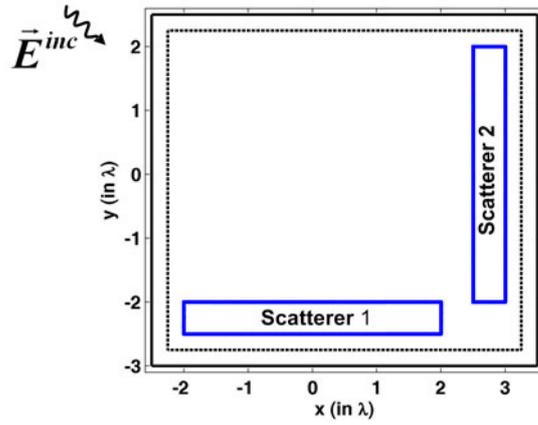


Figure 8.17. Original problem of example 4.

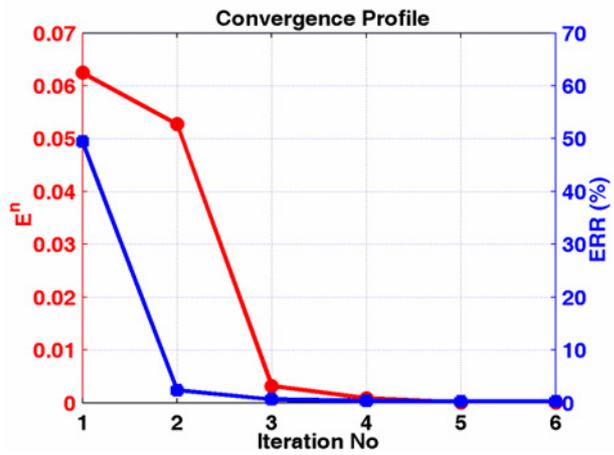


Figure 8.18. Convergence profile of example 4.

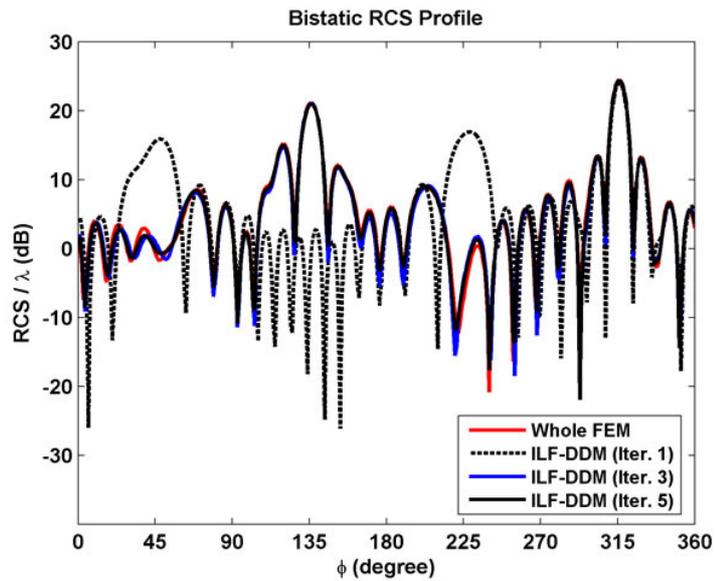


Figure 8.19. Bistatic RCS profile in example 4.

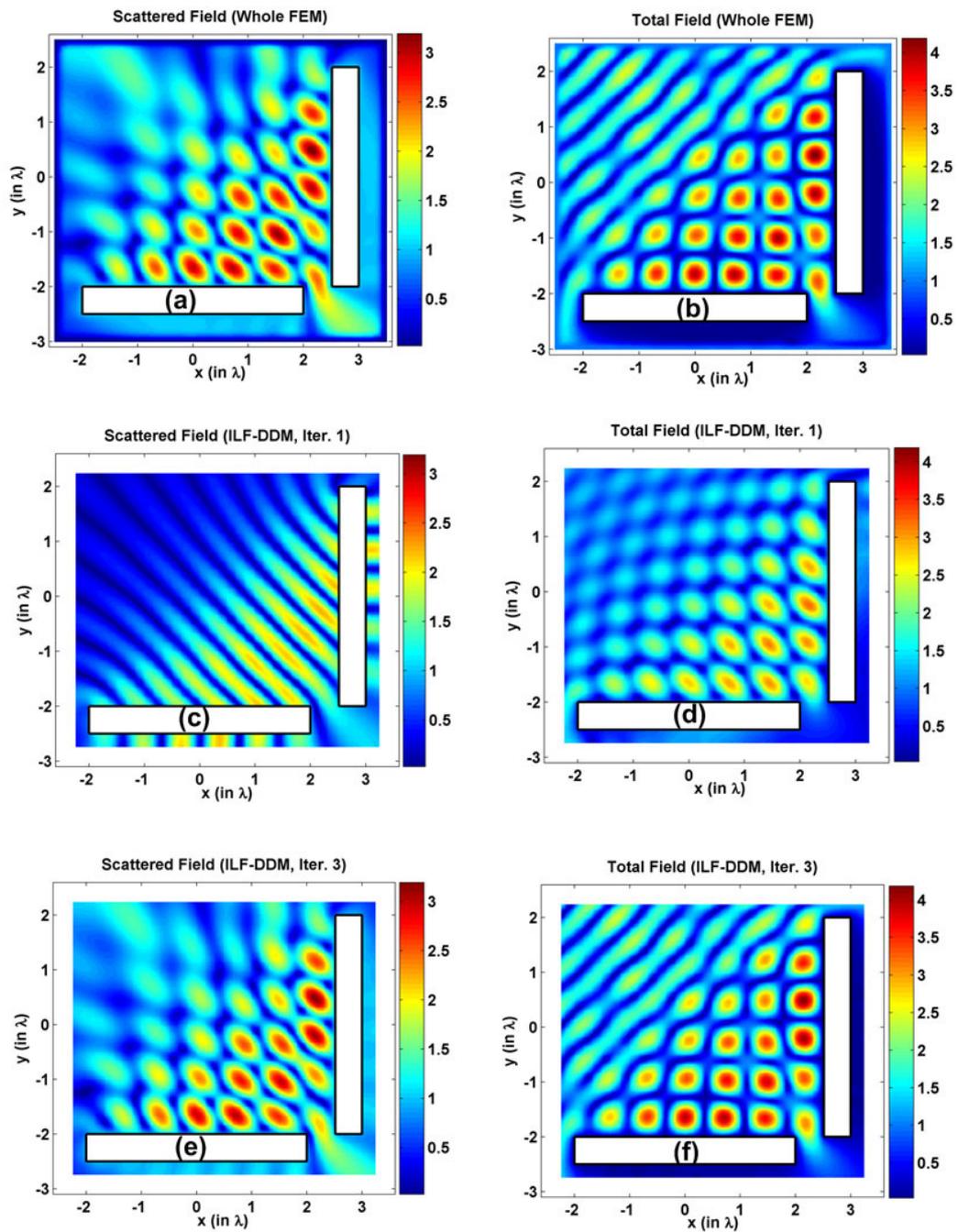


Figure 8.20. Field contours in example 4 (magnitude): (a-b) Scattered and total fields of the whole FEM, (c-d) Scattered and total fields in the 1st iteration, (e-f) Scattered and total fields in the 3rd iteration.

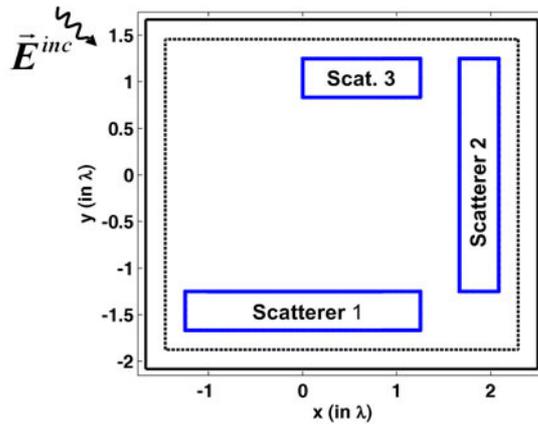


Figure 8.21. Original problem of example 5.

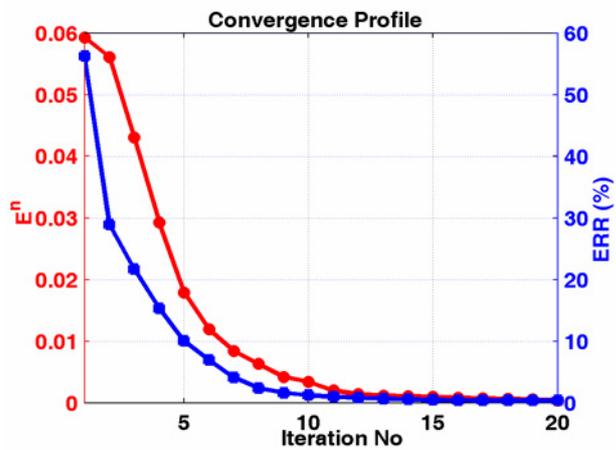


Figure 8.22. Convergence profile of example 5.

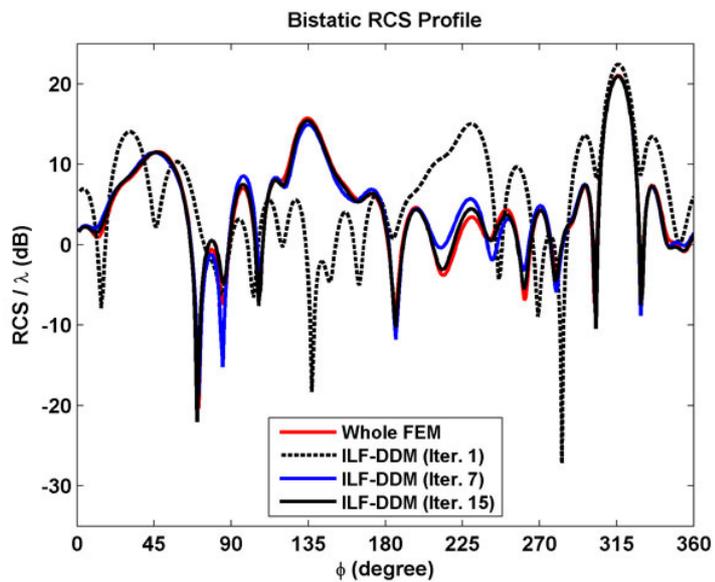


Figure 8.23. Bistatic RCS profile in example 5.

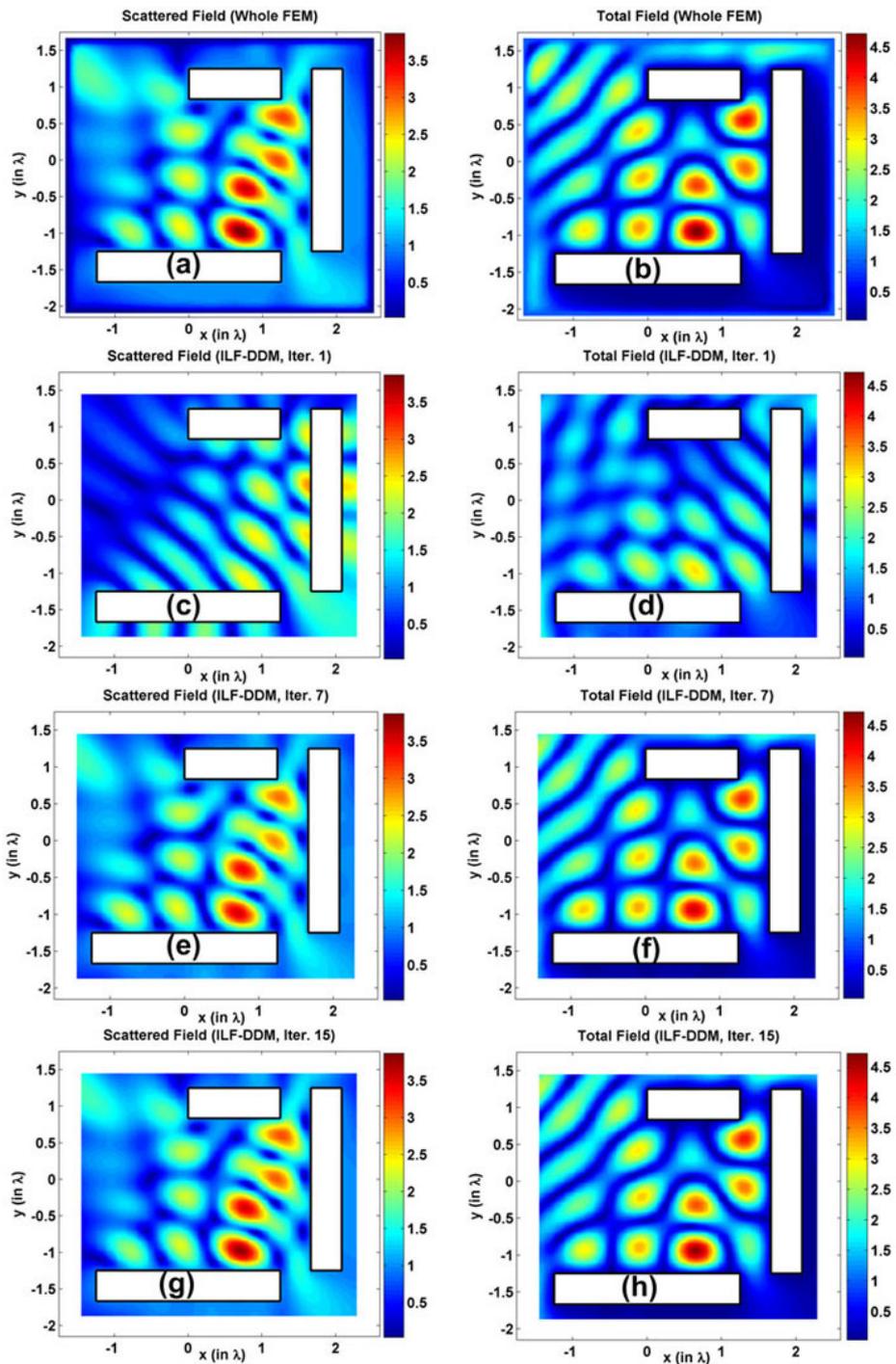


Figure 8.24. Field contours in example 5 (magnitude): (a-b) Scattered and total fields of the whole FEM, (c-d) Scattered and total fields in the 1st iteration, (e-f) Scattered and total fields in the 7th iteration, (g-h) Scattered and total fields in the 15th iteration.

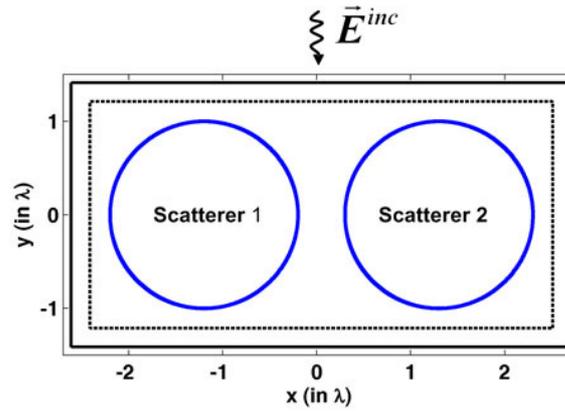


Figure 8.25. Original problem of example 6.

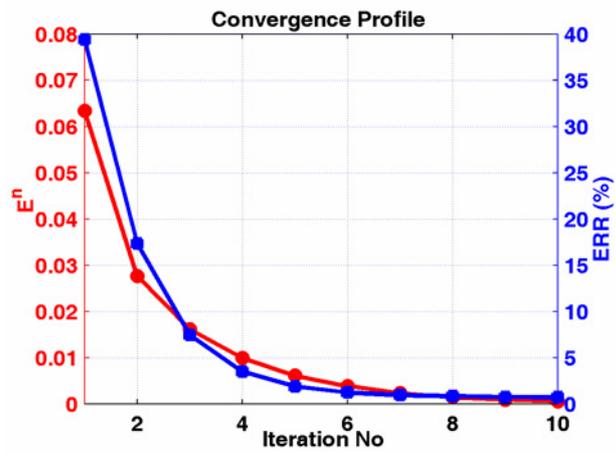


Figure 8.26. Convergence profile of example 6.

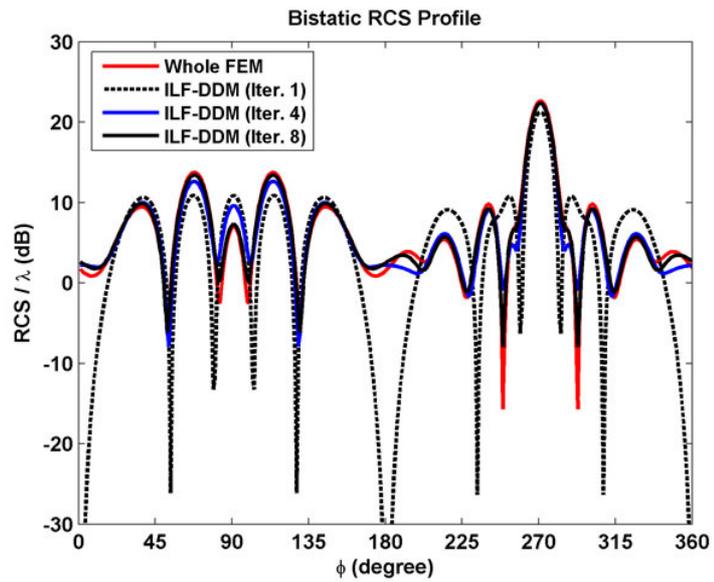


Figure 8.27. Bistatic RCS profile in example 6.

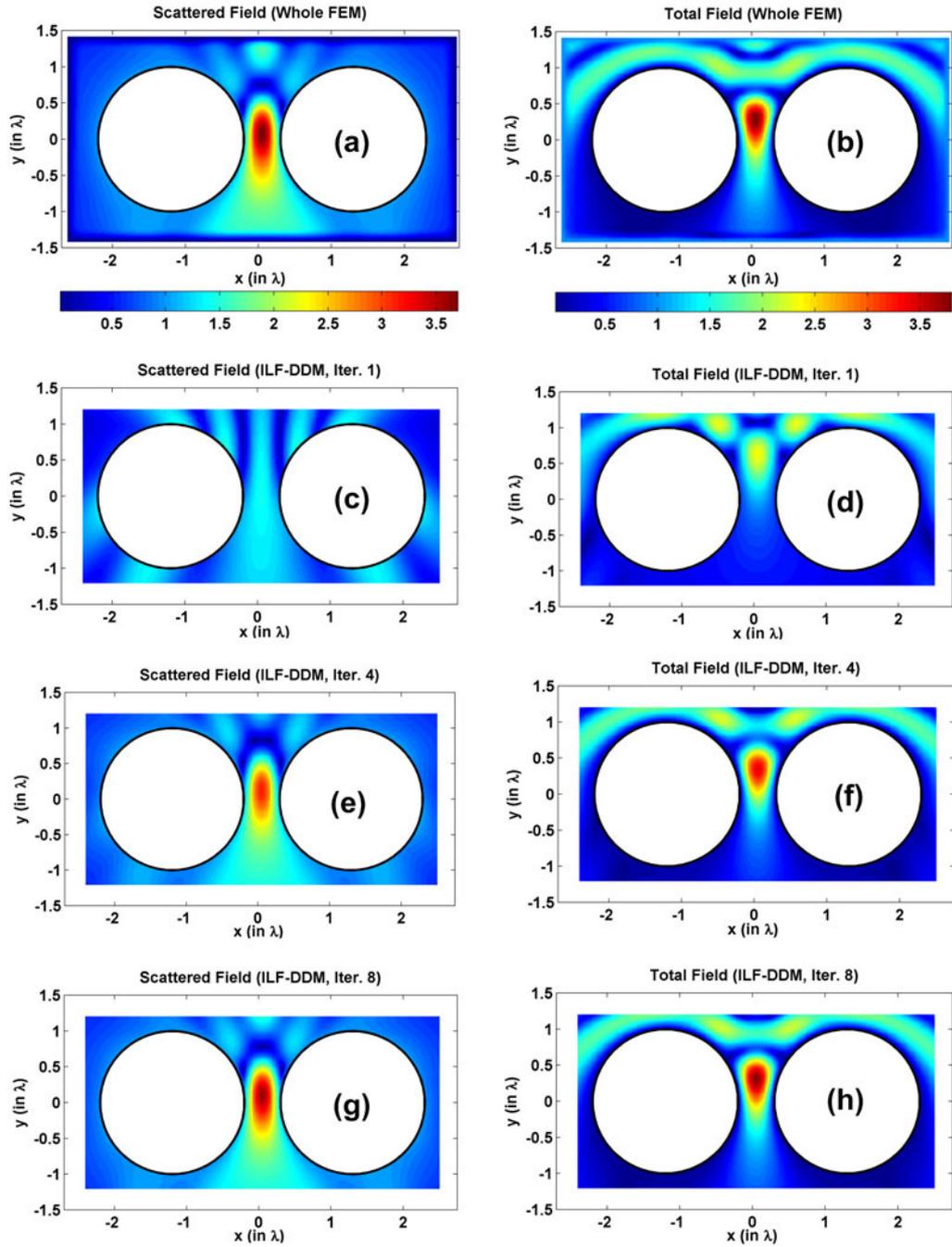


Figure 8.28. Field contours in example 6 (magnitude): (a-b) Scattered and total fields of the whole FEM, (c-d) Scattered and total fields in the 1st iteration, (e-f) Scattered and total fields in the 4th iteration, (g-h) Scattered and total fields in the 8th iteration.

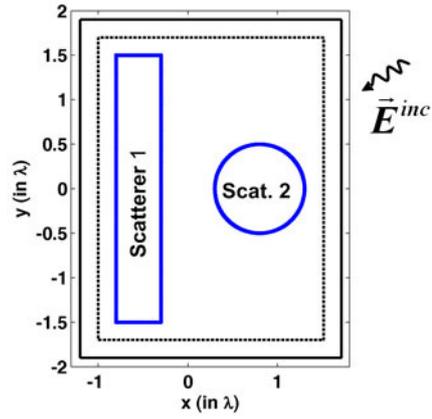


Figure 8.29. Original problem of example 7.

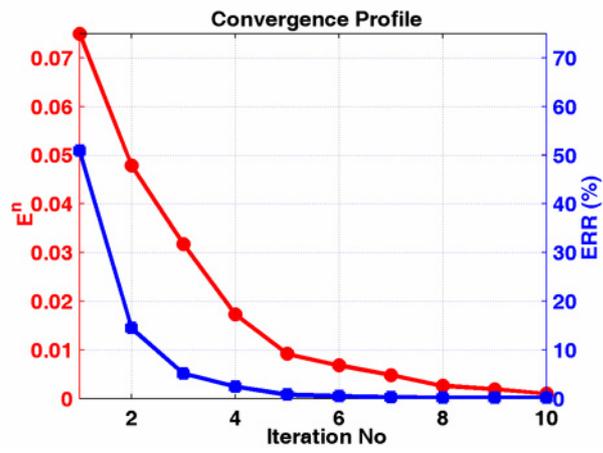


Figure 8.30. Convergence profile of example 7.

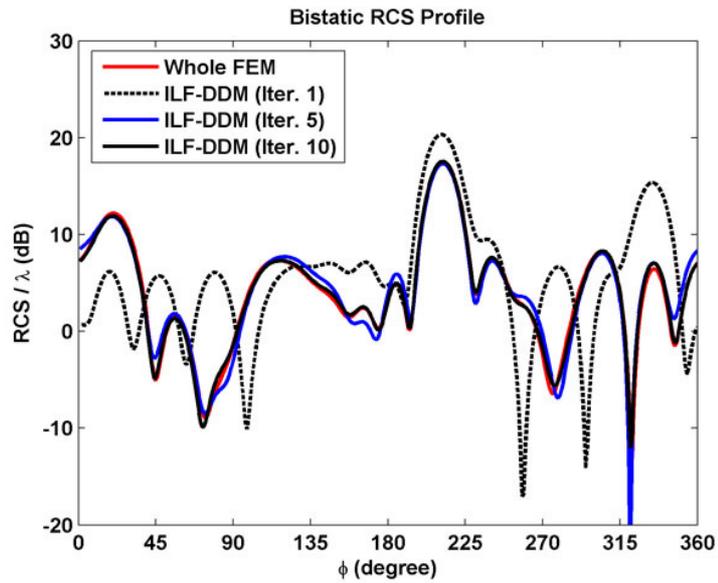


Figure 8.31. Bistatic RCS profile in example 7.

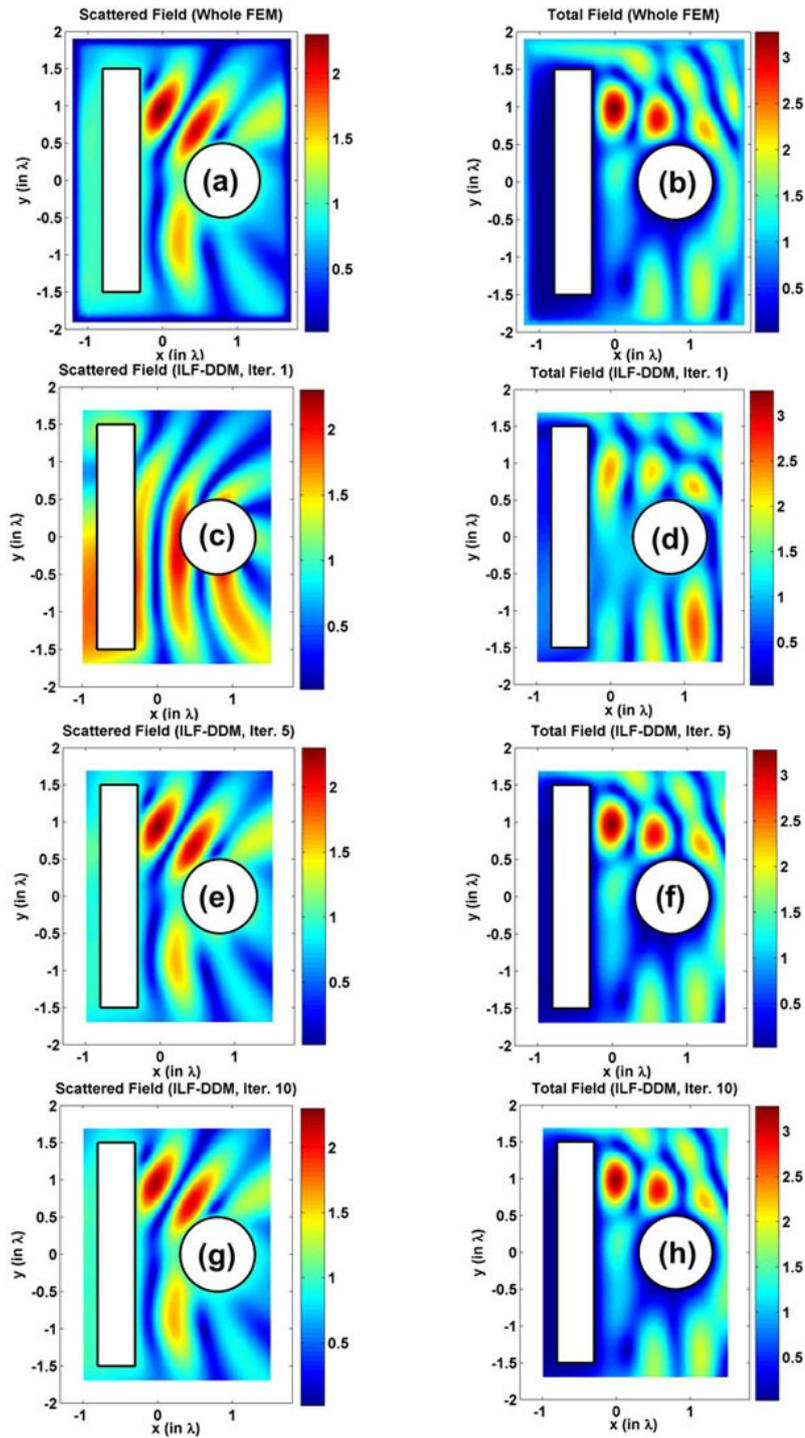


Figure 8.32. Field contours in example 7 (magnitude): (a-b) Scattered and total fields of the whole FEM, (c-d) Scattered and total fields in the 1st iteration, (e-f) Scattered and total fields in the 5th iteration, (g-h) Scattered and total fields in the 10th iteration.

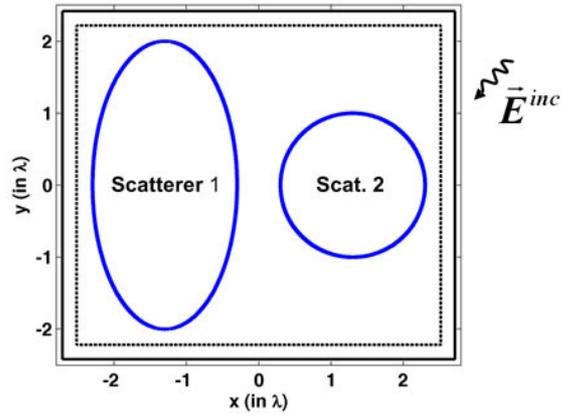


Figure 8.33. Original problem of example 8.

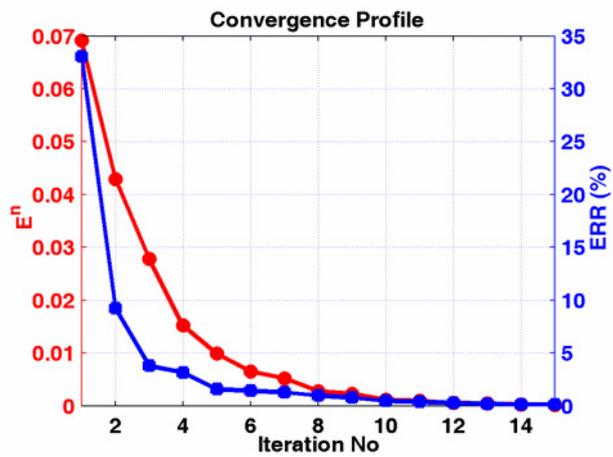


Figure 8.34. Convergence profile of example 8.

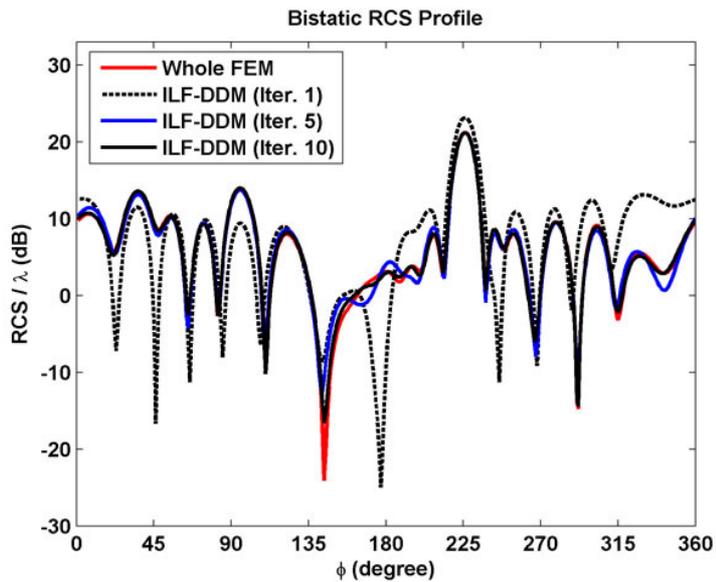


Figure 8.35. Bistatic RCS profile in example 8.

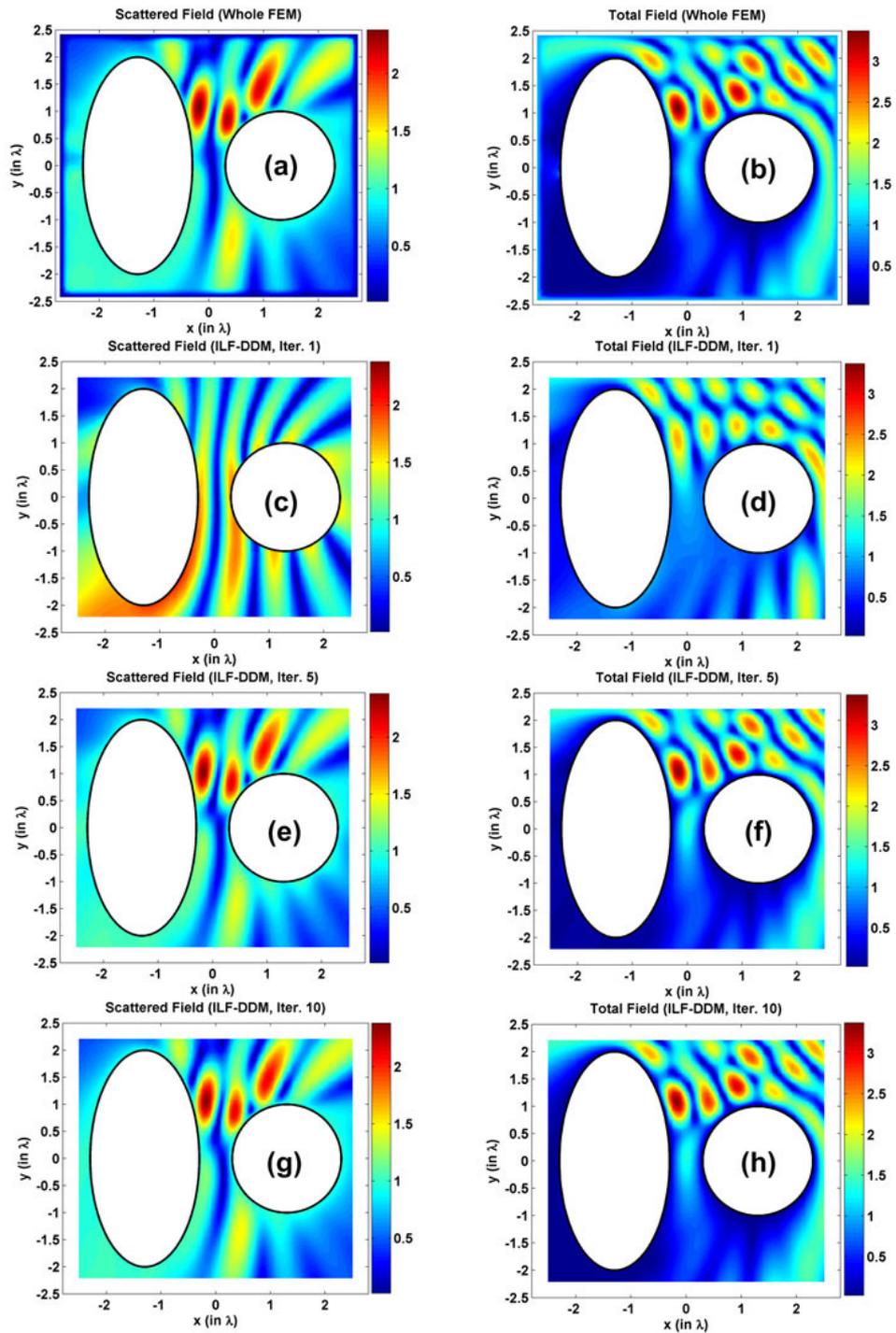


Figure 8.36. Field contours in example 8 (magnitude): (a-b) Scattered and total fields of the whole FEM, (c-d) Scattered and total fields in the 1st iteration, (e-f) Scattered and total fields in the 5th iteration, (g-h) Scattered and total fields in the 10th iteration.

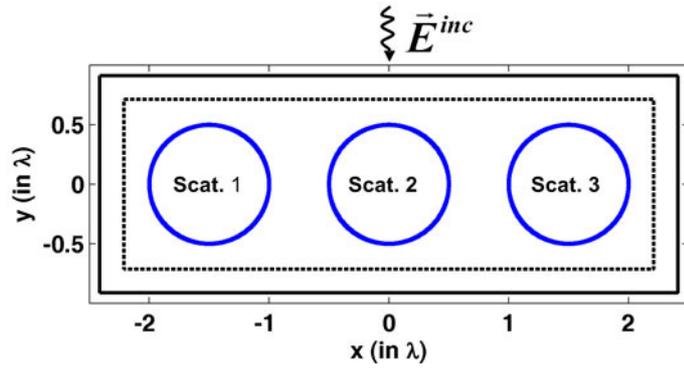


Figure 8.37. Original problem of example 9.

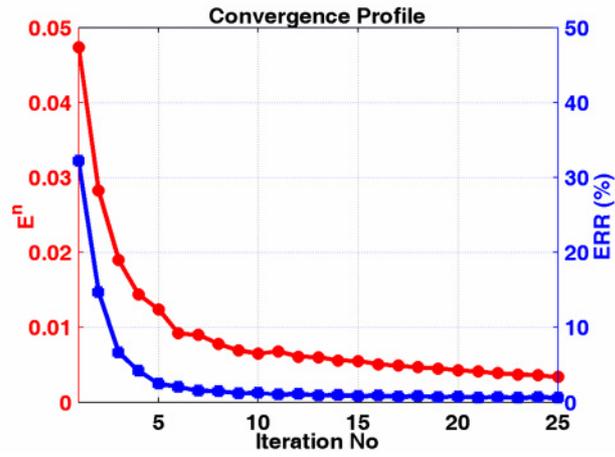


Figure 8.38. Convergence profile of example 9.

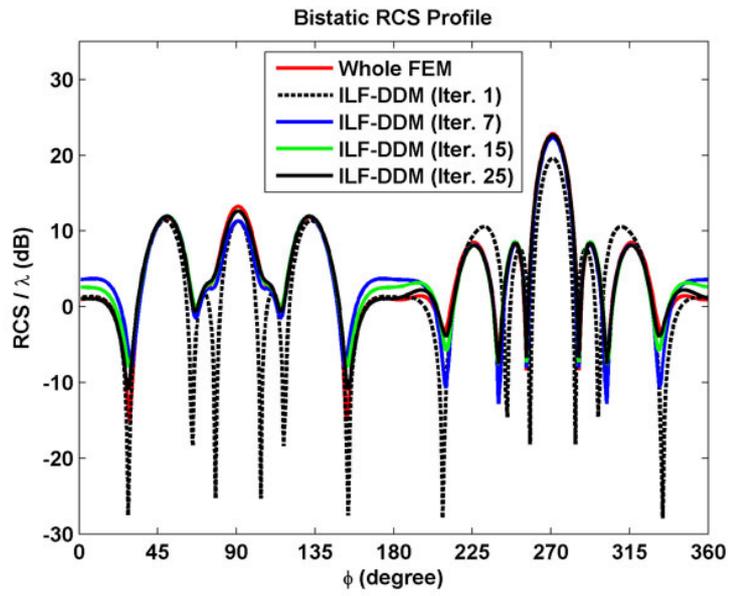


Figure 8.39. Bistatic RCS profile in example 9.

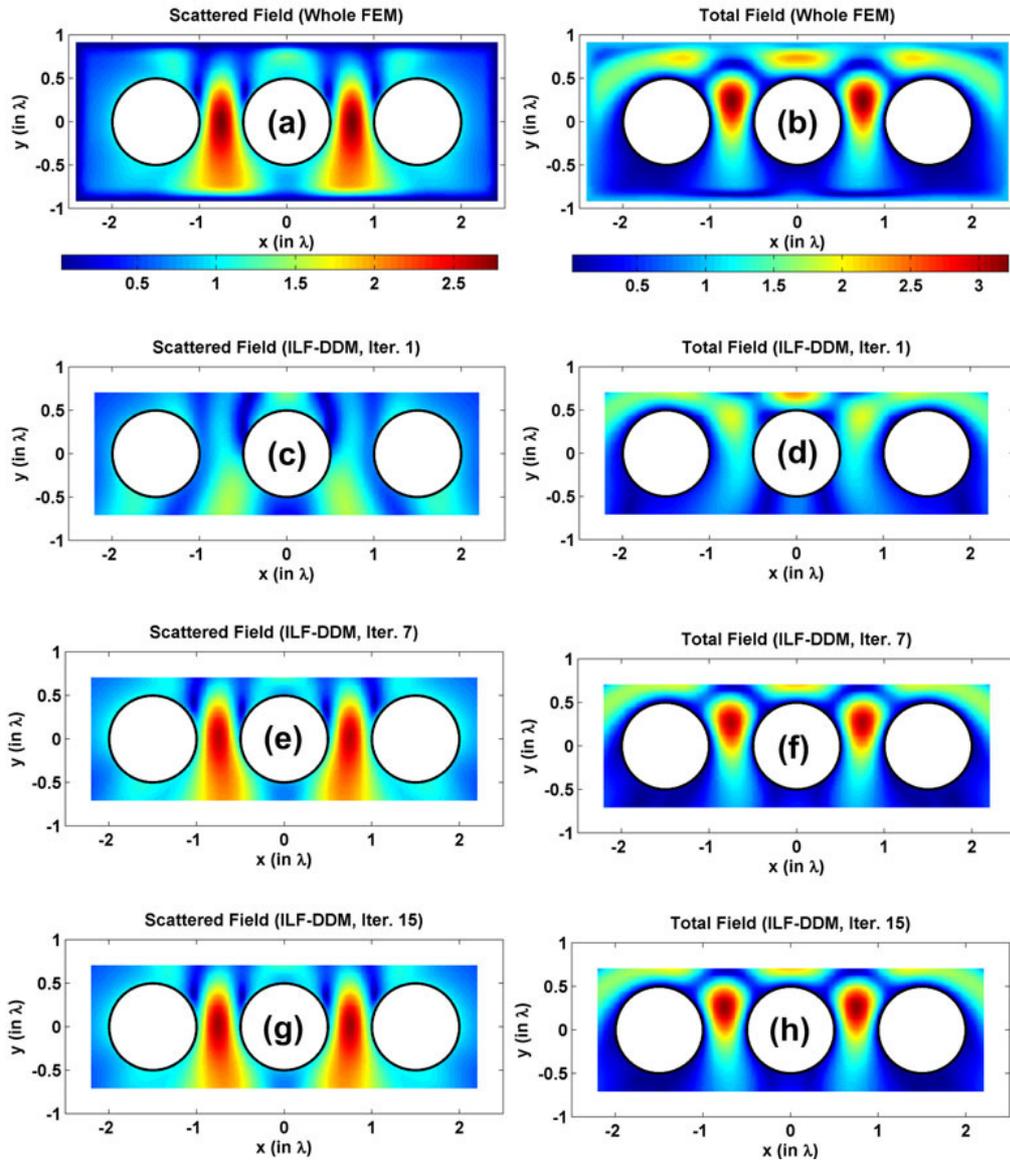


Figure 8.40. Field contours in example 9 (magnitude): (a-b) Scattered and total fields of the whole FEM, (c-d) Scattered and total fields in the 1st iteration, (e-f) Scattered and total fields in the 7th iteration, (g-h) Scattered and total fields in the 15th iteration.

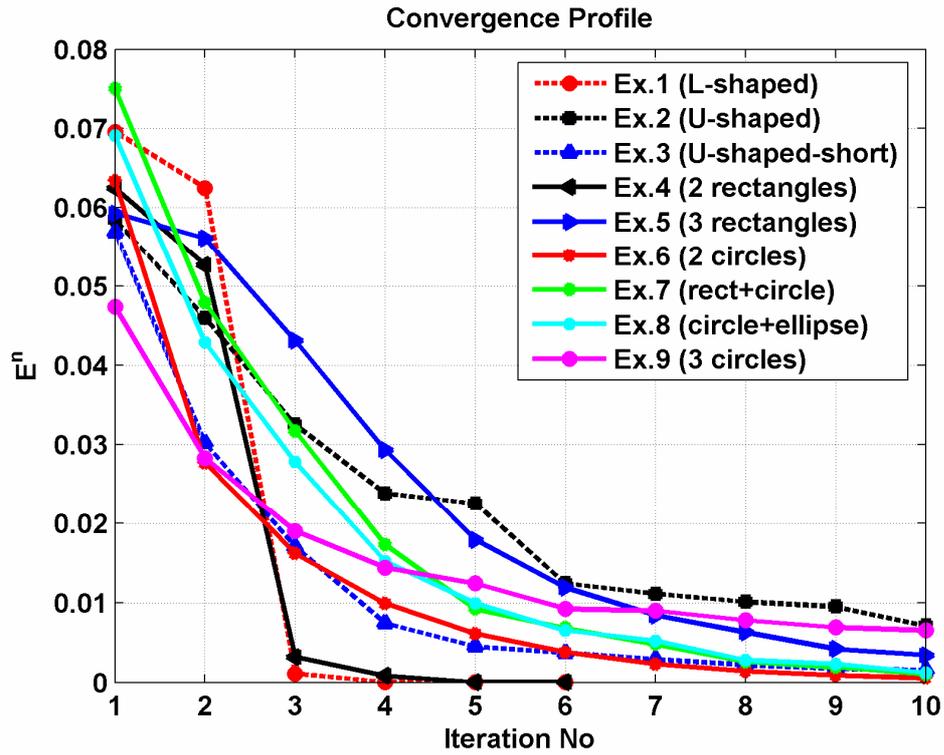


Figure 8.41. Convergence profiles of all examples.

8.2. 3D Simulations

In this section, we test the ILF-DDM algorithm in 3D electromagnetic scattering problems. In all examples, the locally-conformal PML method is implemented using the parameters $\alpha=5k$ and $m=3$, and the approximate PML thickness is chosen as $\lambda/5$. The wavenumber k is set to 20π (i.e., the wavelength λ is 0.1 meter). The element size is approximately set to $\lambda/10$. The first three examples involve single ‘connected’ scatterers, but the other six examples involve multiple scatterers.

The 1st example is the problem of scattering from a dihedral (see Figure 8.42(a)). We assume that the incident plane wave is $\vec{E}^{inc} = \hat{a}_y \exp[jk(x \sin \theta^{inc} + z \cos \theta^{inc})]$ where $\theta^{inc} = 45^\circ$. The original problem is decomposed into two subproblems, as shown in Figure 5.15. The algorithm converges at the 3rd iteration, and we plot the E^n values versus the number of iterations (i.e., convergence profile) in Figure 8.42(b). In addition, we plot the bistatic RCS profiles for the first four iterations in Figure 8.43 by comparing them with the whole domain FEM solution. In each plot, we also plot the solution of the 3rd iteration with the whole domain solution as an inset in order to better visualize the final result. A similar approach is followed in the remaining examples to present the results.

The 2nd example is the problem of scattering from a ‘larger’ dihedral (see Figure 8.44(a)). We assume the same incident plane wave in the 1st example. The algorithm converges at the 3rd iteration, and we plot the convergence profile in Figure 8.44(b). We also plot the bistatic RCS profiles in Figure 8.45.

The 3rd example deals with the scattering from a trihedral (see Figure 8.46(a)). We assume that the incident plane wave is $\vec{E}^{inc} = \hat{a}_y \exp[jkx]$. The original problem is decomposed into three subproblems, as shown in Figure 5.16. The algorithm converges at the 4th iteration, and we plot the convergence profile in Figure 8.46(b). We also plot the bistatic RCS profiles in Figure 8.47.

The 4th example is a scattering from two ‘thin’ rectangular prisms oriented as in Figure 8.48(a). We assume that the incident plane wave is

$$\vec{E}^{inc} = [\hat{a}_x \cos \theta^{inc} - \hat{a}_z \sin \theta^{inc}] \exp[jk(x \sin \theta^{inc} + z \cos \theta^{inc})] \text{ where } \theta^{inc} = 45^\circ.$$

The original problem is decomposed into two subproblems, similar to Figure 5.17. The algorithm converges at the 3rd iteration, and we plot the convergence profile in Figure 8.48(b). We also plot the bistatic RCS profiles in Figure 8.49.

The 5th example is a scattering from two spheres oriented as in Figure 8.50(a). We assume that the incident plane wave is $\vec{E}^{inc} = \hat{a}_x \exp[jkz]$. The algorithm converges at the 3rd iteration, and we plot the convergence profile in Figure 8.50(b). We also plot the bistatic RCS profiles in Figure 8.51.

The 6th example considers a prolate spheroid and a sphere (see Figure 8.52(a)). We assume that the incident plane wave is $\vec{E}^{inc} = \hat{a}_y \exp[jk(x \sin \theta^{inc} + z \cos \theta^{inc})]$ where $\theta^{inc} = 45^\circ$. The algorithm converges at the 4th iteration, and we plot the convergence profile in Figure 8.52(b). We also plot the bistatic RCS profiles in Figure 8.53.

The 7th problem is a scattering from a sphere in front of a plate, as illustrated in Figure 8.54(a). We assume that the incident plane wave is $\vec{E}^{inc} = \hat{a}_y \exp[jkx]$. The algorithm converges at the 2nd iteration, and we plot the convergence profile in Figure 8.54(b). We also plot the bistatic RCS profiles in Figure 8.55.

The 8th example involves three spheres (see Figure 8.56(a)). We assume that the incident plane wave is $\vec{E}^{inc} = \hat{a}_x \exp[jkz]$. The algorithm converges at the 4th iteration, and we plot the convergence profile in Figure 8.56(b). We also plot the bistatic RCS profiles in Figure 8.57.

The 9th example includes three spheroids and a sphere oriented as in Figure 8.58(a). We assume that the incident plane wave is $\vec{E}^{inc} = \hat{a}_x \exp[jkz]$. The algorithm converges at the 6th iteration, and we plot the convergence profile in Figure 8.58(b). We also plot the bistatic RCS profiles in Figure 8.59. Finally, we plot the convergence profiles of all examples in Figure 8.60 for the purpose of comparison. We present some conclusions about the numerical results in Section 8.3.

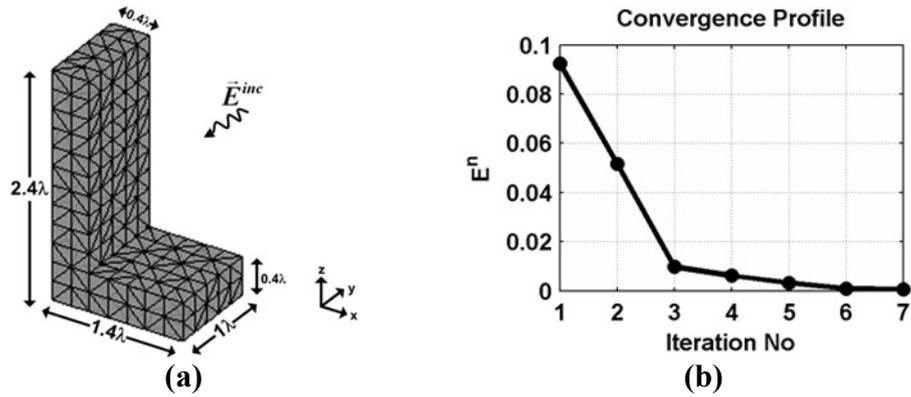


Figure 8.42. Example 1: (a) Original problem, (b) Convergence profile.

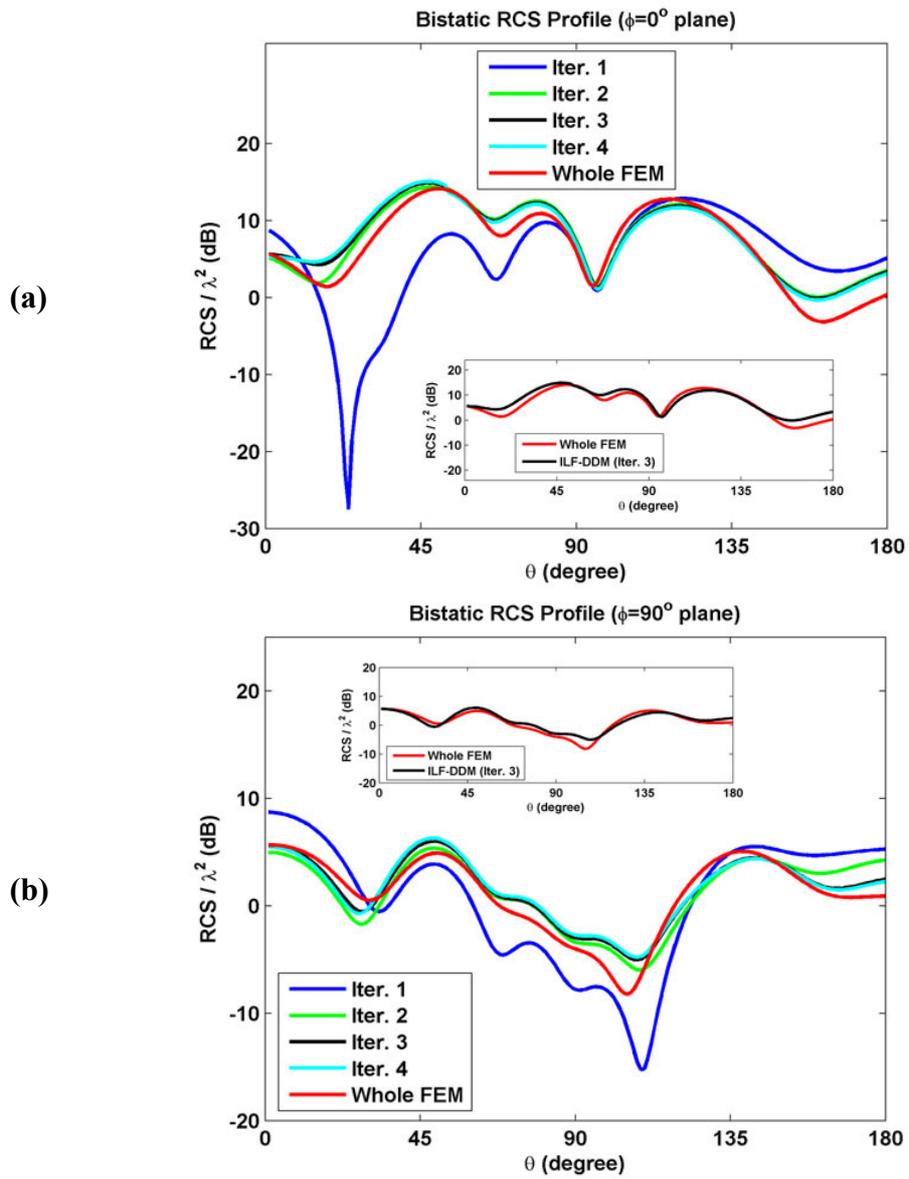


Figure 8.43. Bistatic RCS profile of example 1: (a) $\phi=0^\circ$ plane, (b) $\phi=90^\circ$ plane.

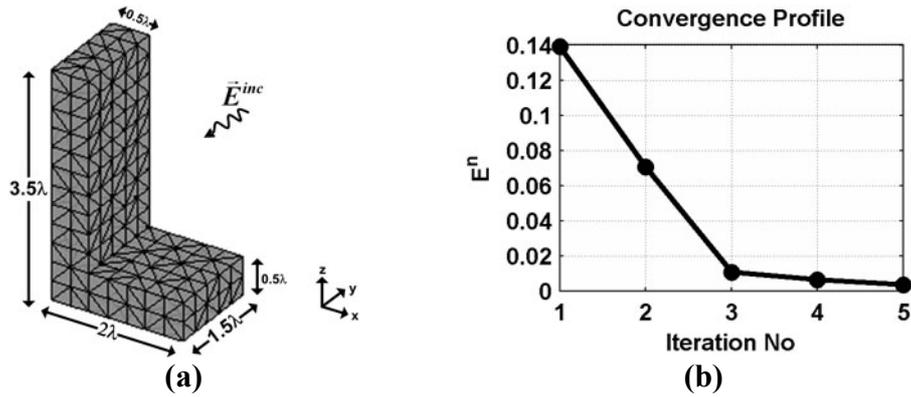


Figure 8.44. Example 2: (a) Original problem, (b) Convergence profile.

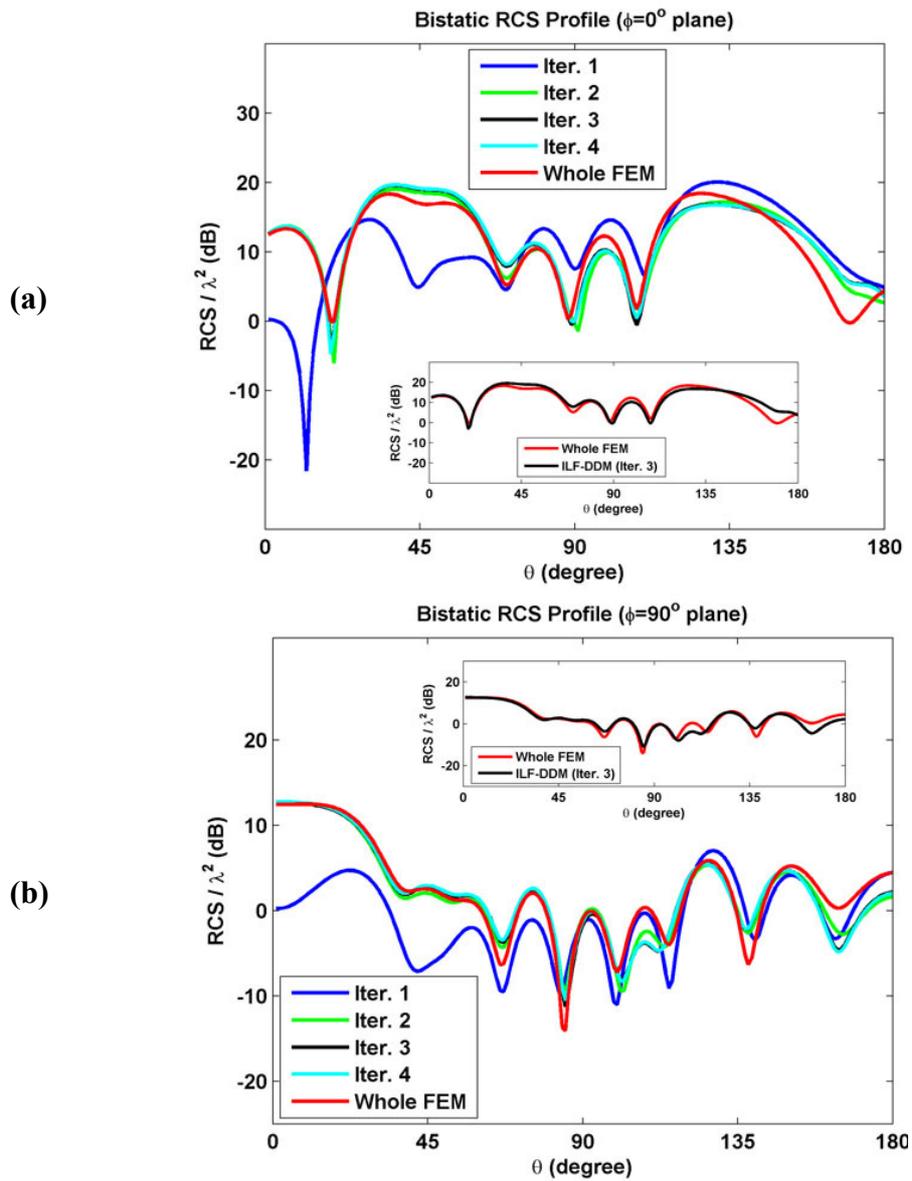


Figure 8.45. Bistatic RCS profile of example 2: (a) $\phi=0^\circ$ plane, (b) $\phi=90^\circ$ plane.

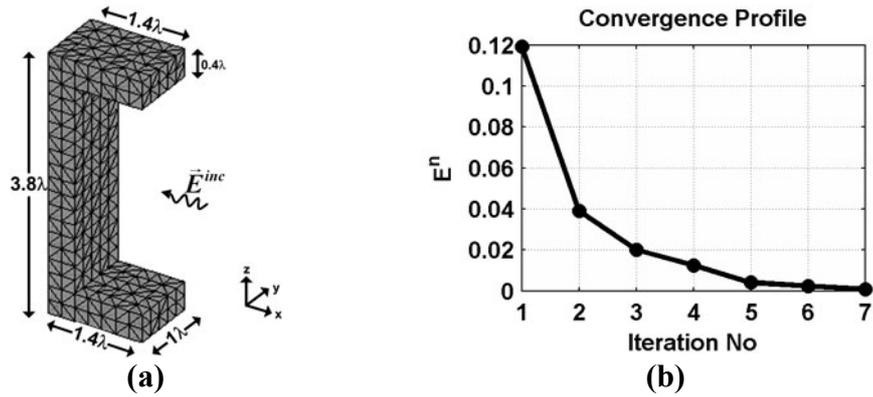


Figure 8.46. Example 3: (a) Original problem, (b) Convergence profile.

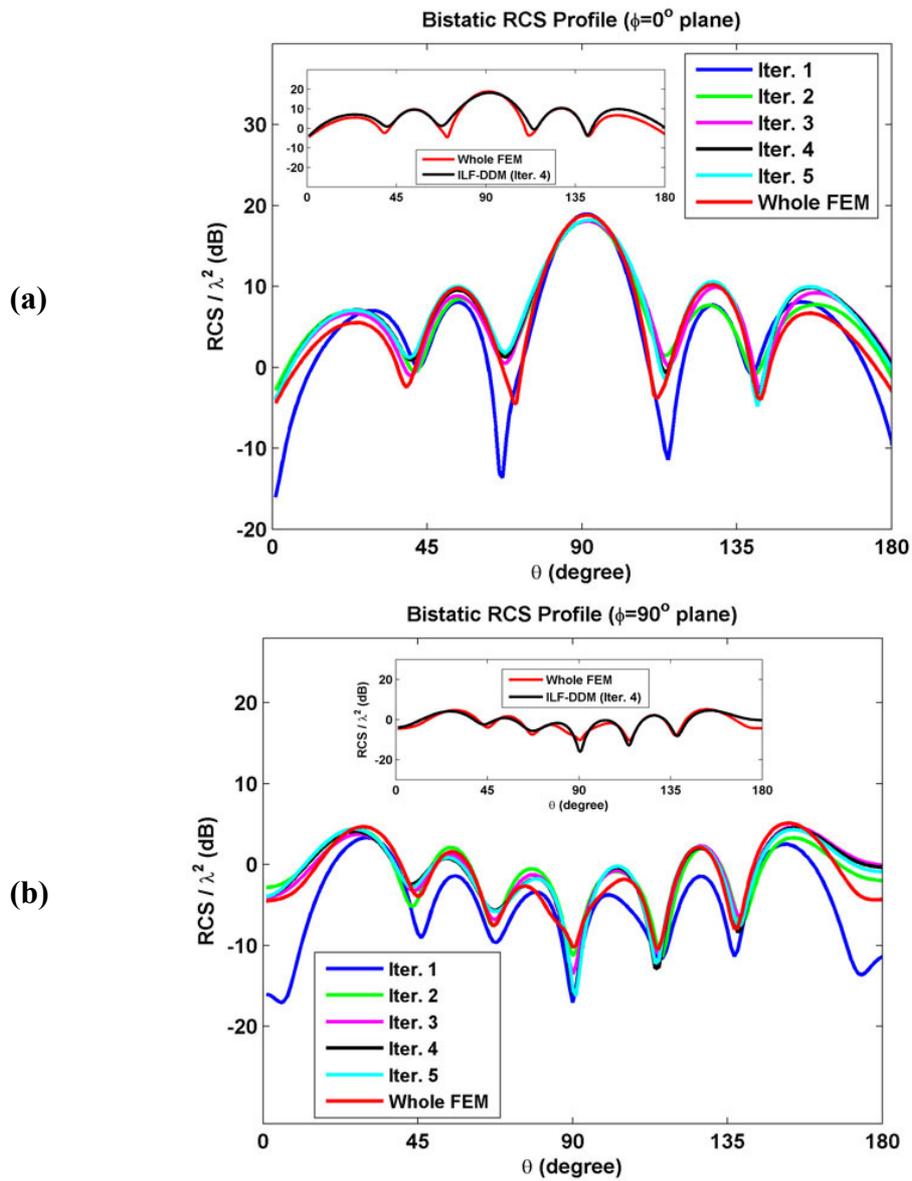


Figure 8.47. Bistatic RCS profile of example 3: (a) $\phi=0^\circ$ plane, (b) $\phi=90^\circ$ plane.

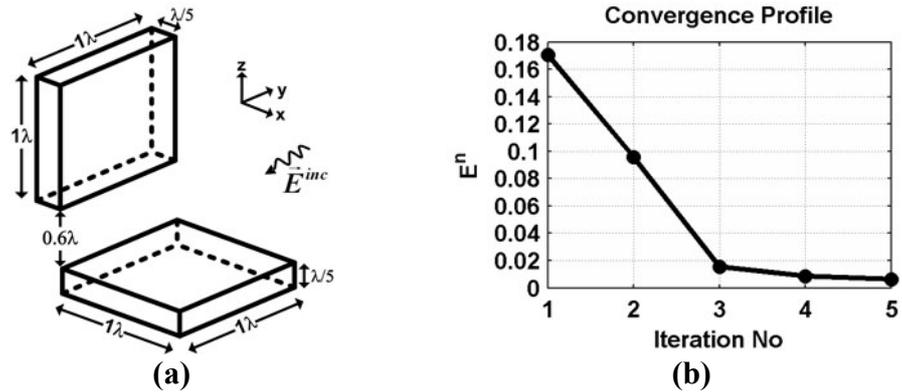


Figure 8.48. Example 4: (a) Original problem, (b) Convergence profile.

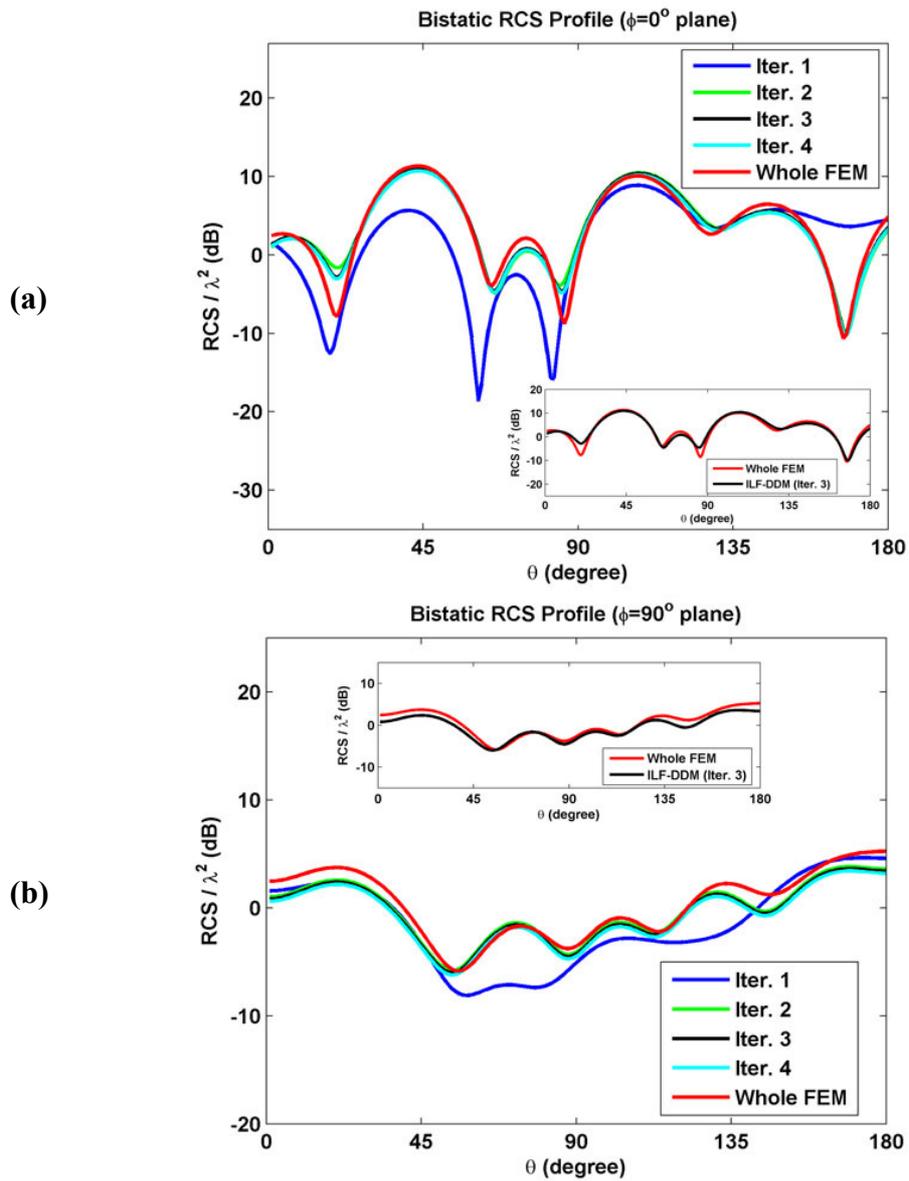


Figure 8.49. Bistatic RCS profile of example 4: (a) $\phi=0^\circ$ plane, (b) $\phi=90^\circ$ plane.

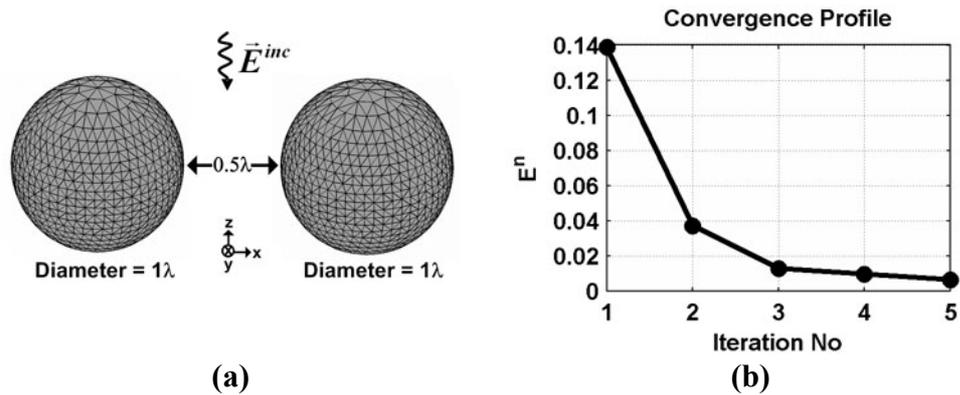


Figure 8.50. Example 5: (a) Original problem, (b) Convergence profile.

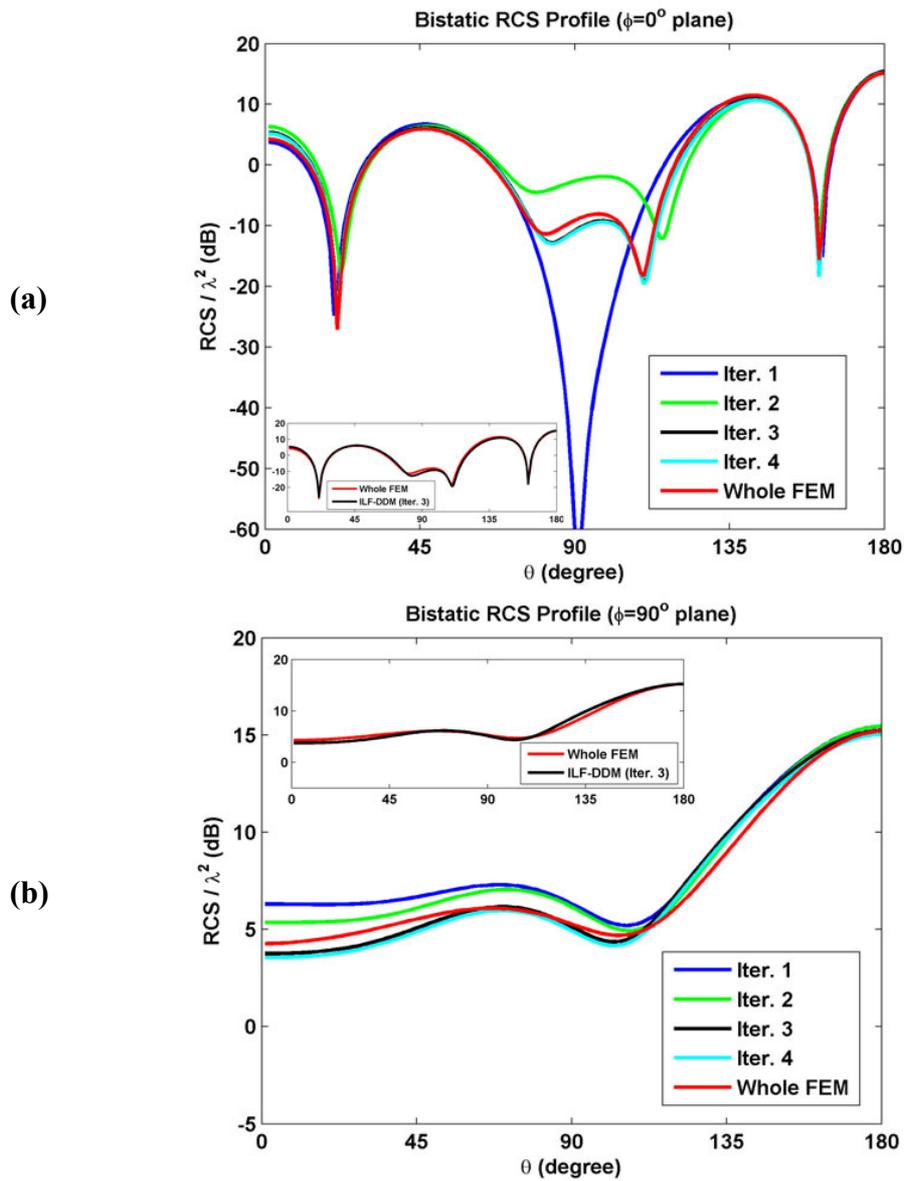


Figure 8.51. Bistatic RCS profile of example 5: (a) $\phi=0^\circ$ plane, (b) $\phi=90^\circ$ plane.

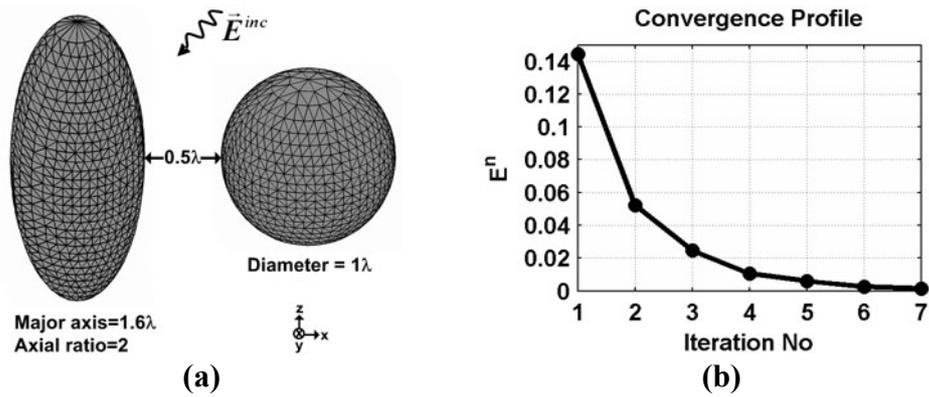


Figure 8.52. Example 6: (a) Original problem, (b) Convergence profile.

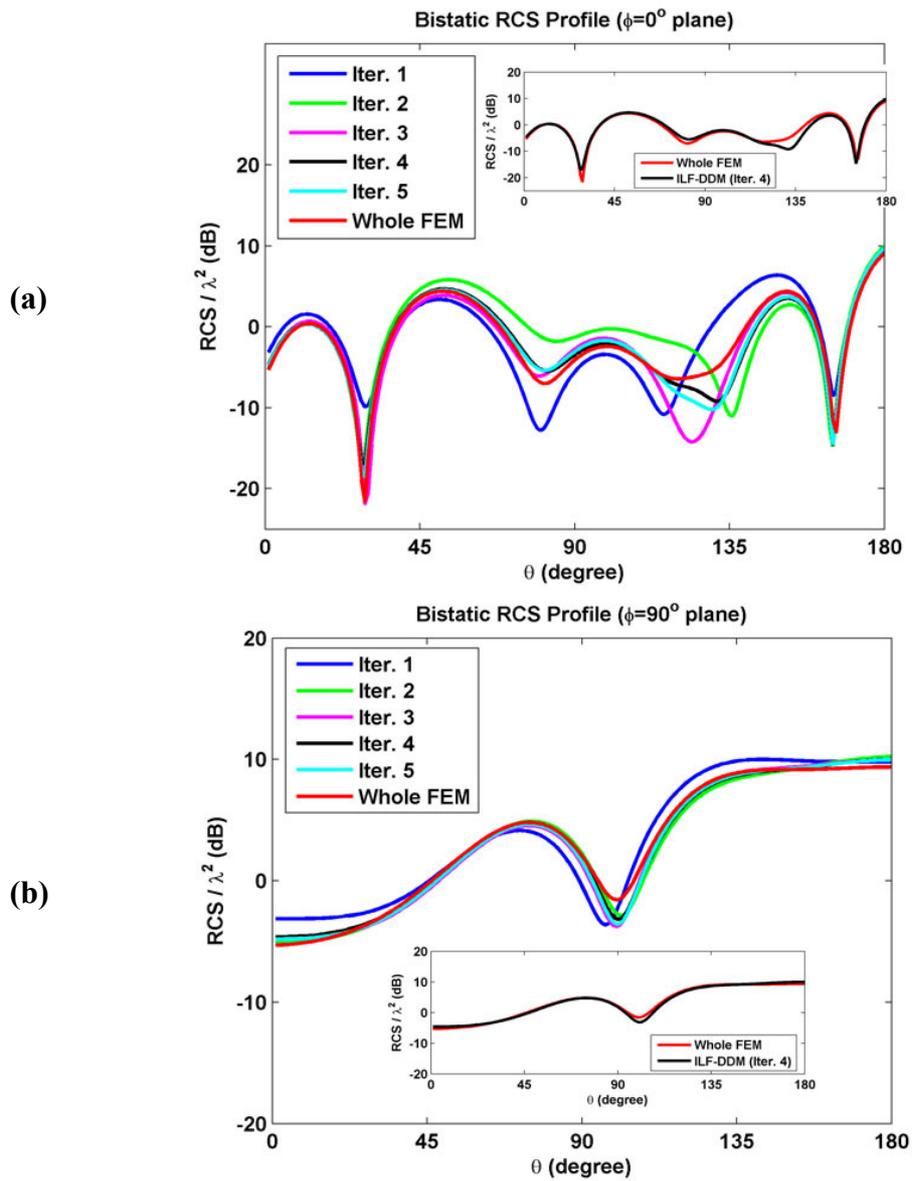


Figure 8.53. Bistatic RCS profile of example 6: (a) $\phi=0^\circ$ plane, (b) $\phi=90^\circ$ plane.

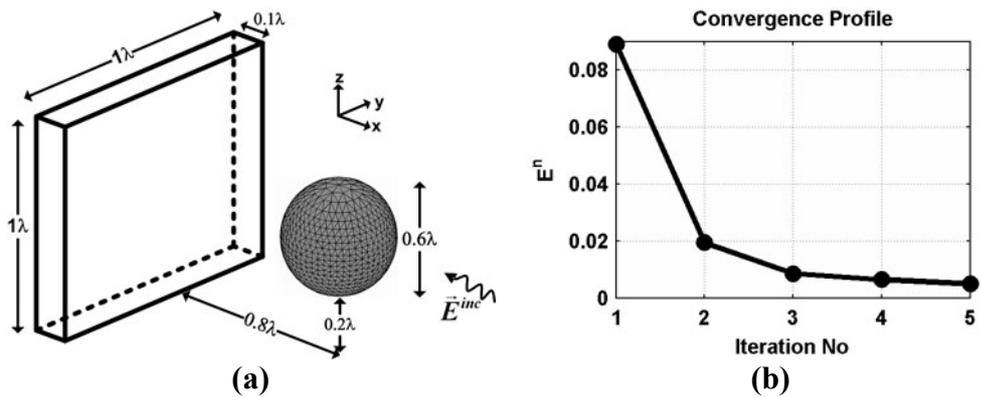


Figure 8.54. Example 7: (a) Original problem, (b) Convergence profile.

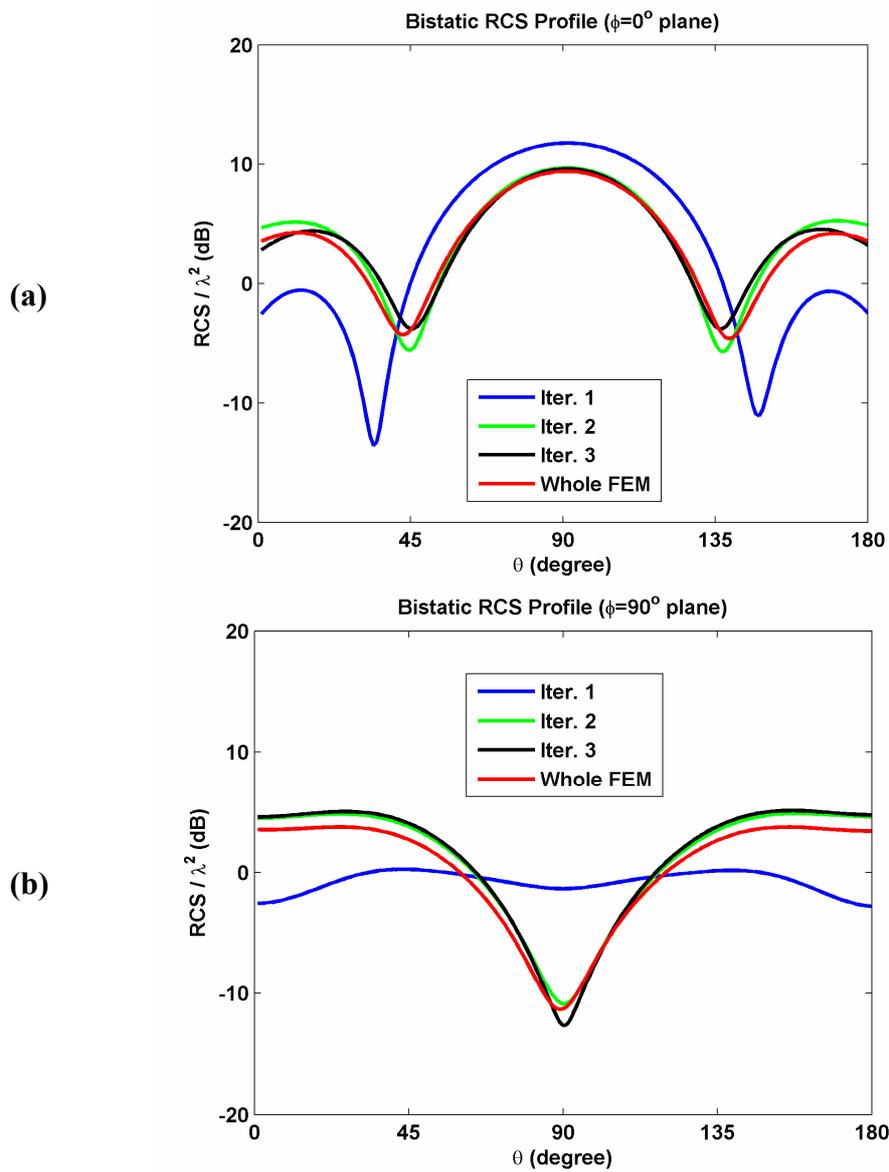


Figure 8.55. Bistatic RCS profile of example 7: (a) $\phi=0^\circ$ plane, (b) $\phi=90^\circ$ plane.

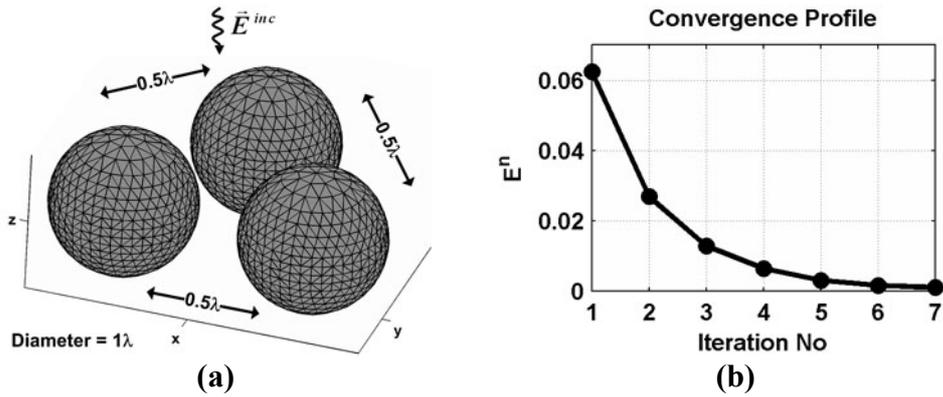


Figure 8.56. Example 8: (a) Original problem, (b) Convergence profile.

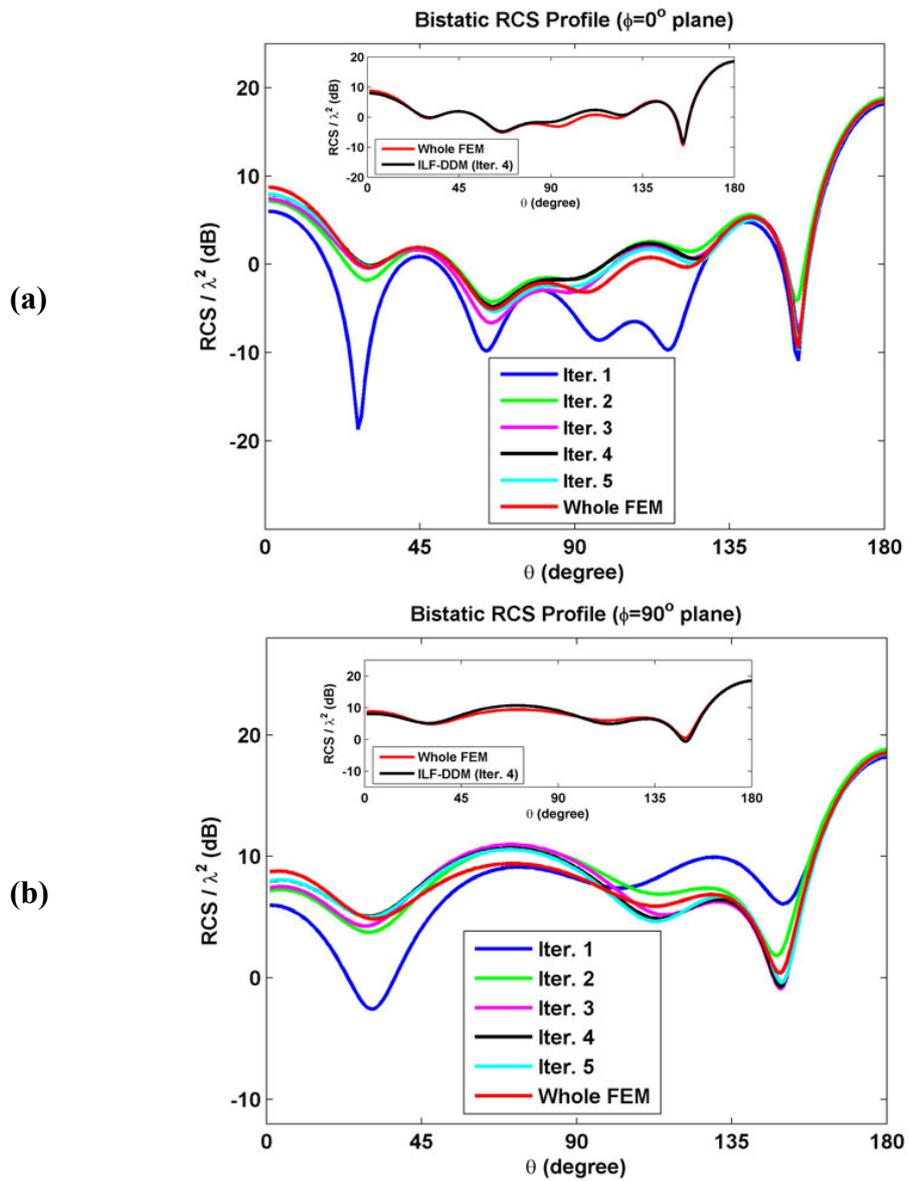


Figure 8.57. Bistatic RCS profile of example 8: (a) $\phi=0^\circ$ plane, (b) $\phi=90^\circ$ plane.

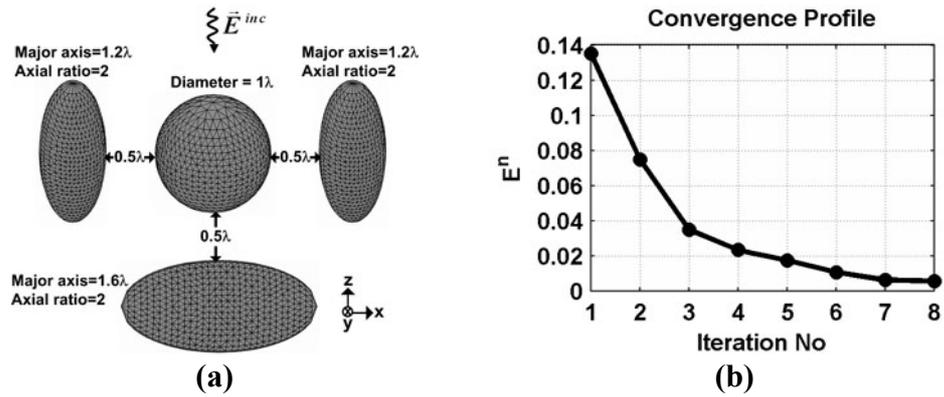


Figure 8.58. Example 9: (a) Original problem, (b) Convergence profile.

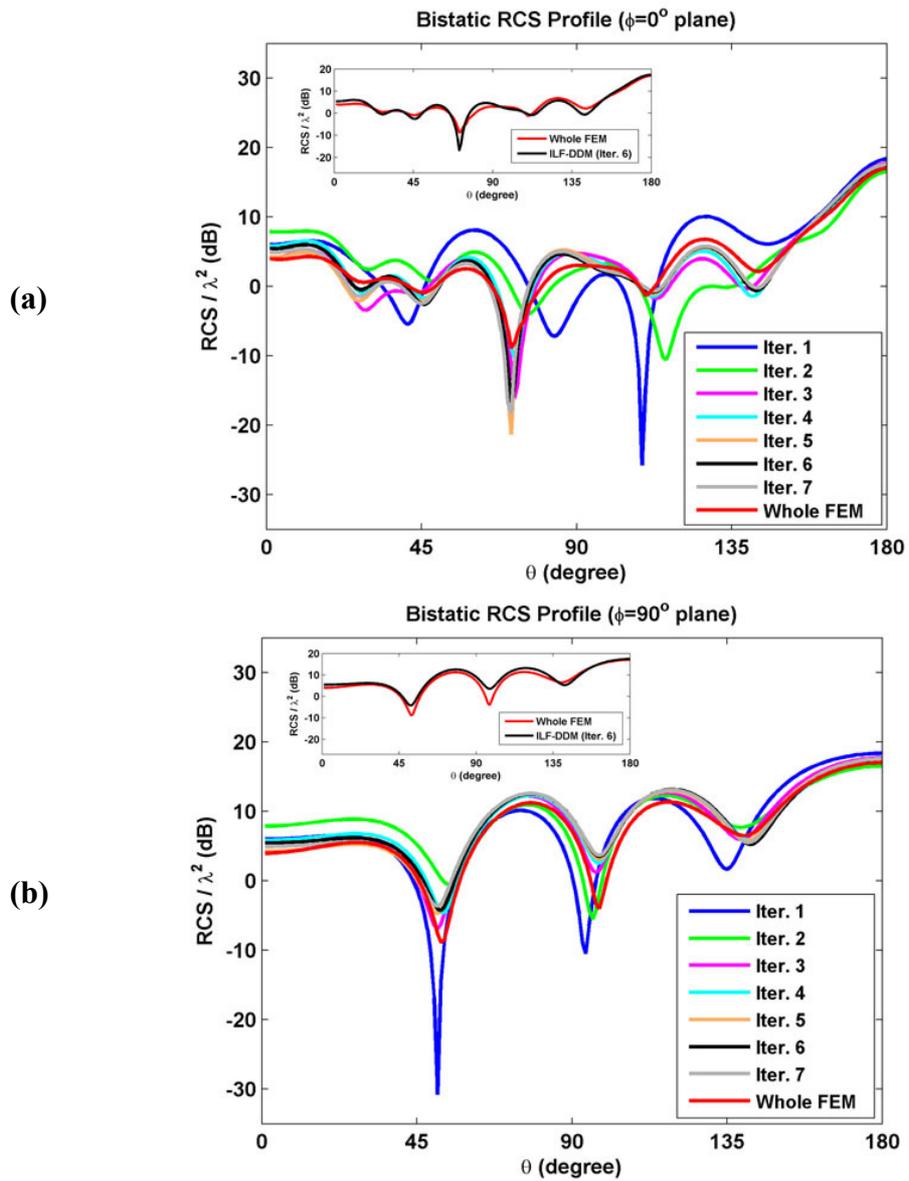


Figure 8.59. Bistatic RCS profile of example 9: (a) $\phi=0^\circ$ plane, (b) $\phi=90^\circ$ plane.

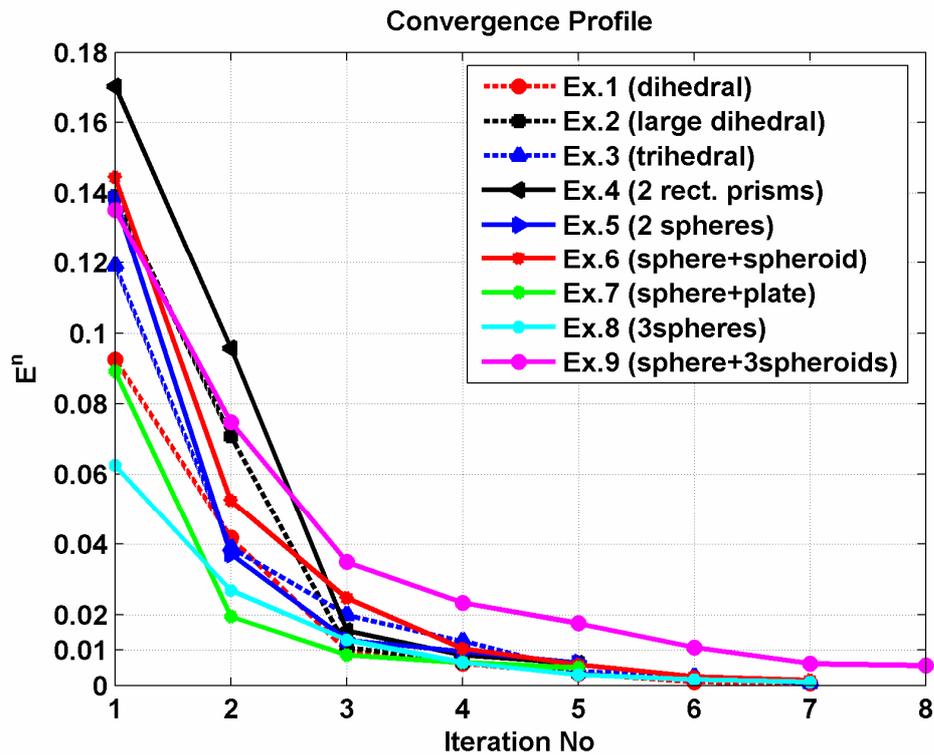


Figure 8.60. Convergence profiles of all examples.

8.3. Observations

The numerical experiments in this chapter lead us to the following important observations:

- (i) The convergence rate of the algorithm directly depends on the geometry of the original problem, as well as the frequency, plane-wave illumination etc. That is, if there are strong field interactions (such as multiple-order reflections and diffractions) between the subproblems, then the convergence rate decreases. Thus, the field which leaps in each iteration may be regarded as the field interaction between the corresponding subproblems. For instance, since the field interaction for

the 3D example in Figure 8.58(a) is the largest among all other 3D examples, its convergence rate is the slowest (see Figure 8.60).

- (ii) It is obvious that the FEM solution for a fine mesh is more reliable than the one obtained by using a coarse mesh. Therefore, although the final RCS results in 3D simulations are close to the whole-domain FEM results, the discrepancies between them may be improved further by decreasing the element size, but sacrificing from the memory requirement due to the larger number of unknowns. In 2D simulations, the final RCS results are very reliable due to the very fine mesh.
- (iii) As mentioned in Section 5.2.1, the Green's function singularity must be handled if the scatterer boundaries of the subproblems are very close to each other. When we implement the expression in (5.17) numerically in 3D simulations, we have observed that the results for the single scatterer (such as dihedral and trihedral) are not as successful as those for the multiple distinct scatterers which are not located close to each other. Similarly, as the distinct scatterers become closer, the results start to deteriorate. In addition, we have observed that the radius of the discarded volume should be at least approximately $\lambda/2$ to get reliable results in these examples. Thus, we have concluded that the differences between the results may be due to the imperfect handling of the Green's function singularity. In any case, it may be asserted that the algorithm provides a good approximation to the original problem.
- (iv) The ILF-DDM algorithm provides a considerable decrease in the memory storage requirements and the computation time. It usually converges in a few number of iterations even in problems having multiple field interactions. In addition, the subproblems may be solved in parallel in each iteration, thus the computation time may be further

decreased using different computers. As an example, if we consider the last 3D example (see Figure 8.58(a)), the number of unknowns in the whole problem is 219,439. For the 1st, 2nd, 3rd and 4th subproblems, the number of unknowns are 27214, 29217, 19886 and 19886, respectively. Thus, the ILF-DDM provides an approximately 56% reduction in the total memory requirements if it is implemented in a single computer. Furthermore, we have solved the matrix system of the whole domain problem iteratively in approximately 50,000 iterations and in 30 hours (up to 10^{-5} convergence tolerance) using the quasi-minimal residual (QMR) method with Gauss-Seidel pre-conditioner using an ordinary PC (Pentium4 with 1GB RAM). However, we have solved the matrix system of each subproblem using the direct solver (i.e., SuperLU [104]) in approximately 15 minutes at most. Therefore, the solution of a single subproblem in six iterations requires approximately 90 minutes. In other words, since we have four subproblems, the total computation time of the algorithm is 6 hours using a single PC and 1.5 hours using four PCs. This means that the ILF-DDM provides an approximately 80% reduction in the computation time using a single PC, and 95% reduction using four PCs.

CHAPTER 9

CONCLUSIONS

The major concern of the research in this thesis is to develop Domain Decomposition Methods (DDMs) for the finite element solution of electromagnetic radiation/scattering problems. A DDM algorithm decomposes a large problem into several coupled subproblems which can be solved independently. Thus, the most attractive advantage of the DDM algorithm is to reduce considerably the memory storage requirements and computation time. Within the framework of this research, two DDM algorithms have been proposed: Forward-Backward DDM (FB-DDM) and Iterative Leap-field DDM (ILF-DDM).

The most distinguished feature of these DDM algorithms is the implementation of the PML concept along the boundaries of the subdomains, as well as the boundary of the unbounded domain. Therefore, an efficient and flexible PML algorithm is required in order to realize the DDM algorithms properly. In this context, two PML algorithms have been proposed: Locally-conformal PML and Multi-center PML.

In addition, since a well-designed FEM software is obviously essential in order to implement the above-mentioned algorithms successfully, two FEM softwares have been developed: FEMS^{2D} (employing triangular nodal elements to solve the scalar Helmholtz equation in 2D problems) and FEMS^{3D} (employing tetrahedral edge elements to solve the vector wave equation in 3D problems).

On the basis of the contributions and the results of this research, the following conclusions can be drawn (considering the chronological order of the progress in this research):

1. Mesh generation is perhaps the most crucial phase in the implementation of the FEM. In this research, mesh generation programs have been developed in MATLAB[®] for some 2D objects using triangular elements, and for some 3D objects using tetrahedral elements. The most distinctive characteristic of the mesh programs is the design of the PML and the free-space regions as ‘conformal’ to the surface of the object (if it is a single and convex object) by taking the quality of the mesh into account. Although it seems that a special effort should be performed for the mesh generation of each object, the main philosophy behind the meshing algorithm is almost the same for all types of the object in the mesh library. In addition, some certain program modules (such as finding the boundary elements, converting the nodal connectivity matrix into edge connectivity matrix in 3D, etc.) are exactly the same for all programs. Thus, the contents of the mesh library can be extended in a straightforward manner (excluding very complex objects, such as an airplane).
2. Apart from the mesh generation programs, two softwares performing the FEM analysis in 2D and 3D have been developed in MATLAB[®] and C, respectively. These analysis programs perform the construction and the solution of the matrix equation, and then calculate some postprocessing parameters (such as RCS). This analysis task is realized in 2D more easily using some features and the functions of the MATLAB[®], however, its realization in 3D requires more challenging work since it is implemented in the C language using some additional routines, such as vector algebra and matrix solver routines. On the other hand, the

development of the program in MATLAB[®] has some drawbacks in terms of the memory requirement and the speed of the computation, compared to the C language. Thus, the development of the 3D program in C is essential since the memory and the speed of the computer cause a serious bottleneck in 3D applications especially having large number of unknowns. Finally, since the analysis phases of the softwares are separated from the mesh generation programs, they can be easily utilized in other platforms using commercially available mesh generation programs.

3. In order to analyze the radiation/scattering problems via FEM, two mesh truncation schemes based on the PML concept have been presented in detail. Both the “locally-conformal PML” and the “multi-center PML” methods are easily implementable PML techniques based on locally-defined complex coordinate transformations. Their major feature is the flexibility to design a PML domain which encloses an arbitrarily-shaped convex spatial domain, especially having discontinuities in the curvature. In addition, they can be easily incorporated into any FEM code without altering the formulations because a single process is to replace the real coordinates in the PML region with their complex counterparts calculated by the complex coordinate transformation. The implementation of the locally-conformal PML is more straightforward because the multi-center PML implementation requires the knowledge of some center position(s) prior to the implementation. However, the selection of the center(s) is not a difficult task depending on the geometry of the problem, and a center selection scheme has been determined in this research. The performances of these methods have been investigated in several 2D and 3D simulations involving particularly the scattering problems. These methods have also been tested via the Monte Carlo simulations considering a simple 2D radiation

problem. The results are convincing with regard to the reference results, and prove the efficiency of the PML methods.

4. In the solution of radiation/scattering problems, two DDM algorithms have been introduced with their detailed implementation procedures. Although the FB-DDM and the ILF-DDM are different from each other, the common characteristic of both algorithms is the utilization of the PML methods along the boundaries of the subdomains. Their major advantage is the reduction in the memory requirements and the computation time, since they usually converge in a few number of iterations (or a single iteration in some cases). The FB-DDM is based on the classical Schwarz method with overlapping subdomains. However, the FB-DDM considers the expected field behavior inside the original domain, and thus, is superior to the Schwarz method in terms of the efficiency and rate of convergence. The FB-DDM converges in just a single iteration in some smooth geometries by appropriately defined subdomains and PML regions. Thus, the partitioning scheme in this method is crucial in order to improve the rate of convergence. For this method, a fast mesh partitioning program has been developed in MATLAB[®] (with some functions implemented in C to speed up the program), which performs an automatic partitioning to the whole mesh irrespective of the type of the object. The second method, ILF-DDM, is based on the iterative application of the Huygens' Principle, using the equivalent currents in each iteration. The method can be applied to a single object, as well as multiple objects. The ILF-DDM does not partition the whole problem, but it creates small subproblems along the boundaries of the object(s). In addition, it is well-suited to the parallel implementation of the subproblems in each iteration. The performances of the FB-DDM and ILF-DDM algorithms have been investigated using several 2D and 3D examples involving the scattering problems. Although

they achieve reliable results, the 2D results are more successful than their 3D counterparts due to the ability to handle very fine meshes in 2D. The 3D results, which have been calculated using relatively coarser meshes in a regular PC, may be further improved by decreasing the element size in the mesh using 'super computers' which can handle very large number of unknowns. In addition, the performances of these algorithms may be investigated in some radiation problems involving complicated structures, as a future work.

Finally, we believe that both PML and DDM algorithms developed in this research would serve as efficient solution approaches for more complicated radiation/scattering problems in electromagnetics.

REFERENCES

- [1] M. Salazar-Palma et al., *Iterative and Self-Adaptive Finite-Elements in Electromagnetic Modeling*. Artech House, London, 1998.
- [2] R. F. Harrington, *Field Computation by Moment Methods*. Macmillan, 1968.
- [3] C. A. Brebbia, *The Boundary Element Method for Engineers*. Pentech Press, London, 1978.
- [4] W. F. Ames, *Numerical Methods for Partial Differential Equations*. Academic Press, San Diego, CA, 1969, 1977, 1992.
- [5] K. S. Yee, "Numerical solution of initial boundary value problems involving Maxwell's equations in isotropic media," *IEEE Transactions on Antennas and Propagation*, vol. 14, no. 3, pp. 302-307, 1966.
- [6] O. A. Liskovets, "The Method of Lines," *Review. Differ. Uravneniya*, vol. 1, no. 2, pp. 1662-1678, 1965.
- [7] J. R. Hofer, "The transmission line matrix method – Theory and applications," *IEEE Transactions on Microwave Theory and Techniques*, vol. MTT-33, no. 10, pp. 882-893, 1985.
- [8] S. G. Mikhlin, *Variational Methods in Mathematical Physics*. Pergamon Press, Oxford, NY, 1964.
- [9] J. R. Shewchuk, "What is a good linear finite element? Interpolation, conditioning, anisotropy, and quality measures," *Technical Report*, University of California at Berkeley, 2002.
- [10] D. A. Field, "Qualitative measures for initial meshes," *International Journal for Numer. Methods in Eng.*, vol. 47, no. 4, pp.887–906, 2000.
- [11] I. Babuska, O. C. Zienkiewicz, J. Gago, and E. R. A. Oliveria (eds.), *Accuracy Estimates and Adaptive Refinements in Finite Element Computations*. John Wiley & Sons, London, 1986.

- [12] H. Whitney, *Geometric Integration Theory*. Princeton University Press, 1957.
- [13] J. C. Nedelec, "Mixed finite elements in R^3 ," *Numer. Math.*, vol. 35, no.3, pp. 315-341, 1980.
- [14] J. F. Lee, D. K. Sun and Z. J. Cendes, "Tangential vector finite elements for electromagnetic field computation," *IEEE Transactions on Magnetics*, vol. 27, no. 5, pp. 4032-4035, 1991.
- [15] J. Jin, *The Finite Element Method in Electromagnetics*. John Wiley & Sons, London, 2002.
- [16] J. N. Reddy, *An Introduction to Finite Element Method*. McGraw Hill, 1993.
- [17] Y. Saad, "SPARSKIT: a basic tool kit for sparse matrix computations," Technical Report, 1994.
- [18] B. M. Irons, "A frontal solution program for finite element analysis," *Int. J. Numer. Meth. Eng.*, vol. 2, no.1, pp. 5-32, 1970.
- [19] M.R. Hestenes, and E. Stiefel, "Methods of conjugate gradients for solving linear systems," *J. Res. Natl. Bur. Stand.*, vol. 49, no. 6, pp.409-436, 1952.
- [20] C. Lanczos, "Solution of systems of linear equations by minimized iterations," *J. Res. Natl. Bur. Stand.*, vol. 49, no.1, pp. 33-53, 1952.
- [21] P. Sonneveld, "CGS: a fast solver for nonsymmetric linear systems," *SIAM J. Sci. Stat. Comput.*, vol. 10, no.1, pp. 35-52, 1989.
- [22] H. A. Van der Vorst, "Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems," *SIAM J. Sci. Stat. Comput.*, vol. 13, no.2, pp. 631-644, 1992.
- [23] R. W. Freund, and N. M. Nachtigal, "QMR: A quasi-minimal residual method for non-Hermitian linear systems," *Numer. Math.*, vol. 60, no.1, pp. 315-339, 1991.
- [24] Y. Saad, "GMRES: A generalized minimal residual algorithm for solving non-symmetric linear systems," *SIAM J. Sci. Stat. Comput.*, vol. 7, no.3, pp. 856-869, 1986.

- [25] Y. Saad, "ILUT: A dual threshold incomplete LU factorization," Technical Report UMSI 92/38, University of Minnesota Supercomputer Institute, Minneapolis, MN, 1992.
- [26] A. Hrenikoff, "Solution of problems in elasticity by the framework method," Transactions of the ASME, Journal of Applied Mechanics, vol. 8, pp. 169-175, 1941.
- [27] R.L. Courant, "Variational methods for the solution of problems of equilibrium and vibration," Bulletin of the American Mathematical Society, vol. 49, pp. 1-43, 1943.
- [28] M. J. Turner, R. W. Clough, H. C. Martin, and L. J. Topp, "Stiffness and deflection analysis of complex structures," Journal of Aeronautical Science, vol. 23, no.9, pp. 805-823, 1956.
- [29] J. H. Argyris, and S. Kelsey, *Energy Theorems and Structural Analysis*. Butterworth Scientific Publications, London, 1960.
- [30] R. W. Clough, "The finite element method in plane stress analysis," Journal of Structures Division, ASCE, Proceedings of the 2nd Conference on Electronic Computation, pp. 345-378, Pittsburgh, PA, 1960.
- [31] S. Ahmed, "Finite element method for waveguide problems," Electronics Letters, vol. 4, no.9, pp. 381-389, 1968.
- [32] P. P. Silvester, "Finite element solution of homogeneous waveguide problems," Alta Frequenza, vol. 38, pp. 313-317, 1969.
- [33] P. L. Arlett, A. K. Bahrani, and O. C. Zienkiewicz, "Application of finite elements to the solution of Helmholtz equation," Proc. IEE, vol. 115, pp. 1762-1766, 1968.
- [34] R. L. Higdon, "Absorbing boundary conditions for difference approximations to the multi-dimensional wave equation," Mathematical Computations, vol. 47, pp. 437-459, Oct. 1986.
- [35] A. Bayliss, M. Gunzburger, and E. Turkel, "Boundary conditions for the numerical solution of elliptic equations in exterior domains," SIAM Journal of Applied Mathematics, vol. 42, no. 2, pp. 430-451, 1982.
- [36] B. Stupfel, "Absorbing boundary conditions on arbitrary boundaries for the scalar and vector wave equations," IEEE Transactions on Antennas and Propagation, vol. 42, no. 6, pp. 773-780, 1994.

- [37] R. Gordon, R. Mittra, A. Glisson, and E. Michielssen, "Finite element analysis of electromagnetic scattering by complex bodies using an efficient numerical boundary condition for mesh truncation," *Electron. Lett.*, vol. 29, no. 12, pp. 1102-1103, 1993.
- [38] A. Boag, and R. Mittra, "A numerical absorbing boundary condition for 3D edge-based finite-element analysis of very low-frequency fields," *Microwave and Optical Technology Letters*, vol. 9, no. 1, pp. 22-27, 1995.
- [39] B. Stupfel, and R. Mittra, "A theoretical study of numerical absorbing boundary conditions," *IEEE Transactions on Antennas and Propagation*, vol. 43, no. 5, pp. 478-487, 1995.
- [40] B. Stupfel, and R. Mittra, "Numerical absorbing boundary conditions for the scalar and vector wave equations," *IEEE Transactions on Antennas and Propagation*, vol. 44, no. 7, pp. 1015-1022, 1996.
- [41] P. Bettess, "Infinite Elements," *Int. J. Numer. Methods Eng.*, vol. 11, no. 1, pp. 53-64, 1977.
- [42] M. S. Towers, A. McCowan, and J. A. R. Macnab, "Electromagnetic scattering from an arbitrary, inhomogeneous 2-D object – A finite and infinite element solution," *IEEE Transactions on Antennas and Propagation*, vol. 41, no. 6, pp. 770-777, 1993.
- [43] J.P. Berenger, "A perfectly matched layer for the absorption of electromagnetic waves," *J.Comput. Phys.*, vol.114, no.2, pp.185-200, 1994
- [44] D.S. Katz, E.T. Thiele, and A. Taflove, "Validation and extension to three dimensions of the Berenger PML absorbing boundary condition for FD-TD meshes," *IEEE Microwave Guided Wave Lett.*, vol. 4, no. 8, pp. 268-270, 1994.
- [45] J. C. Weihl, R. Mittra, "Efficient implementation of Berenger's perfectly matched layer (PML) for finite-difference-time-domain mesh truncation," *IEEE Microwave Guided Wave Lett.*, vol. 6, no. 2, pp. 94-96, 1996.
- [46] Z. Wu, and J. Fang, "Numerical implementation and performance of perfectly matched layer boundary condition for waveguiding structures," *IEEE Transactions on Microwave Theory and Techniques*, vol. 43, no. 12, pp. 2676-2683, 1995.
- [47] M. F. Pasik, G. Aguirre, and A. C. Cangellaris, "Application of the PML boundary condition to the analysis of patch antennas," *Electromagnetics*, vol. 16, no. 4, pp. 435-449, 1996.

- [48] Z. S. Sacks, D. M. Kingsland, R. Lee, and J.-F. Lee, "A perfectly matched anisotropic absorber for use as an absorbing boundary condition," *IEEE Trans. on Antennas and Propagation*, vol. 43, no. 12, pp. 1460-1463, 1995.
- [49] M. Kuzuoglu, and R. Mittra, "Investigation of nonplanar perfectly matched absorbers for finite element mesh truncation," *IEEE Transactions on Antennas and Propagation*, vol. 45, no. 3, pp. 474-486, 1997.
- [50] M. Kuzuoglu, and R. Mittra, "Mesh truncation by perfectly matched anisotropic absorbers in the finite element method," *Microwave and Optical Technology Letters*, vol. 12, no. 3, pp. 136-140, 1996.
- [51] S. D. Gedney, "An anisotropic perfectly matched layer absorbing medium for the truncation of FDTD lattices," *IEEE Transactions on Antennas and Propagation*, vol. 44, no. 12, pp. 1630-1639, 1996.
- [52] M. Kuzuoglu, and R. Mittra, "A systematic study of perfectly matched absorbers," (in: D. H. Werner, R. Mittra eds.), *Frontiers in Electromagnetics*. IEEE Press, 2000.
- [53] J. Y. Wu, D. M. Kingsland, J. F. Lee, and R. Lee, "A comparison of anisotropic PML to Berenger's PML and its application to the finite-element method for EM scattering," *IEEE Transactions on Antennas and Propagation*, vol. 45, no. 1, pp. 40-50, 1997.
- [54] U. Pekel, and R. Mittra, "An application of the perfectly matched layer (PML) concept to the finite element method frequency domain analysis of scattering problems," *IEEE Microwave Guided Wave Lett.*, vol. 5, no. 8, pp. 258-260, 1995.
- [55] M. Kuzuoglu, and R. Mittra, "Frequency dependence of the constitutive parameters of causal perfectly matched anisotropic absorbers," *IEEE Microwave Guided Wave Lett.*, vol. 6, no. 12, pp. 447-449, 1996.
- [56] J. Tang, K. D. Paulsen, and S. A. Haider, "Perfectly matched layer mesh terminations for nodal-based finite element methods in electromagnetic scattering," *IEEE Transactions on Antennas and Propagation*, vol. 46, no. 4, pp. 507-516, 1998.
- [57] M. Kuzuoglu, and R. Mittra, "A systematic approach to the derivation of constitutive parameters of a perfectly matched absorber," *IEEE Microwave Guided Wave Lett.*, vol. 8, no. 9, pp. 313-315, 1998.

- [58] S. D. Gedney, "Anisotropic PML absorbing media for the FDTD simulation of fields in lossy and dispersive media," *Electromagnetics*, vol. 16, no. 4, pp. 399-415, 1996.
- [59] A. J. Roden, and S. D. Gedney, "Efficient Implementation of the uniaxial-based PML media in three-dimensional nonorthogonal coordinates with the use of FDTD technique," *Microwave and Optical Technology Letters*, vol. 14, no. 2, pp. 71-75, 1997.
- [60] M. S. Tong, Y. C. Chen, M. Kuzuoglu, and R. Mittra, "A new anisotropic perfectly matched layer medium for mesh truncation in finite difference time domain analysis," *International Journal of Electronics*, vol. 86, no. 9, pp. 1085-1091, 1999.
- [61] W. C. Chew, and W. Weedon, "A 3D perfectly matched medium from modified Maxwell's equations with stretched coordinates," *Microwave and Optical Technology Letters*, vol. 7, no. 13, pp. 599-604, 1994.
- [62] F. L. Teixeira, and W. C. Chew, "Systematic derivation of anisotropic PML absorbing media in cylindrical and spherical coordinates," *IEEE Microwave Guided Wave Lett.*, vol. 7, no. 11, pp. 371-373, 1997.
- [63] F. L. Teixeira, and W. C. Chew, "Analytical derivation of a conformal perfectly matched absorber for electromagnetic waves," *Microwave and Optical Technology Letters*, vol. 17, no. 4, pp. 231-236, 1998.
- [64] H. Schwarz, *Gesammelte Mathematische Abhandlungen*. vol. 2, pp. 133–143, Springer, Berlin, Germany, 1890.
- [65] Z. Zhu, H. Ji, and W. Hong, "An efficient algorithm for the parametric extraction of 3-D interconnect structures in the VLSI circuits: Domain-decomposition method," *IEEE Transactions on Microwave Theory and Techniques*, vol. 45, no. 8, pp. 1179-1184, 1997.
- [66] V.V. Veremey and R. Mittra, "Efficient computation of interconnect capacitances using the domain decomposition approach," *IEEE Transactions on Advanced Packaging*, vol. 22, no. 3, pp. 348-355, 1999.
- [67] B. Despres, P. Joly, and J.E. Roberts, "A domain decomposition method for the harmonic Maxwell equations," in *Iterative Methods in Linear Algebra*. North-Holland, Amsterdam, Holland, 1992.
- [68] F. Collino, G. Delbue, P. Joly, and A. Piacentini, "A new interface condition in the non-overlapping domain decomposition methods for the

Maxwell equations,” *Computer Methods in Applied Mechanics and Engineering*, vol. 148, no. 1, pp. 195-207, 1997.

- [69] B. Stupfel, “A Fast-Domain Decomposition Method for the Solution of Electromagnetic Scattering by Large Objects,” *IEEE Transactions on Antennas and Propagation*, vol. 44, no.10, pp. 1375-1385, 1996.
- [70] B. Stupfel, and M. Mognot, “A domain decomposition method for the vector wave equation,” *IEEE Transactions on Antennas and Propagation*, vol. 48, no. 5, pp.653-660, 2000.
- [71] C. Farhat and F.-X. Raux, “A method of finite element tearing and interconnecting and its parallel solution algorithms,” *International Journal for Numer. Methods in Eng.*, vol. 32, no. 6, pp. 1205-1227, 1991.
- [72] C.T. Wolfe, U. Navsariwala, and S.D. Gedney, “A parallel finite-element tearing and interconnecting algorithm for solution of the vector wave equation with PML absorbing medium,” *IEEE Transactions on Antennas and Propagation*, vol. 48, no. 2, pp. 278- 284, 2000.
- [73] Y. Li, and J.M. Jin, “A vector dual-primal finite element tearing and interconnecting method for solving 3-D large-scale electromagnetic problems,” *IEEE Transactions on Antennas and Propagation*, vol. 54, no. 10, pp. 3000- 3009, 2006.
- [74] M. Gander, C. Japhet, Y. Maday, and F. Nataf, “A new cement to glue nonconforming grids with Robin interface conditions: The finite element case,” *15th International Conference on Domain Decomposition Methods*, Berlin, Germany, 2003.
- [75] S.C. Lee, M.N. Vouvakis, and J.F. Lee, “A non-overlapping domain decomposition method with non-matching grids for modeling large finite antenna arrays,” *Journal of Comput. Physics*, vol. 203, pp. 1-21, 2005.
- [76] M. Carr and J.L. Volakis, “Domain decomposition by iterative field bouncing,” in *Proc. IEEE Antennas and Propagation Society, AP-S Int. Symp. Digest*, vol. 3, pp. 298–301, San Antonio, TX, 2002.
- [77] F. Xu and W. Hong, “Analysis of two dimensions sparse multicylinder scattering problem using DD-FDTD Method,” *IEEE Transactions on Antennas and Propagation*, vol. 52, no. 10, pp. 2612-2617, 2004.
- [78] I. V. Lindell, *Methods for Electromagnetic Field Analysis*. Oxford University Press, 1992.

- [79] B.F. Smith, P.E. Bjorstad, and W.D. Gropp, *Domain Decomposition, Parallel Multilevel Methods for Elliptic Partial Differential Equations*. Cambridge University Press, 1996.
- [80] E. Giladi, and J.B. Keller, "A hybrid numerical asymptotic method for scattering problems," *Journal of Computational Physics*, vol. 174, pp. 226–247, 2001.
- [81] R. Bancroft, *Understanding Electromagnetic Scattering using the Moment Method*. Artech House, 1996.
- [82] C.A. Balanis, *Advanced Engineering Electromagnetics*. John Wiley & Sons, 1989.
- [83] B.R. Mahafza, *Radar Systems Analysis and Design Using MATLAB*. Chapman & Hall / CRC, 2005.
- [84] R.F. Harrington, *Time-Harmonic Electromagnetic Fields*. McGraw-Hill, 1961.
- [85] A. Ishimaru, *Electromagnetic Wave Propagation, Radiation and Scattering*. Pentice Hall, 1991.
- [86] J.L. Volakis, A. Chatterjee, and L.C. Kempel, *Finite Element Method for Electromagnetics*. IEEE Press, 1998.
- [87] T.K. Sarkar, E. Arvas, and S. Ponnappalli, "Electromagnetic scattering from dielectric bodies," *IEEE Transactions on Antennas and Propagation*, vol. 37, no: 5, pp. 673-676, 1989.
- [88] M. Marrone, A.M.F. Frasson, and H.E. Hernandez-Figueroa, "A novel numerical approach for electromagnetic scattering: The cell method," *Proceedings of the 2002 IEEE AP-S and URSI, San Antonio, Texas, USA*, pp. 160-163, vol. 1, June 16-21, 2002.
- [89] B.P. Sinha, and R.H. MacPhie, "Electromagnetic Scattering from prolate spheroids for axial incidence," *IEEE Transactions on Antennas and Propagation*, vol. 23, no. 5, pp. 676–679, 1975.
- [90] L.N. Medgyesi-Mitschang, and D.S. Wang, "Hybrid solutions for scattering from perfectly conducting bodies of revolution," *IEEE Transactions on Antennas and Propagation*, vol. AP-31, no: 4, pp. 570-583, 1983.

- [91] A. Chatterjee, M. Jin, and L. Volakis, "Edge-based finite elements and vector ABC's applied to 3-D scattering," *IEEE Transactions on Antennas and Propagation*, vol. 41, no: 2, pp. 221-226, 1993.
- [92] J.B. Keller, "Backscattering from a finite cone – Comparison of theory and experiment," *IEEE Transactions on Antennas and Propagation*, vol. 9, no: 4, pp. 411-412, 1961.
- [93] W.E. Blore, "The radar cross section of ogives, double-backed cones, double-rounded cones, and cone spheres," *IEEE Transactions on Antennas and Propagation*, vol. 12, no: 5, pp. 582-590, 1964.
- [94] J.M. Rius, E. Ubeda, and J. Parron, "On the testing of the magnetic field integral equation with RWG basis functions in method of moments," *IEEE Trans. on Antennas and Propagation*, vol. 49, no: 11, pp. 1550-1553, 2001.
- [95] R.G. Kouyoumjian, "The calculation of the echo areas of perfectly conducting objects by the variational method," Ph.D. dissertation, The Ohio State University, 1953.
- [96] S.S. Bindiganavale, J.L. Volakis, and H. Anastassiou, "Scattering from planar structures containing small features using the adaptive integral method (AIM)," *IEEE Transactions on Antennas and Propagation*, vol. 46, no: 12, pp. 1867-1878, 1998.
- [97] M.G. Cote, M.B. Woodworth, and A.D. Yaghjian, "Scattering from the perfectly conducting cube," *IEEE Transactions on Antennas and Propagation*, vol. 36, no: 9, pp. 1321-1329, 1988.
- [98] J.V. Bladel, "Some remarks on Green's dyadic for infinite space," *IEEE Trans. on Antennas and Propagation*, vol. 9, no. 11, pp. 563-566, 1961.
- [99] Mathworks Company, <http://www.mathworks.com>, Last accessed date: March 2007.
- [100] C. B. Barber, D. P. Dobkin, and H. T. Huhdanpaa, "The quickhull algorithm for convex hulls," *ACM Transactions on Mathematical Software*, vol. 22, no. 4, pp. 469-483, 1996.
- [101] E. R. Bank, "PLTMG: A Software Package for Solving Elliptic Partial Differential Equations, User's Guide 6.0," Society for Industrial and Applied Mathematics, Philadelphia, PA, 1990.

- [102] J. J. Guzman, C. Schiff, "A preliminary study for a tetrahedron formation: Quality factors and visualization," AIAA/AAS Astroynamics Specialist Conference and Exhibit, 5-8 August 2002, Monterey, California.
- [103] SPARSKIT tool-kit, <http://www-users.cs.umn.edu/~saad/software/SPARSKIT/sparskit.html>, Last accessed date: March 2007.
- [104] SuperLU software package, <http://crd.lbl.gov/~xiaoye/SuperLU/#superlu>, Last accessed date: March 2007.
- [105] O. Ozgun, and M. Kuzuoglu, "Locally-conformal perfectly matched layer implementation for finite element mesh truncation," *Microwave and Optical Technology Letters*, vol. 48, no. 9, pp. 1836-1839, Sept. 2006.
- [106] O. Ozgun, and M. Kuzuoglu, "Non-Maxwellian locally-conformal PML absorbers for finite element mesh truncation," *IEEE Transactions on Antennas and Propagation*, vol. 55, no. 3, pp. 931-937, March 2007.
- [107] O. Ozgun, and M. Kuzuoglu, "Near-field performance analysis of locally-conformal perfectly matched absorbers via Monte Carlo simulations," *Journal of Computational Physics*, (under review).
- [108] O. Ozgun, and M. Kuzuoglu, "Locally-conformal and multi-center perfectly matched layer implementations for finite element mesh truncation," 2006 IEEE AP-S International Symposium and USNC/URSI National Radio Science Meeting, July 9-14, 2006, Albuquerque, New Mexico, USA.
- [109] O. Ozgun, and M. Kuzuoglu, "Multi-center perfectly matched layer implementation for finite element mesh truncation," *Microwave and Optical Technology Letters*, vol. 49, no. 4, pp. 827-832, April 2007.
- [110] O. Ozgun, and M. Kuzuoglu, "Forward-Backward Domain Decomposition Method for Finite Element Solution of Boundary Value Problems," *Microwave and Optical Technology Letters*, (under review).
- [111] O. Ozgun, and M. Kuzuoglu, "Iterative Leap-Field Domain Decomposition Method for Finite Element Solution of Electromagnetic Boundary Value Problems," *IEEE Transactions on Antennas and Propagation*, (under review).
- [112] O. Ozgun, and M. Kuzuoglu, "Finite Element Analysis of Electromagnetic Scattering Problems via Iterative Leap-Field Domain Decomposition Method," *Microwave and Optical Technology Letters*, (under review).

APPENDIX A: 2D FEM SOFTWARE (FEMS^{2D})

FEMS^{2D} has been developed in the MATLAB[®] language [99] to solve the Helmholtz equation in 2D electromagnetic radiation/scattering problems using the FEM with nodal triangular elements. The FEMS^{2D} consists of two independent but successive programs:

1. Mesh Generation (*meshgen_*.m*)
2. FEM Analysis (main body, matrix solution and post-processing phases are incorporated into a single program) (*fem.m*)

A.1. Mesh Generation

Mesh Generation programs (*meshgen_*.m*) have been developed in the MATLAB[®] language to generate mesh parameters for some 2D objects (i.e., infinitely-long cylinders with arbitrary cross-sections) using nodal triangular elements. Some functions of the programs are implemented in the external *.exe* files which are programmed in the C language, in order to accelerate the execution of the programs. The mesh library includes the infinitely-long cylinders with the following cross-sections:

1. Circular (*meshgen_circle.m*)
2. Half-circular (*meshgen_hcircle.m*)
3. Elliptical (*meshgen_elp.m*)
4. Rectangular (or square) (*meshgen_rect.m*)
5. Polygonal (convex) (*meshgen_convex.m*)
6. Polygonal (convex or concave) (*meshgen_concave.m*)
7. Ogive (*meshgen_ogive.m*)

8. Triangular+Halfcircular (*meshgen_trihc.m*)
9. Defined by $f(x, y) \leq 0$ in rectangular domain (*meshgen_fxy.m*)

The 2D mesh is created using the MATLAB[®] function *delaunay*^{*}, and then some mesh parameters (such as boundary nodes, boundary elements, etc.) are calculated using the mesh. The *Delaunay triangulation* is a set of simplices with the property that, for each simplex, the unique circle circumscribed about the simplex contains no data points. In a 2D space, a simplex is a triangle. In order to construct a proper mesh, the quality of each triangular element is computed according to some criteria, which is given by the following formula [101]

$$Q_{2D} = \frac{4\sqrt{3}a}{h_1^2 + h_2^2 + h_3^2} \quad (\text{A.1})$$

where a is the area, and h_1 , h_2 , and h_3 are the side lengths of the triangle. This parameter has a minimum value of zero and equals to 1 for a regular triangle (viz., equilateral triangle, $h_1 = h_2 = h_3$).

Some examples from the mesh library are shown in the figures in between Figure A.1 and Figure A.8. In addition, an example for the mesh of multiple scatterers (used in the ILF-DDM simulations) is shown in Figure A.9. For the DDM algorithms, some examples from the mesh partitioning schemes are shown in Figure A.10 and Figure A.11.

Input Parameters: The input parameters must be entered manually at the beginning of each Mesh Generation program file. In addition to the input parameters which define specifically the geometry of each object (such as radius of circle, vertices of polygon, etc.), there are also some common parameters appearing in all program files (see Table A.1).

* The function *delaunay* in MATLAB[®] uses actually an external program, called as *Qhull* [100].

Output Parameters: At the end of the program, the output parameters are saved in a MAT-file (*meshgen.mat*), whose major contents are listed in Table A.2.

Table A.1. Mesh Generation: List of common input parameters.

Parameter Name	Definition / Unit
scat_type	1 or 2 1: object is PEC 2: object is dielectric
delh	Approximate edge size per element (meter)
dpml	Distance between object and PML boundaries (meter)
dhuyg	Distance between object and Huygens surface boundaries (meter)
dvoi	Distance between PML and volume-of-interest (VOI) boundaries (meter)

Table A.2. Mesh Generation: List of major output parameters.

Parameter Name	Definition
Nnode	Number of nodes
Nelement	Number of elements
conn	Node-connectivity matrix (Nelement \times 3)
xcoord	x-coordinates of nodes (Nnode \times 1)
ycoord	y-coordinates of nodes (Nnode \times 1)
scatb_node	Array containing the node IDs on the object boundary
pmlb_node	Array containing the node IDs on the PML boundary
voib_node	Array containing the node IDs on the volume-of-interest (VOI) boundary
scatb_elm	Array containing the element IDs on the object boundary (if object is PEC)
huygb_elm	Array containing the element IDs on the Huygens box boundary (if object is dielectric)
inside_elm	Array containing the element IDs inside the object (if object is dielectric)
pml_node	Array containing the node IDs inside the PML region

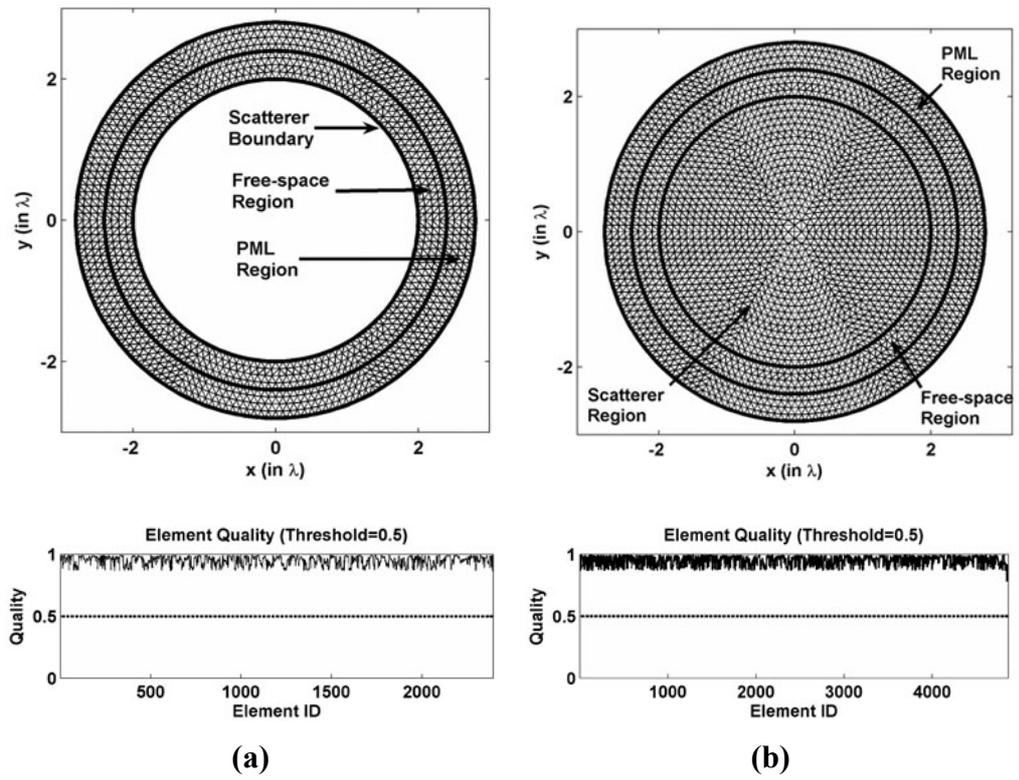


Figure A.1. Mesh Library, Circular: (a) PEC, (b) Dielectric.

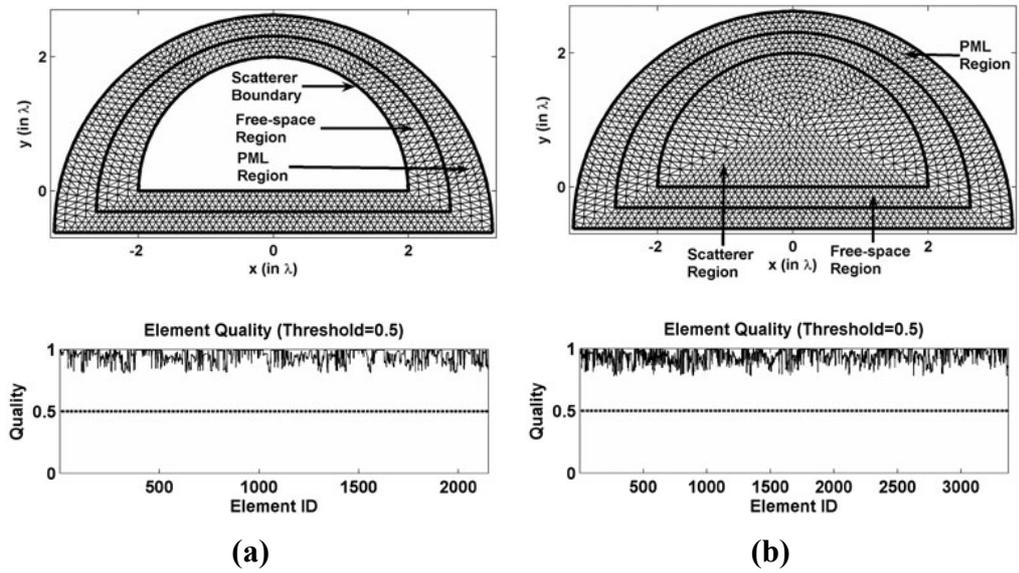


Figure A.2. Mesh Library, Half-circular: (a) PEC, (b) Dielectric.

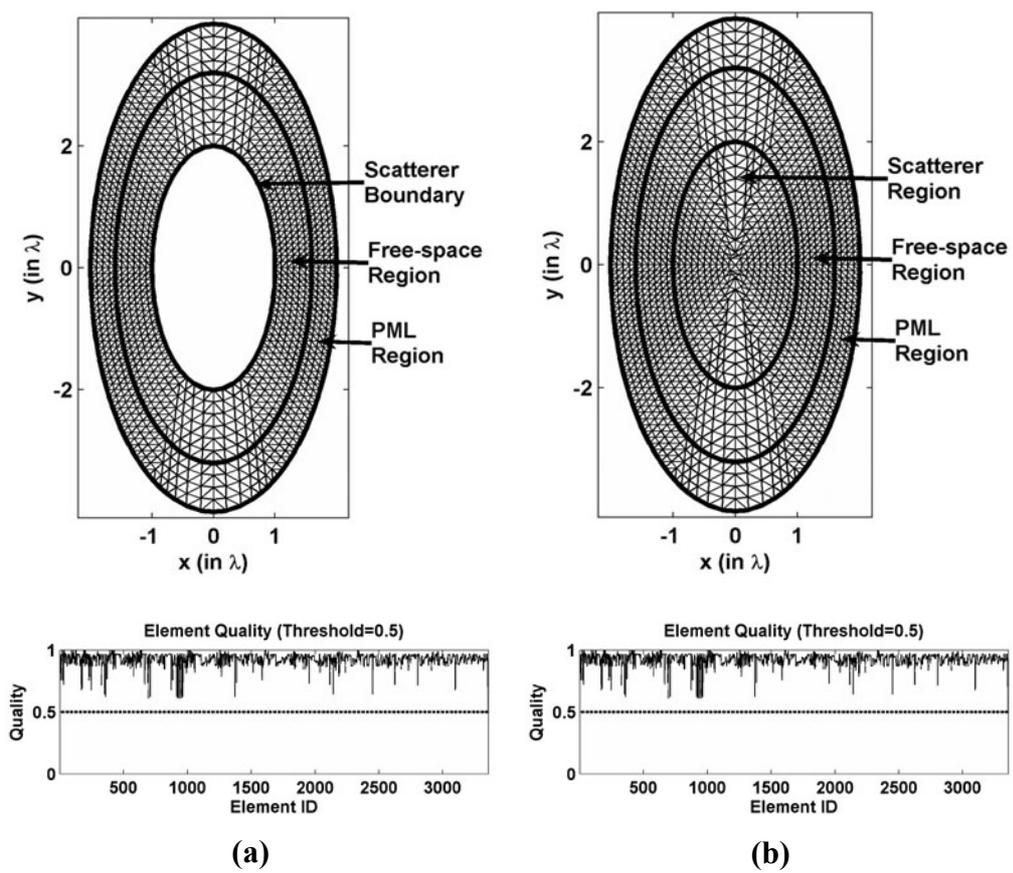


Figure A.3. Mesh Library, Elliptical: (a) PEC, (b) Dielectric.

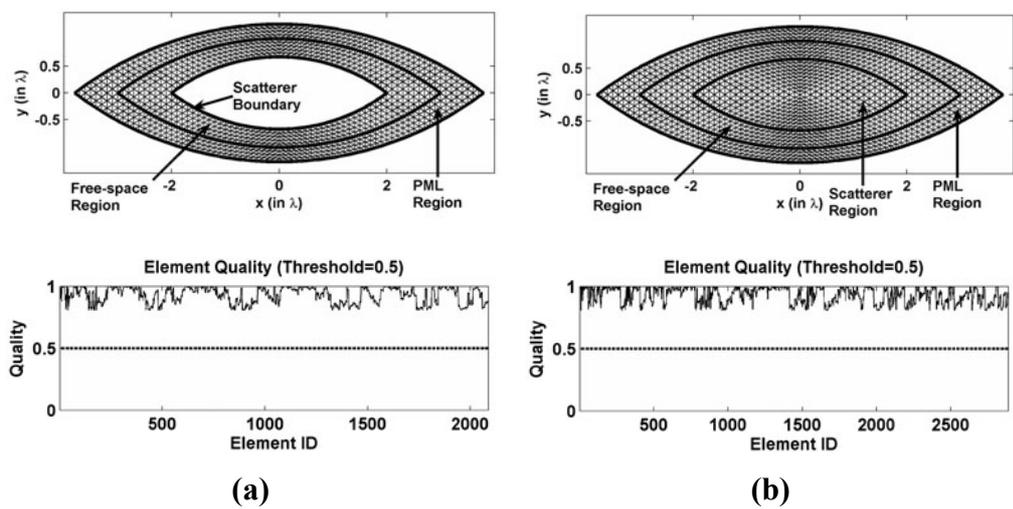


Figure A.4. Mesh Library, Ogive: (a) PEC, (b) Dielectric.

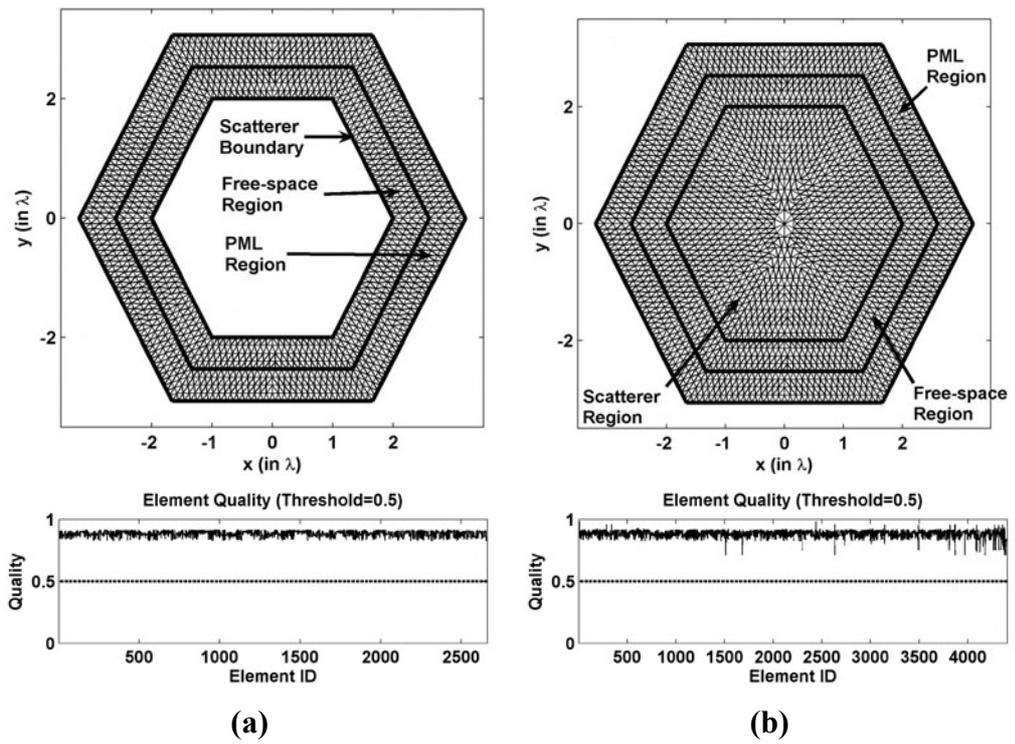


Figure A.5. Mesh Library, Convex Polygonal: (a) PEC, (b) Dielectric.

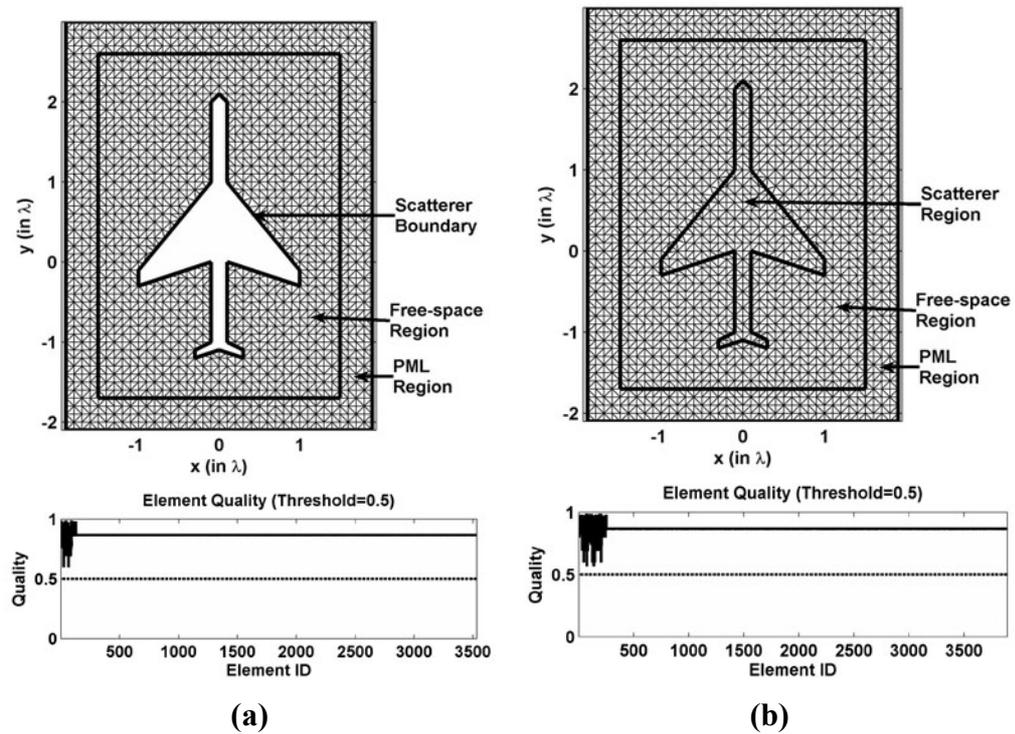


Figure A.6. Mesh Library, Concave Polygonal: (a) PEC, (b) Dielectric.

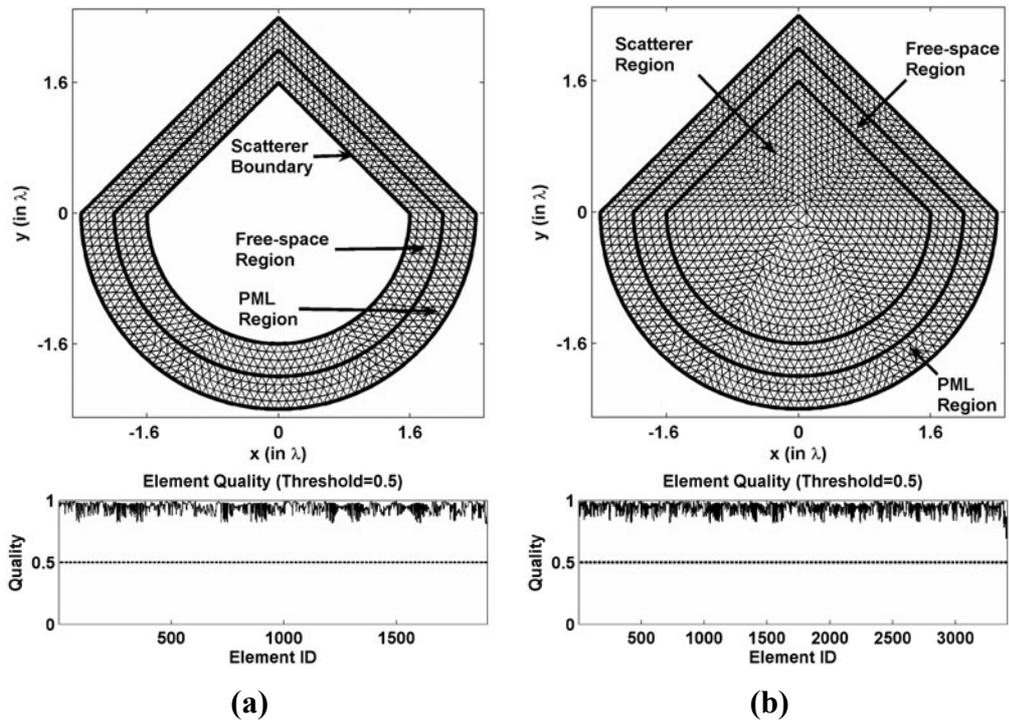


Figure A.7. Mesh Library, Triangular+Halfcircular: (a) PEC, (b) Dielectric.

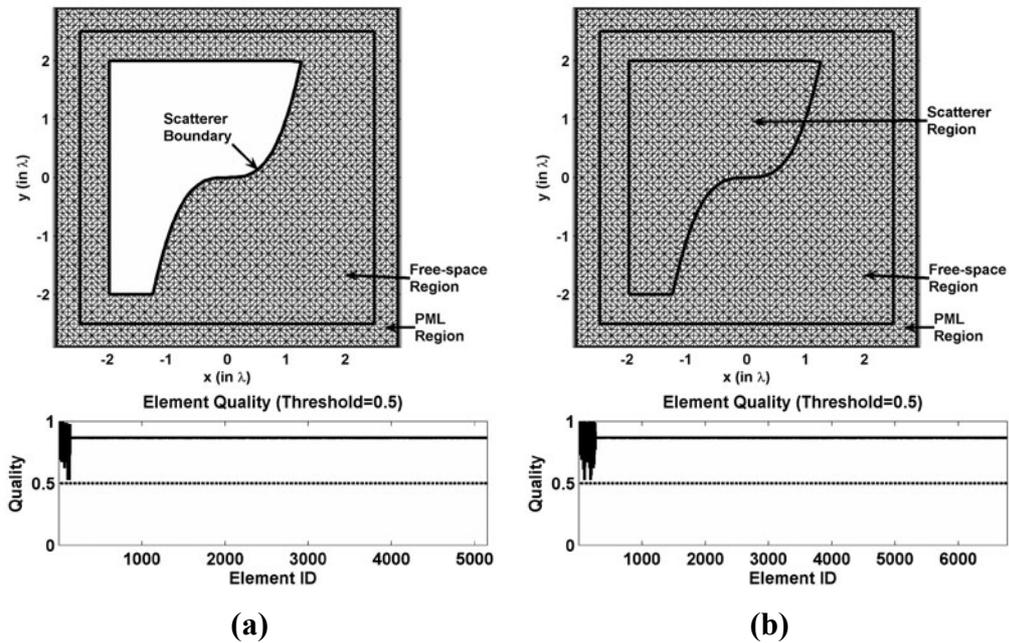


Figure A.8. Mesh Library, Defined by $\sin(x^3) - 0.01y \leq 0$ in square domain: (a) PEC, (b) Dielectric.

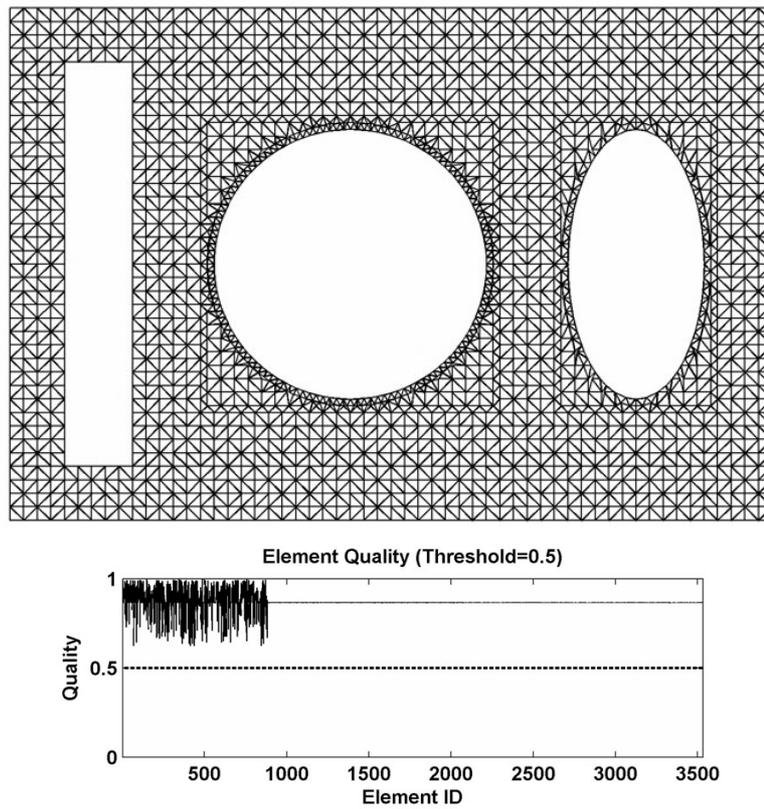


Figure A.9. Mesh Library, Multiple PEC scatterers.

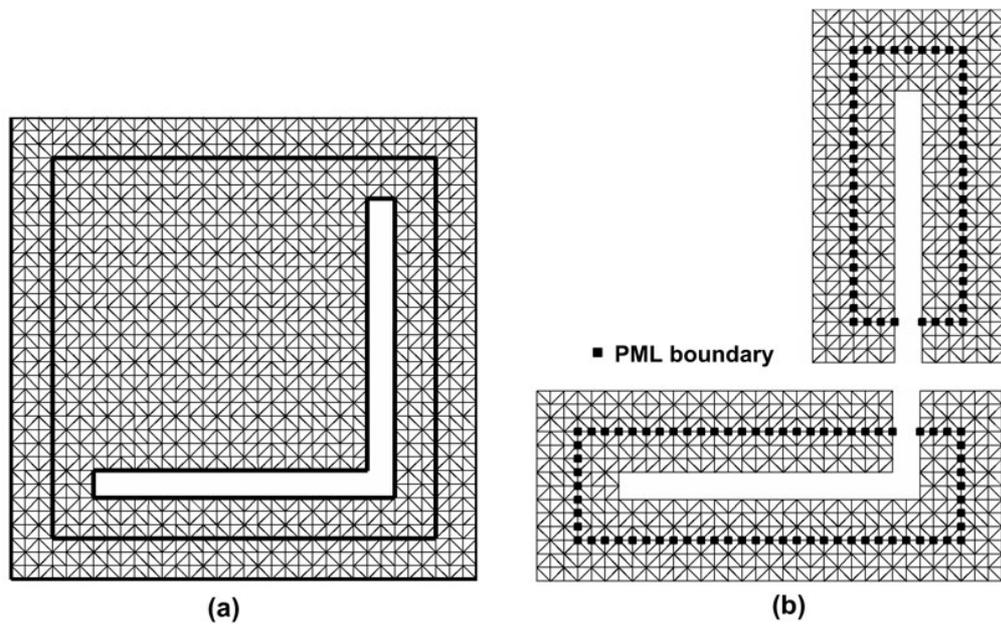


Figure A.10. Mesh partitioning of L-shaped cylinder in the ILF-DDM: (a) Original (whole) domain, (b) 2 subdomains.

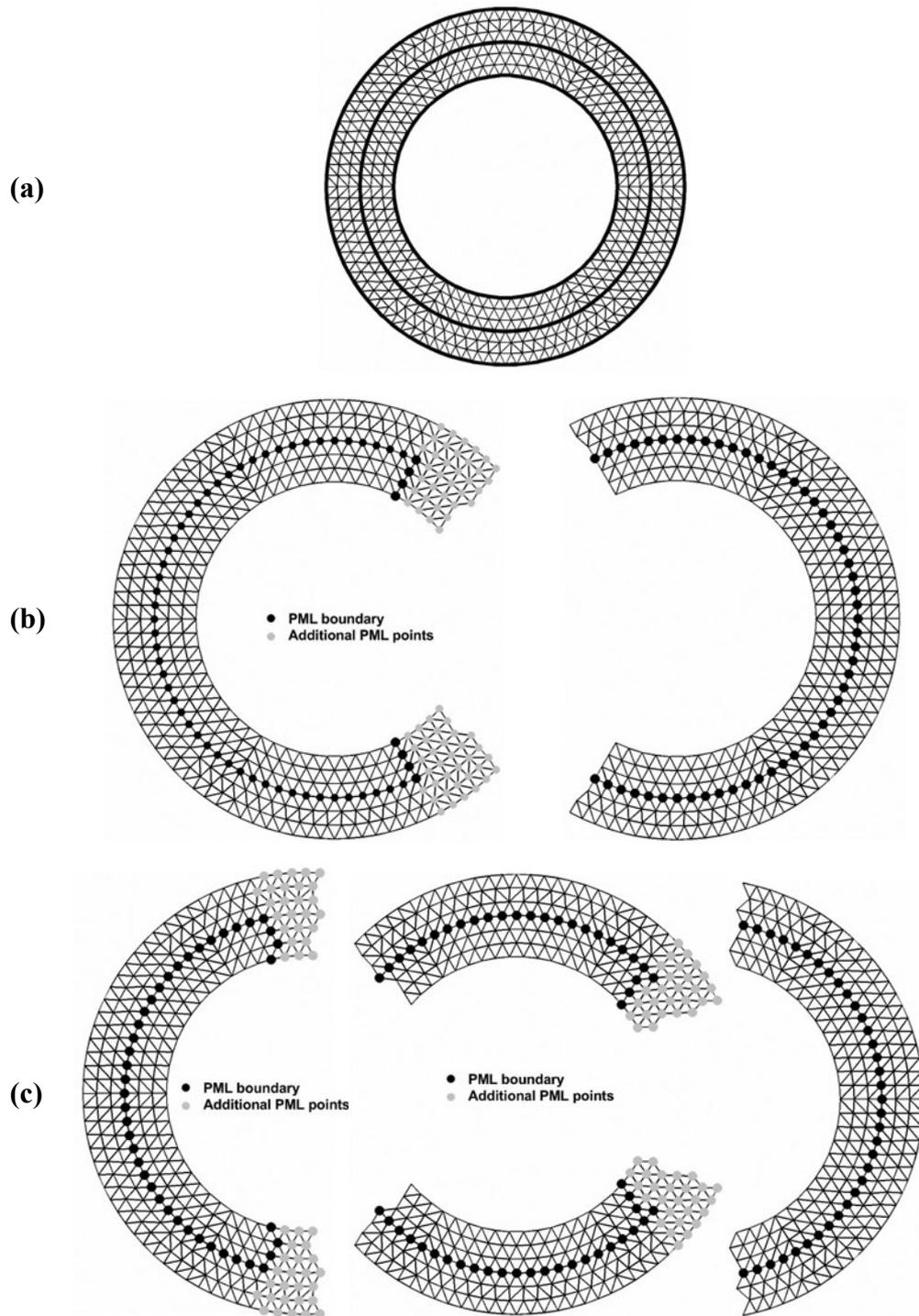


Figure A.11. Mesh partitioning of circular cylinder in the FB-DDM: (a) Original (whole) domain, (b) 2 subdomains, (c) 3 subdomains.

A.2. FEM Analysis

FEM Analysis program (*fem.m*) performs the following main tasks:

- It constructs the global matrix equation ($AX = B$) using the formulations in Chapter 2. In this matrix equation, the X values are the unknown field values at each node in the mesh. The global matrix (A), which is stored in the coordinate (COO) format (internal storage scheme of the MATLAB[®]), is assembled successively using the local matrices of the triangular elements. The program requires the mesh parameters (see Section A.1) and some input parameters for the analysis (see Table A.3).
- It solves the matrix equation ($AX = B$) using either the direct solver of the MATLAB[®] (i.e., backslash “\” operator performing the Gaussian elimination) or the iterative solution algorithms embedded in the MATLAB[®] (such as *qmr*, *bicgstab*, etc.).
- It calculates some far-field parameters from the calculated nodal field values, which are the solutions of the matrix equation. These parameters are: RCS normalized to λ in dB scale, magnitude and phase of the scattered electric field at the far-zone. (This task is applicable only for the scattering problem).

Table A.3. FEM Analysis: Format of the input parameters for the analysis.

Parameter Name	Definition
freq	Frequency in Hertz
epsr	Relative permittivity (only for dielectric object)
mur	Relative permeability (only for dielectric object)
WHICH_PML	Indicator of which PML method will be used in the program 0: Locally-conformal PML method 1: Multi-center PML method
alpha	Alpha constant in the PML implementation (positive real)
CENTER_X	x-coordinates of the centers in the multi-center PML method
CENTER_Y	y-coordinates of the centers in the multi-center PML method
CENTER_Z	z-coordinates of the centers in the multi-center PML method

APPENDIX B: 3D FEM SOFTWARE (FEMS^{3D})

FEMS^{3D} has been developed mainly in the C language, as well as the MATLAB[®] language [99] only in the mesh generation phase, to solve the vector wave equation in 3D electromagnetic scattering problems using the vector FEM with tetrahedral edge elements. The *FEMS^{3D}* consists of four independent but successive programs, which correspond to the four main phases of a general FEM procedure:

1. Mesh Generation (*meshgen_*.m*, MATLAB[®] file)
2. Main Body (*fem3dmain.exe*)
3. Matrix Solver: Iterative (*mysolver.exe*), Direct (*superlu.exe*)
4. Post-Processing (*fem3dpost.exe*)

B.1. Mesh Generation

Mesh Generation programs (*meshgen_*.m*) have been developed in the MATLAB[®] language to generate mesh parameters for some 3D objects using tetrahedral edge elements. Some functions of the programs are implemented in the external *.exe* files which are programmed in the C language, in order to accelerate the execution of the programs. The mesh library includes the following objects:

1. Cube (or rectangular prism) (*meshgen_cube.m*)
2. Sphere (*meshgen_sphere.m*)
3. Ellipsoid (or spheroid) (*meshgen_ellipsoid.m*)
4. Polygonal (convex) finite cylinder (*meshgen_cyl.m*)
5. Circular finite cylinder (*meshgen_cylcirc.m*)

6. Elliptical finite cylinder (*meshgen_cylelp.m*)
7. Polygonal (convex) plate (*meshgen_plateconvex.m*)
8. Rectangular plate (*meshgen_platerect.m*)
9. Polygonal (convex or concave) plate (*meshgen_plateconcave.m*)
10. Circular plate (*meshgen_platecirc.m*)
11. Elliptical plate (*meshgen_plateelp.m*)
12. Finite cone with circular flat-base (*meshgen_conecirc.m*)
13. Conesphere (finite cone with spherical-base) (*meshgen_conesphere.m*)
14. Rectangular inlet (*meshgen_inletrect.m*)
15. Circular inlet (*meshgen_inletcirc.m*)
16. Ogive (*meshgen_ogive.m*)
17. Pyramid with square-base (*meshgen_pyramid.m*)
18. Octahedron (2 pyramids sharing a square base) (*meshgen_octahedron.m*)
19. Dihedral (*meshgen_dihedral.m*)
20. Trihedral (*meshgen_trihedral.m*)
21. Missile (*meshgen_missile.m*)

The 3D mesh is created using the MATLAB[®] function *delaunay3*^{*}, and then some mesh parameters (such as boundary edges, boundary elements, etc.) are calculated using the mesh. The *Delaunay tessellation* is a set of simplices with the property that, for each simplex, the unique sphere circumscribed about the simplex contains no data points. In a 3D space, a simplex is a tetrahedron. In order to construct a proper mesh, the quality of each tetrahedral element is computed according to some criteria, which is given by the following formula (Robert/Roux parameter) [102]

$$Q_{3D} = \left(\frac{9\pi}{2\sqrt{3}} \cdot \frac{\text{True Volume}}{\text{Sphere Volume}} \right)^{1/3} \quad (\text{B.1})$$

* The function *delaunay3* in MATLAB[®] uses actually an external program, called as *Qhull* [100].

where the sphere volume is the volume of the circumscribing sphere. This parameter has a minimum value of zero and equals to 1 for a regular tetrahedron.

How to Run: Since the Mesh Generation programs are MATLAB[®] source files, the user must just write the name of the program file (*meshgen_*.m*) on the MATLAB[®] command window to execute the program. For instance, in order to execute the program for a spherical object, the user must write the following:

Example: >> meshgen_sphere

Some examples from the mesh library are shown in the figures in between Figure B.1 and Figure B.20. In addition, for the DDM algorithms, some examples from the mesh partitioning schemes are shown in the figures in between Figure B.21 and Figure B.24.

Input Parameters: The input parameters must be entered manually at the beginning of each Mesh Generation program file. In addition to the input parameters which define specifically the geometry of each object (such as radius of sphere, angle of cone, etc.), there are also some common parameters appearing in all program files (see Table B.1).

Table B.1. Mesh Generation: List of common input parameters.

Parameter Name	Definition / Unit
scat_type	1 or 2 1: object is PEC 2: object is dielectric (not applicable for plates)
scat_shape_type	1 ... 21 ID no defining the shape of the object in mesh library. If unknown, it may be zero.
delh	Approximate edge size per element (meter)
dpml	Distance between the object and PML boundaries (meter)
dhuyg	Distance between object and Huygens surface boundaries (meter)
dvoi	Distance between the PML and VOI boundaries (meter)

Output Parameters: At the end of the program, the output parameters are saved in a text file (*meshgen.txt*), whose contents are listed in Table B.2 line by line.

Table B.2. Mesh Generation: Format of the output text file.

delh (approximate edge size per element in meter)
scat_shape_type (1...20, object shape type, or zero if unknown)
scat_type (1 or 2, scattering type, PEC or dielectric)
Nelement (number of elements)
Nnode (number of nodes)
Nedge (number of edges)
dummy line (this line is skipped in the program)
1st component of x-coordinates 2nd component of x-coordinates : (Nnode)th component of x-coordinates
dummy line (this line is skipped in the program)
1st component of y-coordinates 2nd component of y-coordinates : (Nnode)th component of y-coordinates
dummy line (this line is skipped in the program)
1st component of z-coordinates 2nd component of z-coordinates : (Nnode)th component of z-coordinates
dummy line (this line is skipped in the program)
(1,1) (1,2) (1,3) (1,4) (1,5) (1,6) components of edge-connectivity-matrix (2,1) (2,2) (2,3) (2,4) (2,5) (2,6) components of edge-connectivity-matrix : (Nelement,1) (Nelement,2) (Nelement,3) (Nelement,4) (Nelement,5) (Nelement,6) components of edge-connect.-matrix
dummy line (this line is skipped in the program)

Table B.2. (continued)

<p>(1,1) (1,2) (1,3) (1,4) components of node-connectivity-matrix (2,1) (2,2) (2,3) (2,4) components of node-connectivity-matrix : (Nelement,1) (Nelement,2) (Nelement,3) (Nelement,4) components of node-connectivity-matrix</p>
<p>dummy line (this line is skipped in the program)</p>
<p>(1,1) (1,2) components of edge-to-node-connectivity-matrix (2,1) (2,2) components of edge-to-node-connectivity-matrix : (Nedge,1) (Nedge,2) components of edge-to-node-connectivity-matrix</p>
<p>dummy line (this line is skipped in the program)</p>
<p>Npmlb_node (Size of the array containing the IDs of nodes on the PML boundary)</p>
<p>1st component of pml-boundary-nodes array 2nd component of pml-boundary-nodes array : (Npmlb_node)th component of pml-boundary-nodes array</p>
<p>dummy line (this line is skipped in the program)</p>
<p>Npmlin_node (Size of the array containing the IDs of nodes inside the PML region)</p>
<p>1st component of pml-inside-nodes array 2nd component of pml-inside-nodes array : (Npmlin_node)th component of pml-inside-nodes array</p>
<p>dummy line (this line is skipped in the program)</p>
<p>Nvoib_node (Size of the array containing the IDs of nodes on the VOI boundary)</p>
<p>1st component of voi-boundary-nodes array 2nd component of voi-boundary-nodes array : (Nvoib_node)th component of voi-boundary-nodes array</p>
<p>dummy line (this line is skipped in the program)</p>
<p>Nvoib_edge (Size of the array containing the IDs of edges on the VOI boundary)</p>
<p>1st component of voi-boundary-edges array 2nd component of voi-boundary-edges array : (Nvoib_edge)th component of voi-boundary-edges array</p>

Table B.2. (continued)

(if PEC object):
dummy line (this line is skipped in the program)
Nscatb_elm (Number of row items in the matrix array containing the IDs of elements on the object boundary)
(1,1) (1,2) (1,3) (1,4) (1,5) components of scatterer-boundary-elements-matrix (2,1) (2,2) (2,3) (2,4) (2,5) components of scatterer-boundary-elements-matrix : (Nscatb_elm,1) (Nscatb_elm,2) (Nscatb_elm,3) (Nscatb_elm,4) (Nscatb_elm,5) components of scatterer-boundary-elements-matrix
dummy line (this line is skipped in the program)
Nscatb_edge (Size of the array containing the IDs of edges on the object boundary)
1st component of scatterer-boundary-edges array 2nd component of scatterer-boundary-edges array : (Nscatb_edge)th component of scatterer-boundary-edges array
(else if dielectric object):
dummy line (this line is skipped in the program)
1st component of isinside_elm array (the array indicating whether the element is inside the object, or not, ones and zeros) 2nd component of isinside_elm array : (Nelement)th component of isinside_elm array
dummy line (this line is skipped in the program)
Nhuygb_elm (Number of row items in the matrix array containing the IDs of elements on the Huygens surface boundary)
(1,1) (1,2) (1,3) (1,4) (1,5) components of huygens-boundary-elements-matrix (2,1) (2,2) (2,3) (2,4) (2,5) components of huygens-boundary-elements-matrix : (Nhuygb_elm,1) (Nhuygb_elm,2) (Nhuygb_elm,3) (Nhuygb_elm,4) (Nhuygb_elm,5) comps of huyg-bound-elms-mat.

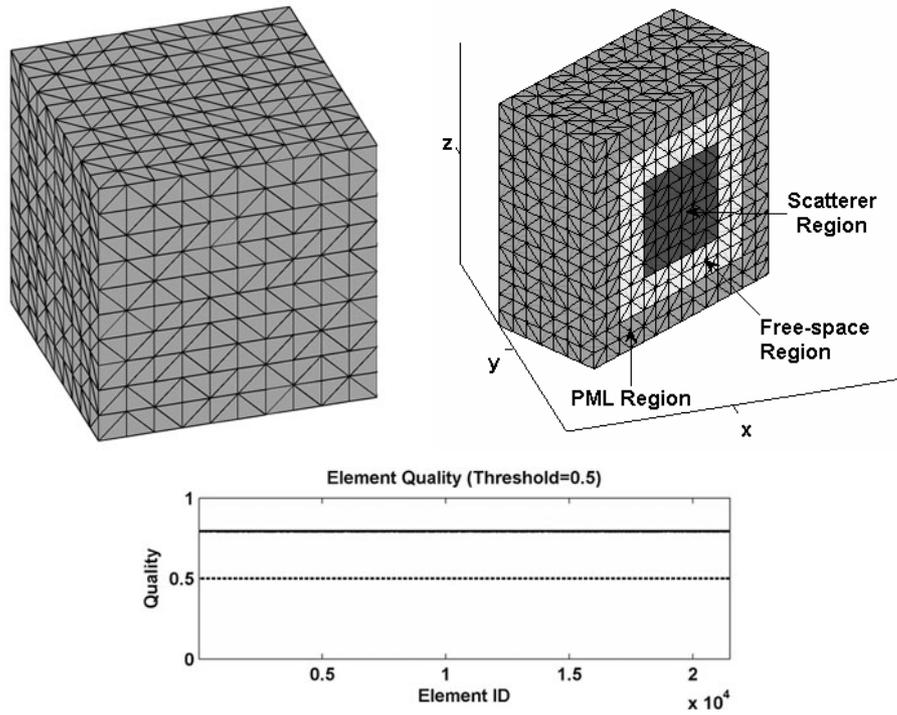


Figure B.1. Mesh Library, Cube.

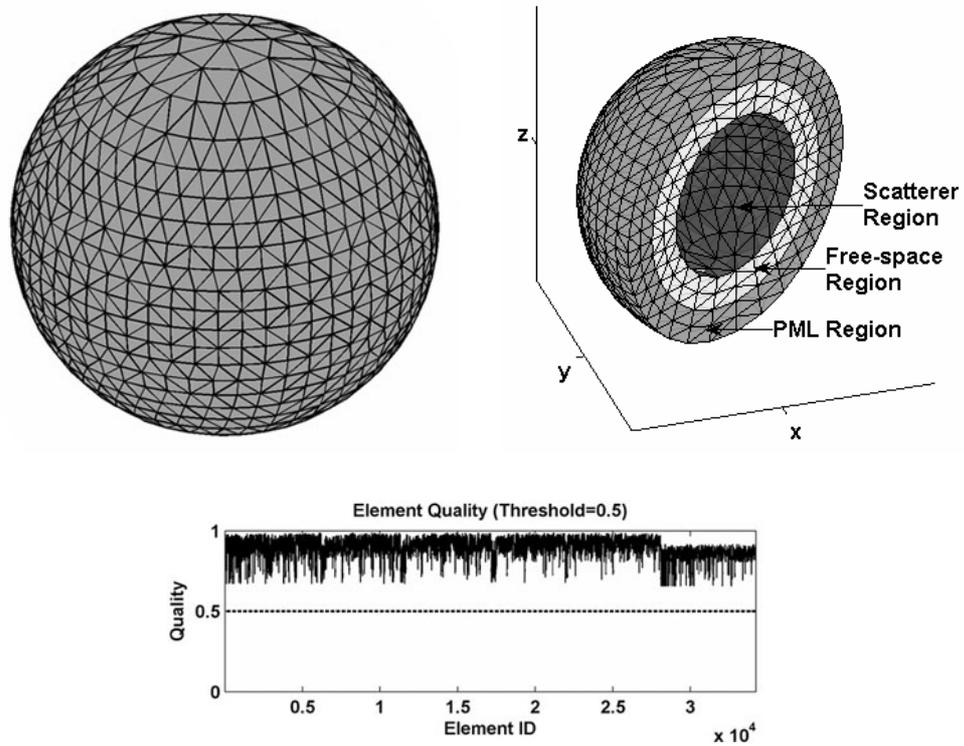


Figure B.2. Mesh Library, Sphere.

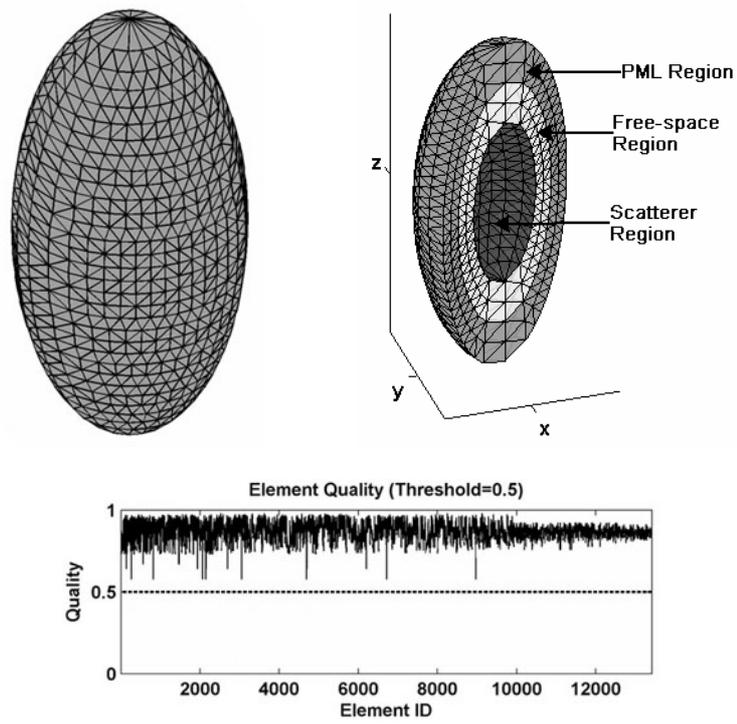


Figure B.3. Mesh Library, Ellipsoid (or spheroid).

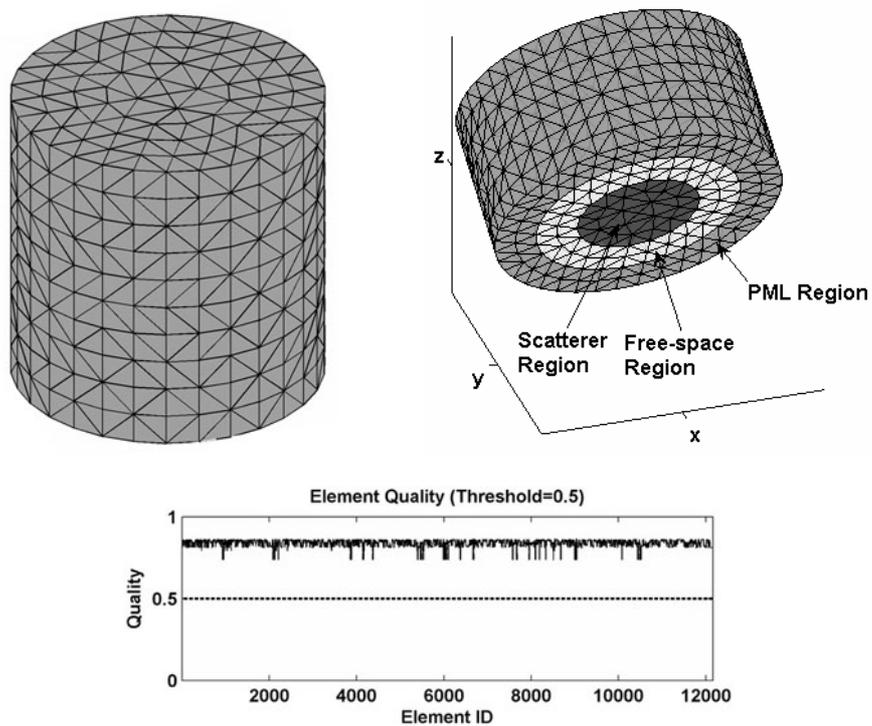


Figure B.4. Mesh Library, Circular Finite Cylinder.

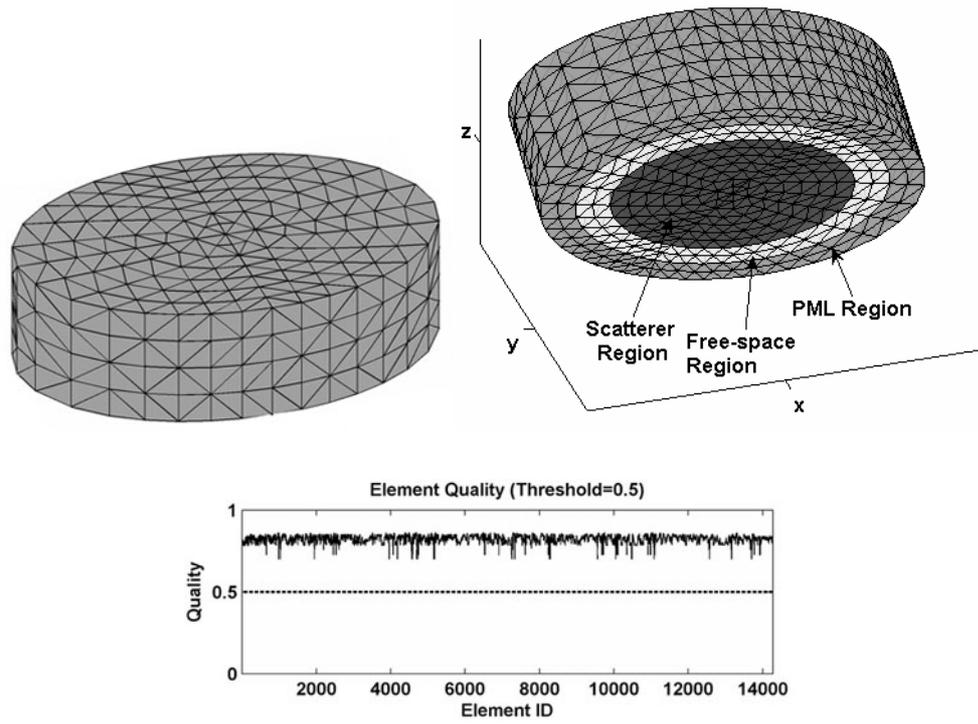


Figure B.5. Mesh Library, Elliptical Finite Cylinder.

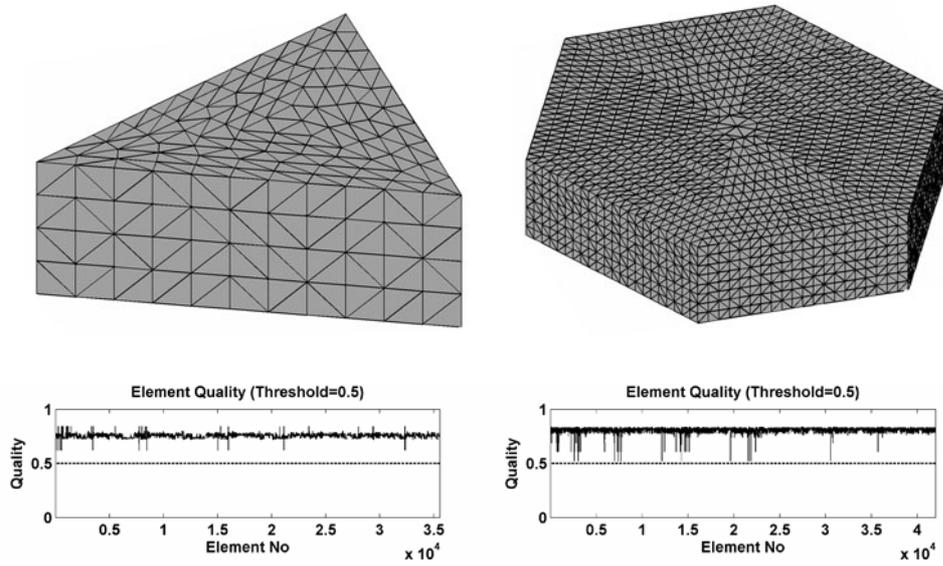


Figure B.6. Mesh Library, Convex Polygonal Finite Cylinder.

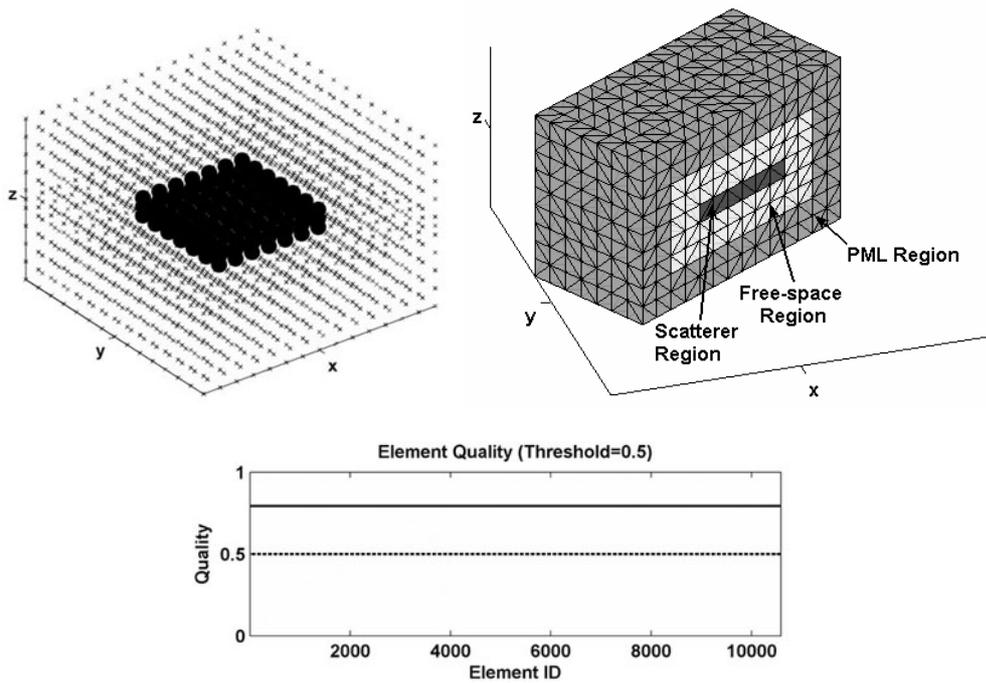


Figure B.7. Mesh Library, Rectangular Plate.

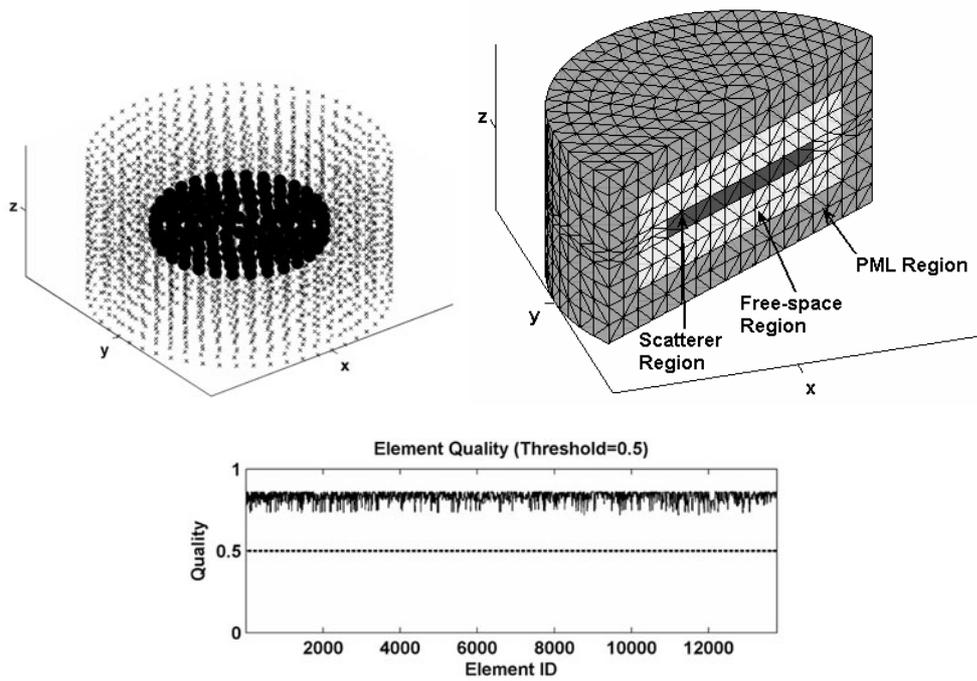


Figure B.8. Mesh Library, Circular Plate.

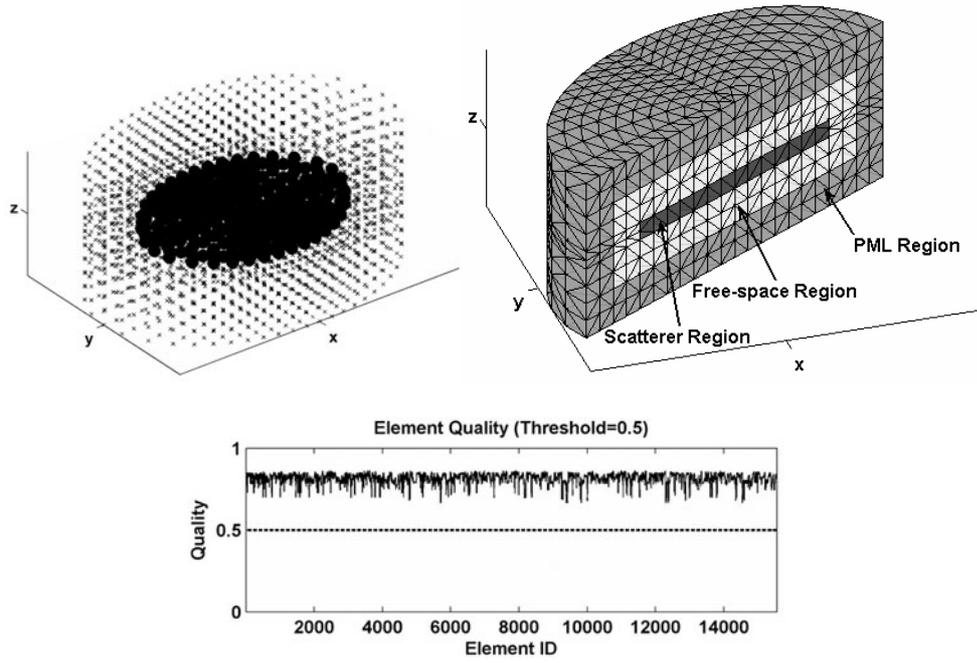


Figure B.9. Mesh Library, Elliptical Plate.

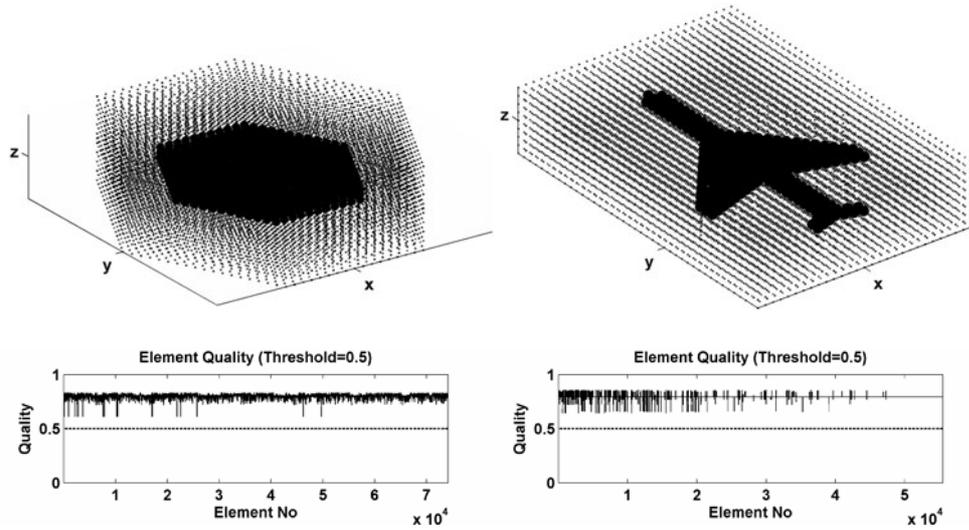


Figure B.10. Mesh Library, Polygonal Plate.

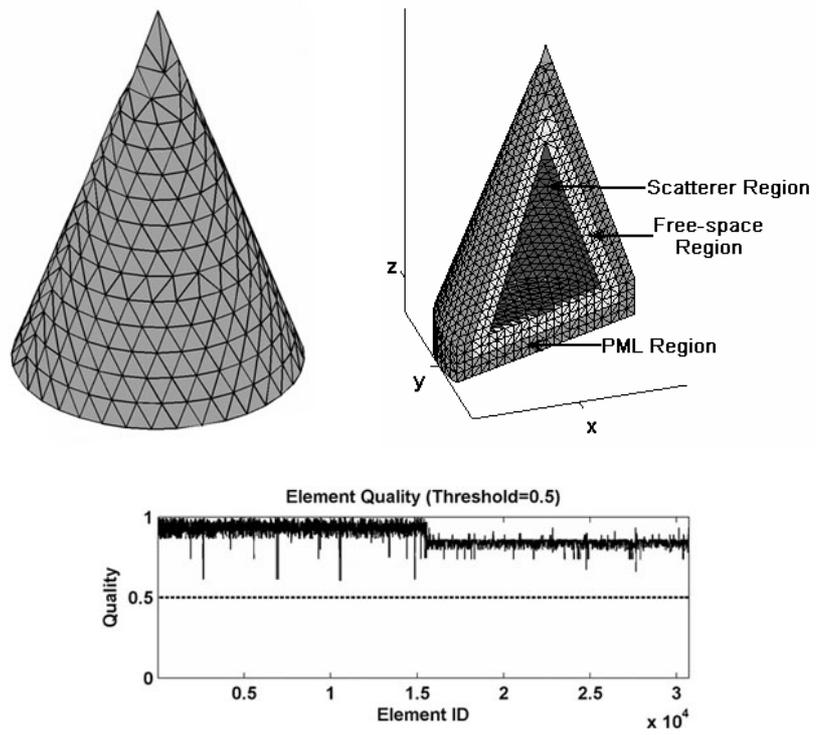


Figure B.11. Mesh Library, Finite Cone with Circular-base.

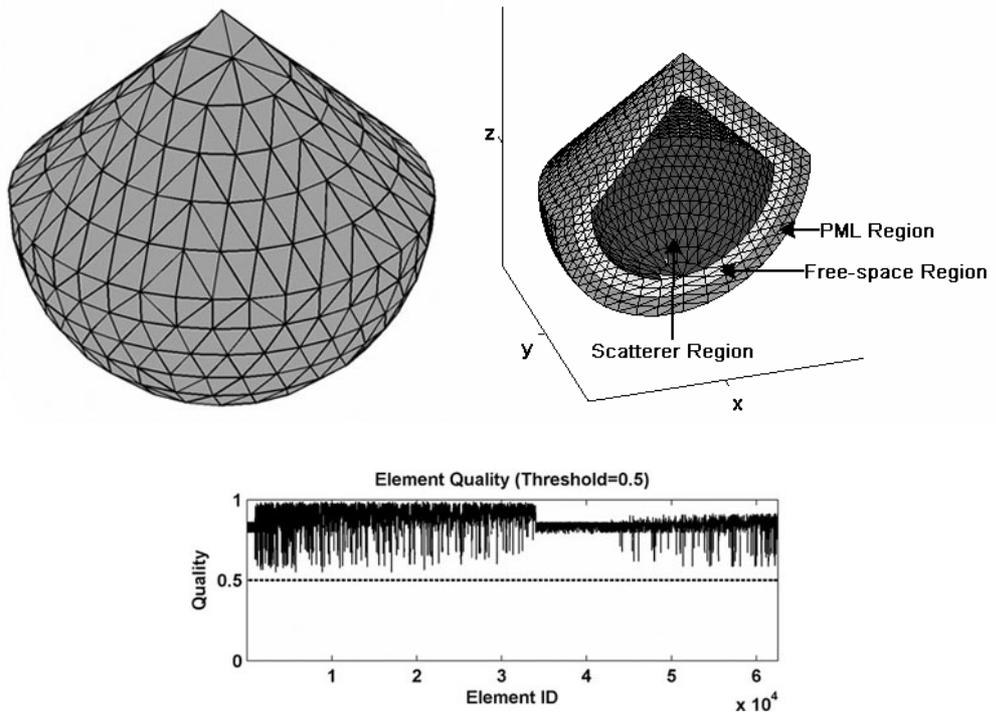


Figure B.12. Mesh Library, Conesphere (finite cone with spherical-base).

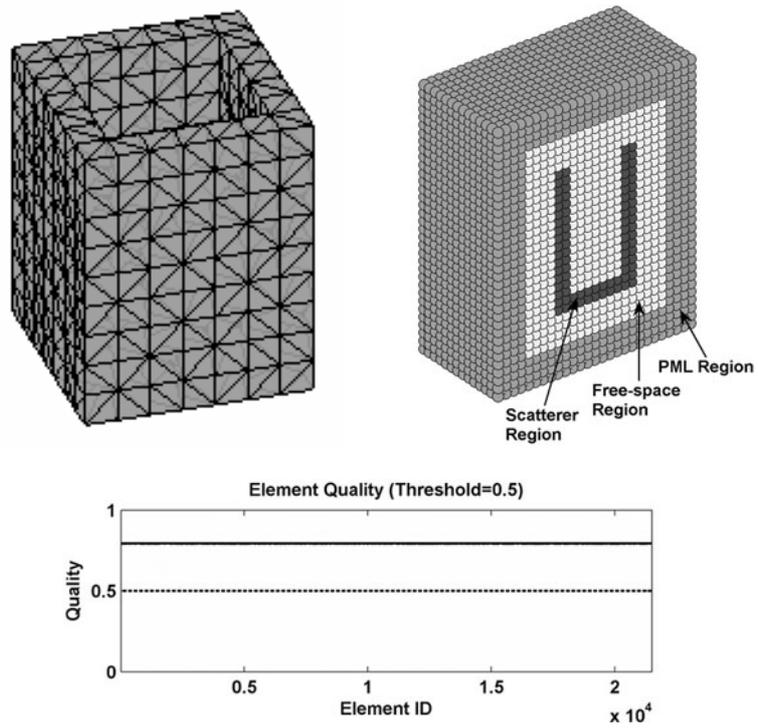


Figure B.13. Mesh Library, Rectangular Inlet.

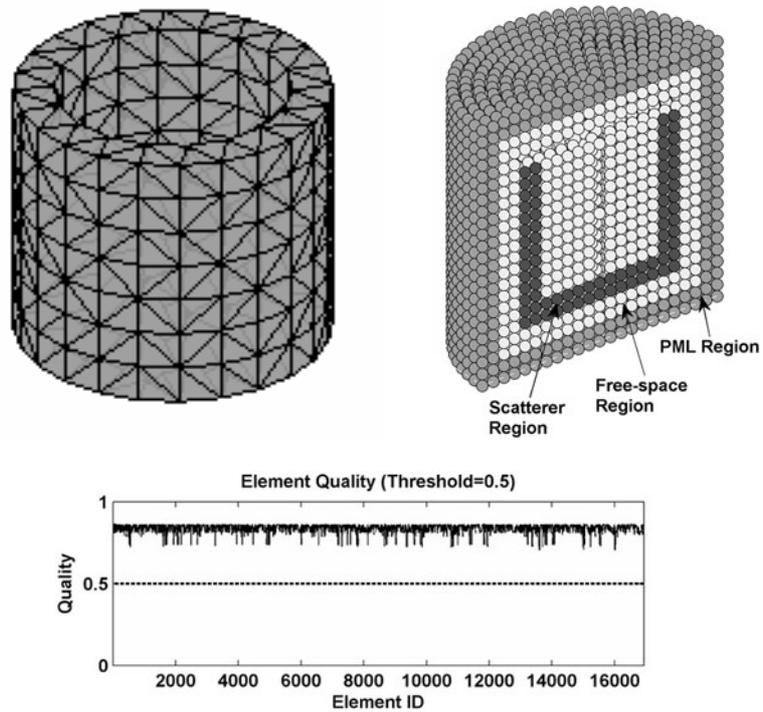


Figure B.14. Mesh Library, Circular Inlet.

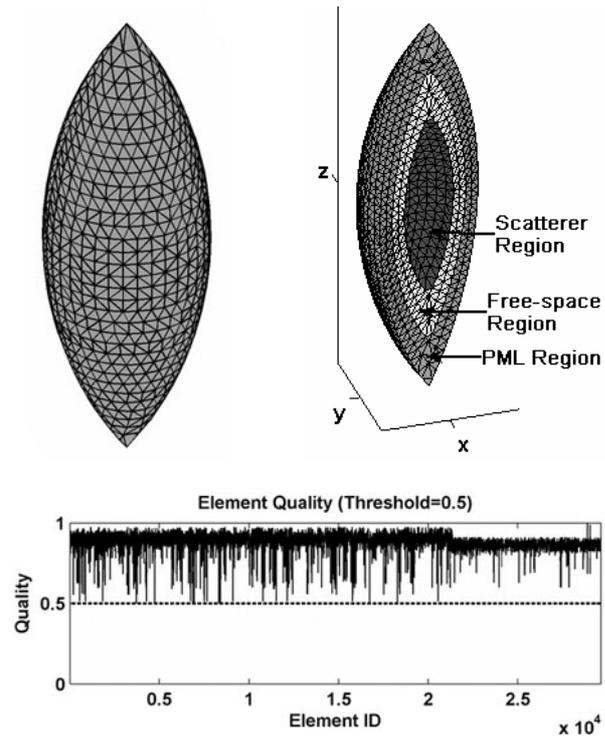


Figure B.15. Mesh Library, Ogive.

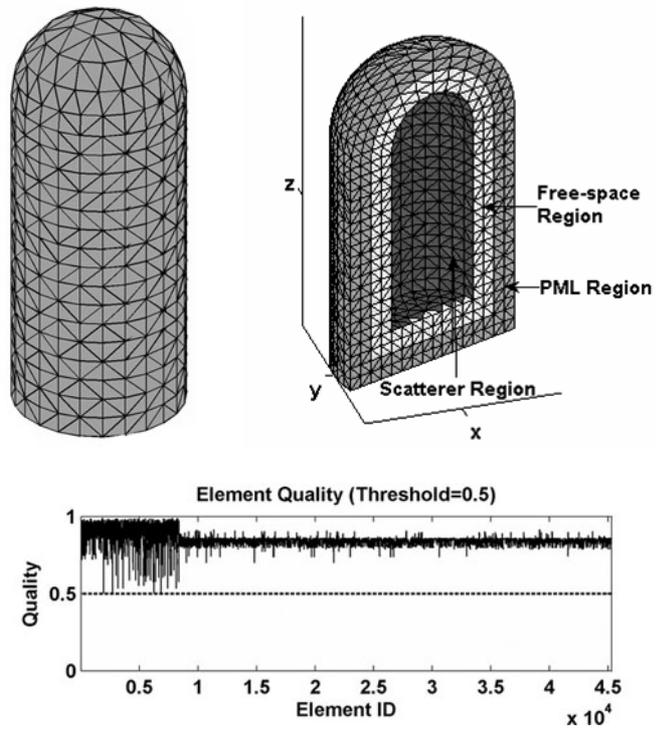


Figure B.16. Mesh Library, Missile.

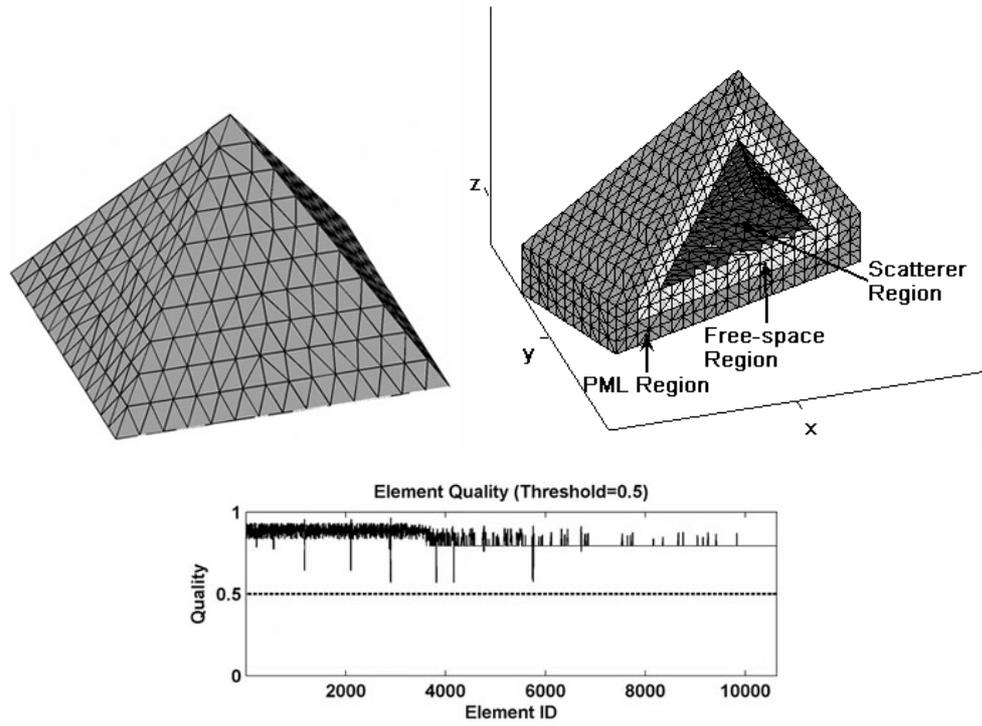


Figure B.17. Mesh Library, Pyramid with Square-base.

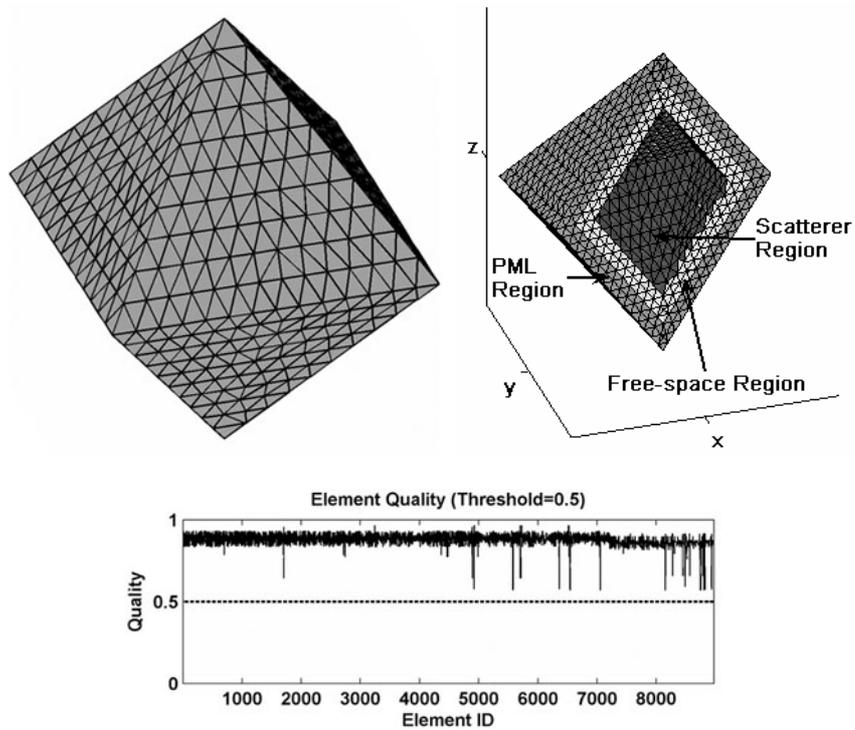


Figure B.18. Mesh Library, Octahedron.

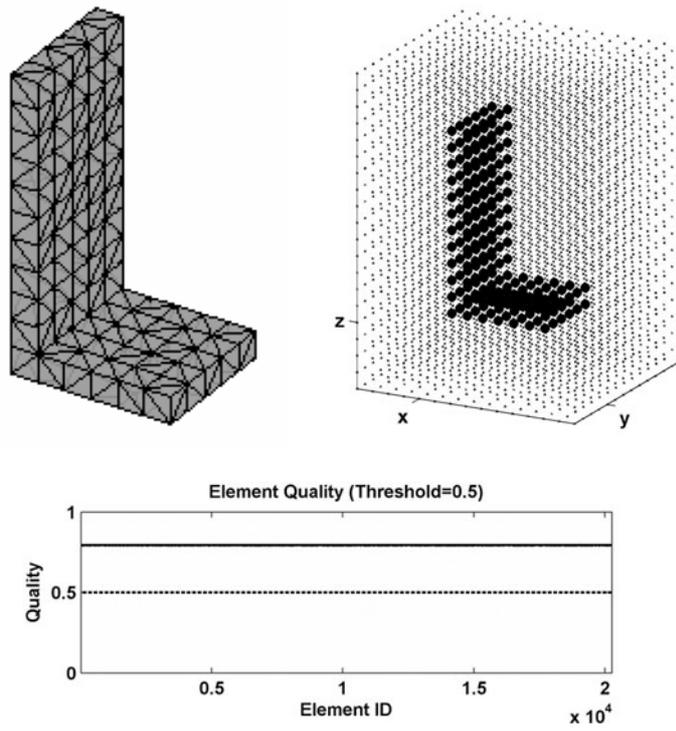


Figure B.19. Mesh Library, Dihedral.

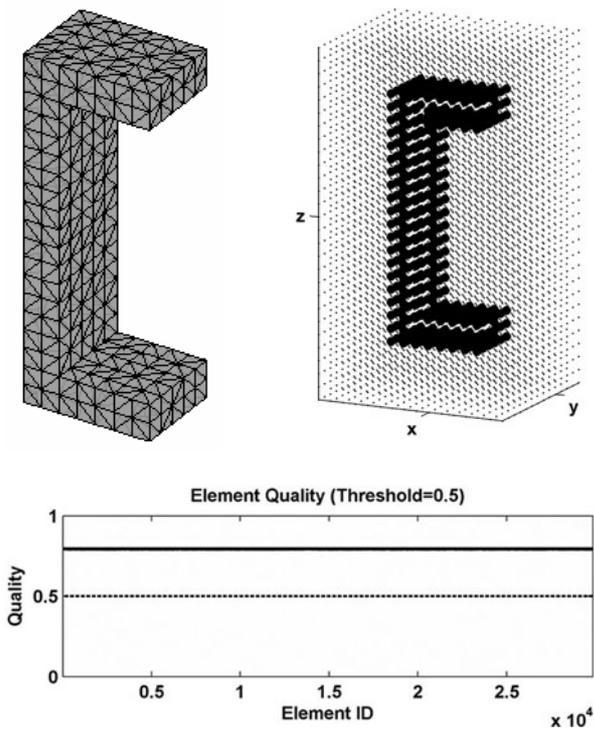


Figure B.20. Mesh Library, Trihedral.

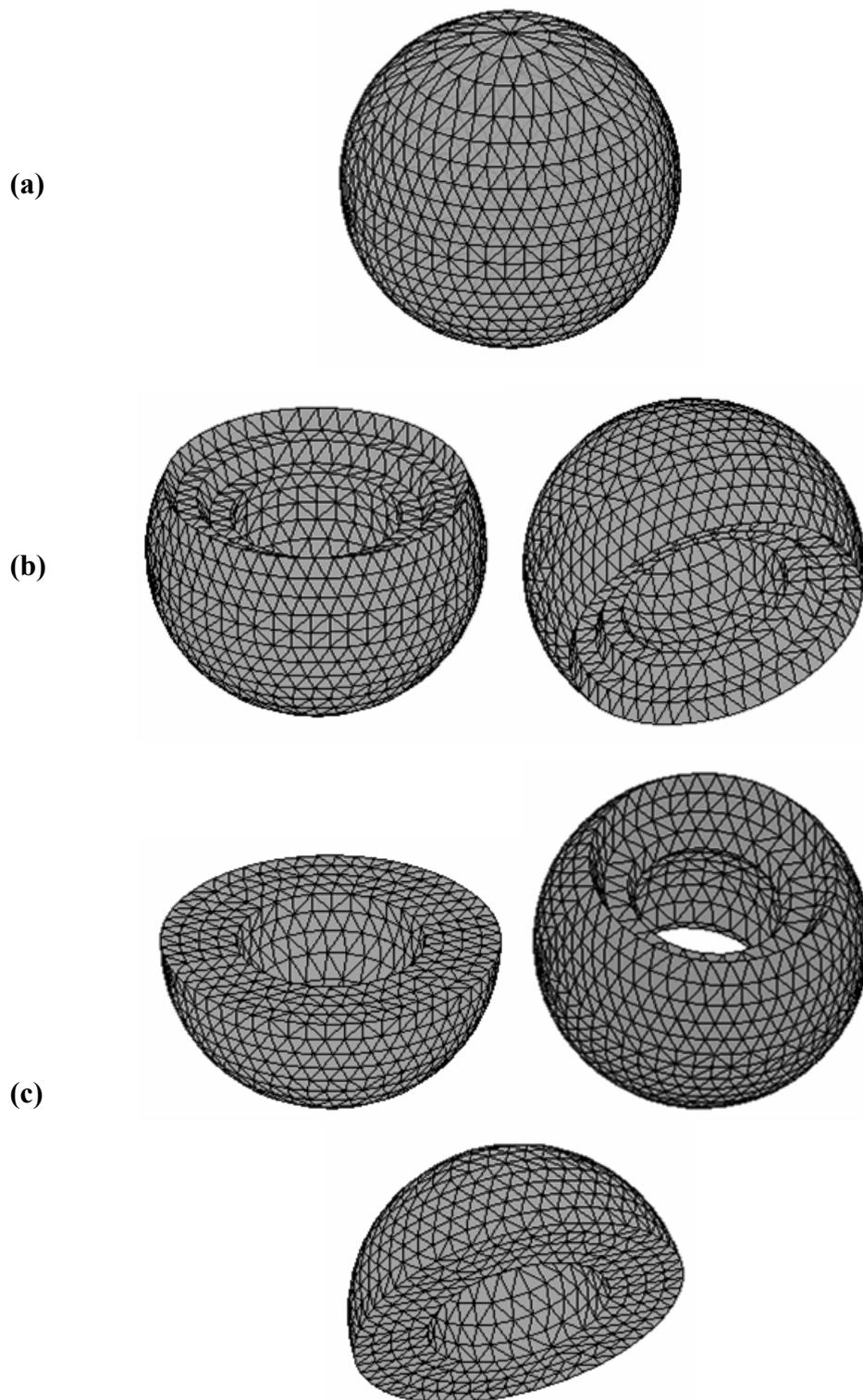


Figure B.21. Mesh partitioning of sphere in the FB-DDM: (a) Original (whole) domain, (b) 2 subdomains, (c) 3 subdomains.

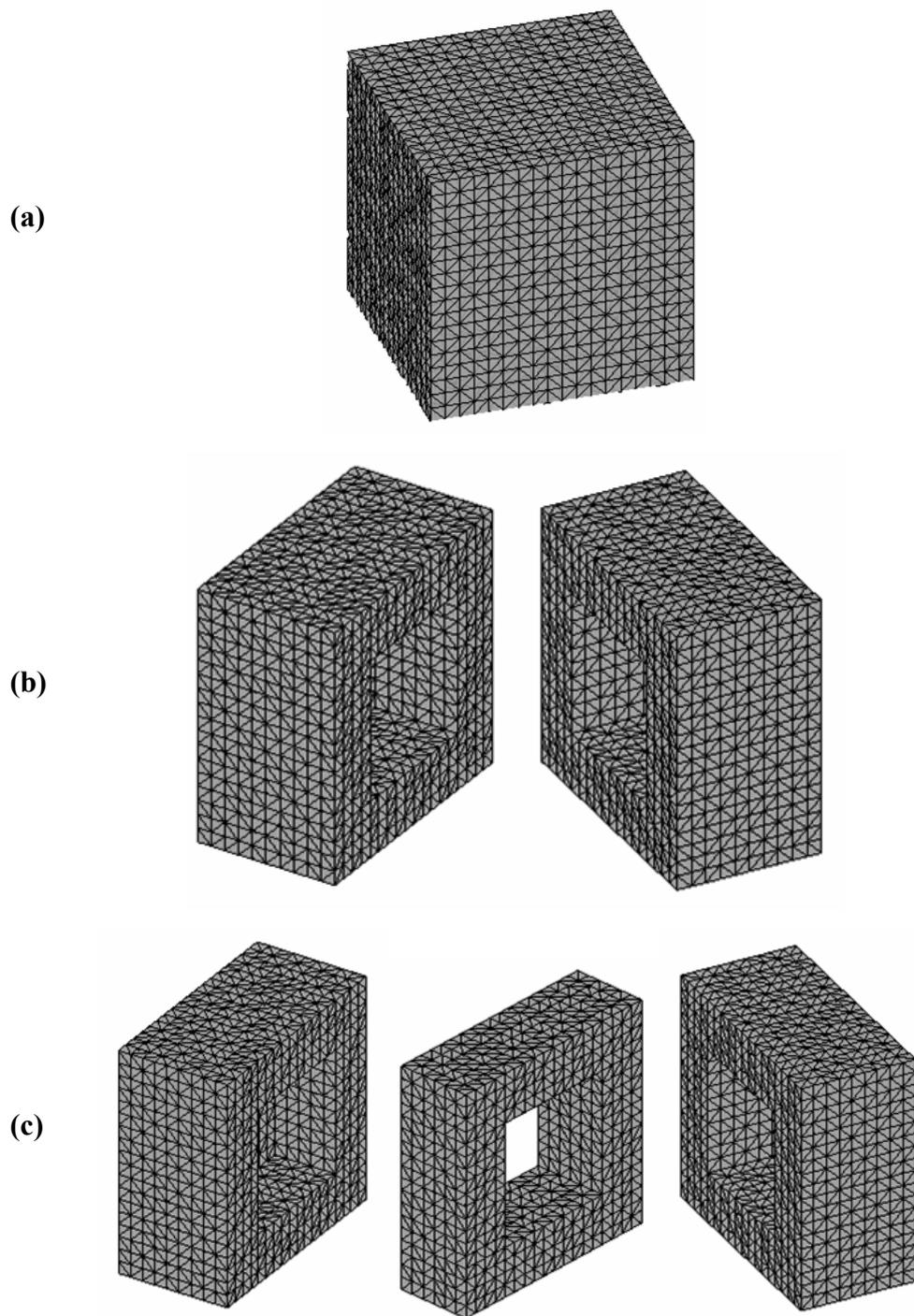


Figure B.22. Mesh partitioning of cube in the FB-DDM: (a) Original (whole) domain, (b) 2 subdomains, (c) 3 subdomains.

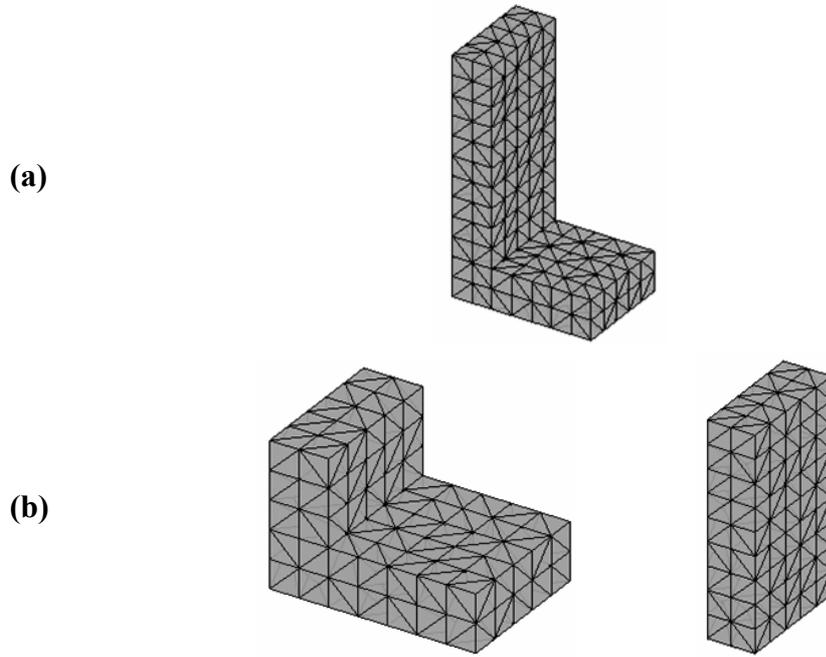


Figure B.23. Mesh partitioning of dihedral in the ILF-DDM: (a) Original (whole) domain, (b) 2 subdomains.

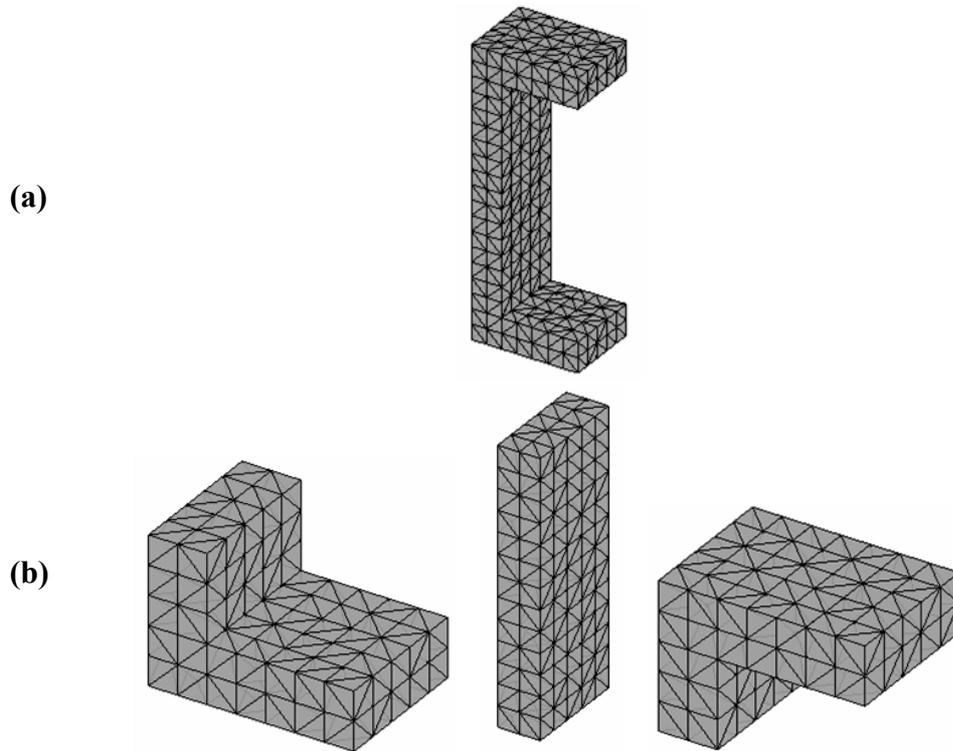


Figure B.24. Mesh partitioning of trihedral in the ILF-DDM: (a) Original (whole) domain, (b) 3 subdomains.

B.2. Main Body

FEMS^{3D} Main Body program has been developed in the C language for constructing the global matrix equation ($AX = B$) using the formulations in Chapter 3. In this matrix equation, the X values are the unknown field values along each edge in the mesh. The global matrix (A), which is stored in “Compressed Sparse Row” (CSR) format, is assembled successively using the local matrices of the tetrahedral elements. The Main Body program requires the mesh parameters (see Section B.1) and some input parameters for the analysis.

How to Run: The Main Body program can be compiled using any C/C++ compiler. After the compilation of all files, the execution file *fem3dmain.exe* is created in the operating directory. This execution file uses two input text files (mesh and analysis input files), which are *compulsory* to execute the program; and an *optional* input text file. The execution file also requires the name of an output file, where the constructed matrix equation ($AX = B$) is written. In addition to this compulsory output file, an optional output file is also created after the successful execution of the program. While executing the program, an analysis input file must exist in the operating directory. In order to execute the Main Body program, the following command must be written on the command shell.

```
fem3dmain [compulsory mesh input file name] [compulsory output file name]
```

```
Example:    > fem3dmain  meshgen.txt  AB.txt
```

Input Parameters: This section describes the structure of both the compulsory and the optional input files. The compulsory mesh input text file must be in the form of Table B.2 as described in Section B.1. The second compulsory input text file, whose default name is *analysis.txt*, contains the input parameters related to the finite element implementation of the scattering problem. If this file does not exist in the directory, the program prompts the user to write the name of another

file or to press enter to exit the program. The contents of the file *analysis.txt* are listed in Table B.3 line by line, and an example file is illustrated in Figure B.25.

Table B.3. Main Body: Format of the input text file for the analysis.

FREQ [value]	Frequency in Hertz
ku [value] [value] [value]	x-y-z components of the unit-direction vector \vec{a}_k
E0inc [value] [value] [value] [value] [value] [value]	Real and imaginary parts of the x-component of the incident plane wave Real and imaginary parts of the y-component of the incident plane wave Real and imaginary parts of the z-component of the incident plane wave
EPSR [value]	Relative permittivity (this is used only for dielectric object)
MUR [value]	Relative permeability (this is used only for dielectric object)
WHICH_PML [0 or 1]	Indicator of which PML method will be used in the program. 0: Locally-conformal PML method 1: Multi-center PML method
ALPHA [value]	Alpha constant in the PML implementation (positive real).
PML_M [value]	Decay parameter in the PML implementation (positive integer)
NCENTER [value]	Number of the centers in the multi-center PML method. (this is used if WHICH_PML=1)
CENTER_X [value1] [value2] ... [value-Ncenter]	x-coordinates of the centers in the multi-center PML method (this is used if WHICH_PML=1)
CENTER_Y [value1] [value2] ... [value-Ncenter]	y-coordinates of the centers in the multi-center PML method (this is used if WHICH_PML=1)
CENTER_Z [value1] [value2] ... [value-Ncenter]	z-coordinates of the centers in the multi-center PML method (this is used if WHICH_PML=1)
USE_CSR_FILE [1 or 0]	The indicator of whether any file containing the CSR arrays (IA, JA) will be used. 1: CSR arrays are read from an input file 0: CSR arrays are constructed during the program.

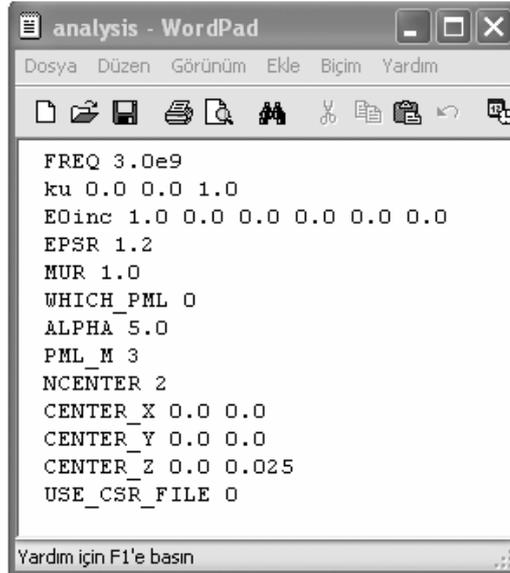


Figure B.25. Main Body: Example file for the input analysis file.

The optional input text file, whose default name is *incsr.txt*, contains the CSR arrays (*IA*, *JA*). This file is used only if the *USE_CSR_FILE* parameter is set equal to 1. If this file does not exist in the directory, the program prompts the user to write the name of another file or to press enter to create the matrices during the execution of the program. The contents of this file are listed in Table B.4 line by line.

Table B.4. Main Body: Format of the file containing CSR arrays.

n (matrix size)
nz (number of the nonzero entries in the matrix A)
dummy line (this line is skipped in the program)
1st component of IA 2nd component of IA : (n+1)st component of IA
dummy line (this line is skipped in the program)
1st component of JA 2nd component of JA : (nz)th component of JA

Output Parameters: This section describes the structure of both the compulsory and the optional output files. At the end of the program, the final matrix equation ($AX = B$) is written on the compulsory output file whose name is previously described in the command shell (e.g. *AB.txt*). The contents of this file, which includes both A in CSR format and B , are listed in Table B.5 line by line. In this file, AA , JA and IA are three matrices related to the CSR format, and are defined as follows:

- AA : $n_z \times 1$ complex matrix containing the nonzero entries of the matrix by row-wise. Here, n_z is the number of nonzero entries in the matrix A .
- JA : $n_z \times 1$ integer matrix containing the column indices of the nonzero entries of the matrix by row-wise.
- IA : $(n+1) \times 1$ integer row-pointer matrix, which points to the beginning of each row. Here, n is the size of the matrix A .

Table B.5. Main Body: Format of the file containing the matrix equation.

n (matrix size)	
nz (number of the nonzero entries in the matrix A)	
dummy line (this line is skipped in the program)	
Real part of 1st component of AA	Imaginary part of 1st component of AA
Real part of 2nd component of AA	Imaginary part of 2nd component of AA
:	:
Real part of (nz)th component of AA	Imaginary part of (nz)th component of AA
dummy line (this line is skipped in the program)	
1st component of IA	
2nd component of IA	
:	
(n+1)st component of IA	
dummy line (this line is skipped in the program)	
1st component of JA	
2nd component of JA	
:	
(nz)th component of JA	
dummy line (this line is skipped in the program)	
Real part of 1st component of B	Imaginary part of 1st component of B
Real part of 2nd component of B	Imaginary part of 2nd component of B
:	:
Real part of (n)th component of B	Imaginary part of (n)th component of B

In addition, the optional output file, whose default name is *outcsr.txt*, is created only if the `USE_CSR_FILE` parameter is set equal to zero. This file also contains the CSR arrays (*IA*, *JA*), and must be in the same form in Table B.4.

B.3. Matrix Solver

FEMS^{3D} Matrix Solver programs can be categorized into two groups: Iterative solver program (*mysolver.exe*) and direct solver program (*superlu.exe*). The Iterative Matrix Solver program has been developed in the C language by converting and adapting the FORTRAN routines of the SPARSKIT package [103]. The Direct Matrix Solver program has also been developed in the C language by adapting the C routines of the SUPERLU package [104], which performs an LU decomposition with partial pivoting and triangular system solutions through forward and back substitution. It is worth mentioning that the Direct Matrix Solver program *cannot* handle “large” matrices that contain very large number of unknowns (for example, larger than 40,000 unknowns). Hence, the Iterative Matrix Solver program provides a wider range of applicability for large problems.

B.3.1. Iterative Matrix Solver

Iterative Matrix Solver program has been developed to solve sparse linear system equations ($AX = B$) using some iterative solver algorithms combined with some pre-conditioning techniques. The matrix A is a complex, square, nonsingular, $n \times n$ sparse matrix; and X and B are dense $n \times 1$ matrices corresponding to the solution and right-hand-side (RHS) vectors, respectively. The matrix A need not to be symmetric and definite; indeed the Iterative Matrix Solver program is appropriate for indefinite matrices and matrices with unsymmetric structure. In addition, the matrix A must be stored in the CSR format.

The Iterative Solver program includes the following iterative solver algorithms:

- GMRES (Generalized Minimum Residual Method)
- QMR (Quasi-Minimal Residual Method)
- BICGSTAB (BiConjugate Gradients Stabilized Method)

The Iterative Solver program is also capable of applying some pre-conditioning techniques, before solving the system of equations with above-mentioned algorithms. The program includes the following pre-conditioning techniques:

- Point pre-conditioners:
 - Jacobi
 - Gauss-Seidel
 - SSOR (Symmetric Successive Overrelaxation)
- ILU (Incomplete LU) pre-conditioners:
 - ILUT (ILU factorization with dual truncation mechanism)
 - ILU(k) (ILU with level of fill-in of k)
 - ILU(0) (ILU with zero-level)

How to Run: The Iterative Matrix Solver program can be compiled using any C/C++ compiler. After the compilation of all files, the execution file *mysolver.exe* is created in the operating directory. This execution file uses an input text file, which is *compulsory* to execute the program, and some *optional* input text files. The execution file requires also the name of an output file, where the final solution matrix (X) is written. In addition to this compulsory output file, an optional output file is also created after the successful execution of the program. In order to execute the Iterative Matrix Solver program, the following command must be written on the command shell:

```
mysolver [compulsory input file name] [compulsory output file name]
```

```
Example:    > mysolver AB.txt sol.txt
```

Input Parameters: This section describes the structure of both the compulsory and the optional input files. The compulsory input text file containing the entries of the matrix equation must be in the form of Table B.5 as described in Section B.2.

The Iterative Matrix Solver program also uses some optional input files. The first optional input text file, whose default name is *insolver.txt*, contains the input parameters related to the implementation of both the iterative solver algorithms and the pre-conditioning techniques. If this file does not exist in the directory, the program prompts the user to write the name of another file or to press enter to use the default values of the parameters. The contents of this file are listed in Table B.6 line by line, and an example file is shown in Figure B.26.

Table B.6. Matrix Solver: Format of the optional input file.

PTYPE [0 ... 6]	ID no of the pre-conditioning technique: 0: No-preconditioning 1: Jacobi 2: Gauss-Seidel 3: SSOR 4: ILUT 5: ILU(k) 6: ILU(0) If this file does not exist, default value = 0
STYPE [1 ... 3]	ID no of the iterative solver algorithm: 1: GMRES 2: QMR 3: BICGSTAB If this file does not exist, default value = 1
TOLERANCE [value]	Convergence tolerance of the iterative solver algorithm. If this file does not exist, default value = 1e-16
MAXITS [value]	Maximum number of iterations. If this file does not exist, default value = 1000 After MAXITS iterations, the program prompts the user to press "1" to perform additional MAXITS iterations, or to press enter to exit the program. If the user presses "1", the temporary solution vector is written on a file <i>tempout.txt</i> , and the program continues with new iterations.

Table B.6. (continued)

USE_PC_FILE [1 or 0]	Indicator of whether any file containing the matrices related to the pre-conditioning techniques will be used. 1: Pre-conditioning matrices are read from an input file 0: Pre-conditioning matrices are created during the execution If this file does not exist, default value = 0
WRITE_PC_FILE [1 or 0]	Indicator of whether the matrices related to the pre-conditioning techniques will be written on an output file. 1: Pre-conditioning matrices are written on an output file 0: Pre-conditioning matrices are not written on an output file If this file does not exist, default value = 0
USE_X0_FILE [1 or 0]	Indicator of whether any file containing the initial solution matrix ($X0$) will be used. 1: Initial solution matrix is read from an input file 0: Initial solution matrix is set equal to all zero. If this file does not exist, default value = 0
SHOW_ITER [1 or 0]	Indicator of whether the iteration steps will be displayed on the command shell. 1: Iteration steps will be displayed on the command shell. 0: Iteration steps will not be displayed on the command shell. If this file does not exist, default value = 1
KRYLOV_SIZE [value]	Size of the Krylov subspace. It is only used in GMRES (if STYPE is 1), and should not exceed 50. If this file does not exist, default value = 20
ILU_DROPTOL [value]	Threshold for dropping small terms in the ILU factorization. It is only used in ILUT and ILU(k) (if PTYPE is 4 and 5) If this file does not exist, default value = 1e-3
ILU_LFIL [value]	Fill-in parameter. It is only used in ILUT and ILU(k) (if PTYPE is 4 and 5), and must be greater than zero. In ILUT: Each row of L and each row of U will have a maximum of lfil elements (excluding the diagonal element). In ILU(k): Each element whose level-of-fill exceeds lfil during the ILU process is dropped. If this file does not exist, default value = 100 for ILUT (if matrix size n is less than 100, default value is set to n) If this file does not exist, default value = 3 for ILU(k)
ILU_NNZ [value]	Factor for the minimum length of pre-conditioning matrices. It is only used in ILUT and ILU(k) (if PTYPE is 4 and 5). The length of pre-conditioning matrices is $ILU_NNZ \times n_z$. If this parameter is not big enough to store matrices of the ILU factorization, the program will stop with an error message. If this file does not exist, default value = 50

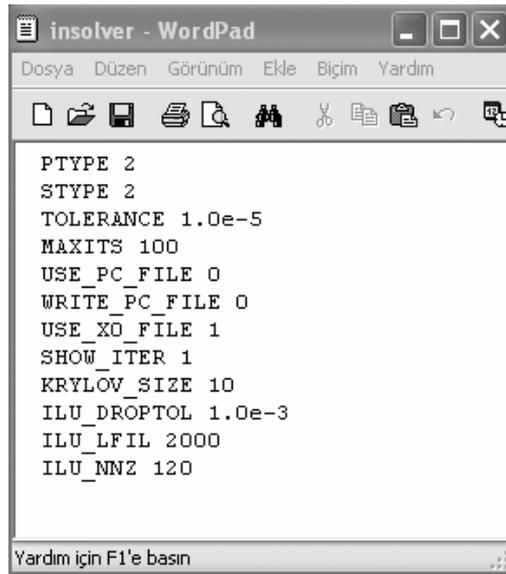


Figure B.26. Matrix Solver: Example file for the optional input file.

The second optional input text file, whose default name is *inpc.txt*, contains the matrices related to the pre-conditioning techniques. This file is used only if the `USE_PC_FILE` parameter is set equal to 1 and the `PTYPE` parameter is different than 0. If this file does not exist in the directory, the program prompts the user to write the name of another file, or to press enter to create the matrices during the execution of the program. The format of this file indeed depends on which pre-conditioning technique is used in the implementation. The contents of this file are listed in Table B.7, Table B.8 and Table B.9 line by line, for the Jacobi and Gauss-Seidel pre-conditioners (if `PTYPE` = 1 and 2), the SSOR and ILU(0) pre-conditioners (if `PTYPE` = 3 and 6), and the ILUT and ILU(k) pre-conditioners (if `PTYPE` = 4 and 5), respectively. In Table B.7 and Table B.8, *AAL*, *JAL* and *IAL* are three matrices related to the lower triangular part of the pre-conditioning matrix stored in the CSR format. Similarly, *AAU*, *JAU* and *IAU* are three matrices related to the upper triangular part of the pre-conditioning matrix stored in the CSR format. In Table B.9, *ALU*, *JLU* and *JU* are three matrices related to the ILU pre-conditioning matrix stored in the Modified Sparse Row (MSR) format. *ALU* and *JLU* contain the *L* and *U* factors together.

Each i^{th} row of the ALU, JLU matrix contains the i^{th} row of L followed by the i^{th} row of U . JU matrix contains the pointers to the beginning of each row of U in the matrix ALU, JLU .

Table B.7. Matrix Solver: Format of the optional file related to the Jacobi and Gauss-Seidel pre-conditioners.

P [1 or 2] (ID no of the pre-conditioning technique)	
nzl (number of the nonzero entries in the lower triangular part of the pre-conditioning matrix)	
dummy line (this line is skipped in the program)	
Real part of 1st component of AAL	Imaginary part of 1st component of AAL
Real part of 2nd component of AAL	Imaginary part of 2nd component of AAL
:	:
Real part of (nzl)th component of AAL	Imaginary part of (nzl)th component of AAL
dummy line (this line is skipped in the program)	
1st component of JAL	
2nd component of JAL	
:	
(nzl)st component of JAL	
dummy line (this line is skipped in the program)	
1st component of IAL	
2nd component of IAL	
:	
(n+1)th component of IAL	

Table B.8. Matrix Solver: Format of the optional file related to the SSOR and ILU(0) pre-conditioners.

P [3 or 6] (ID no of the pre-conditioning technique)	
nzl (number of nonzero entries in the lower triangular part of the pre-conditioning matrix)	
nzu (number of nonzero entries in the upper triangular part of the pre-conditioning matrix)	
dummy line (this line is skipped in the program)	
Real part of 1st component of AAL	Imaginary part of 1st component of AAL
Real part of 2nd component of AAL	Imaginary part of 2nd component of AAL
:	:
Real part of (nzl)th component of AAL	Imaginary part of (nzl)th component of AAL
dummy line (this line is skipped in the program)	

Table B.8. (continued)

1st component of JAL 2nd component of JAL : (nzl)st component of JAL
dummy line (this line is skipped in the program)
1st component of IAL 2nd component of IAL : (n+1)th component of IAL
dummy line (this line is skipped in the program)
Real part of 1st component of AAU Imaginary part of 1st component of AAU Real part of 2nd component of AAU Imaginary part of 2nd component of AAU : Real part of (nzu)th component of AAU Imaginary part of (nzu)th component of AAU
dummy line (this line is skipped in the program)
1st component of JAU 2nd component of JAU : (nzu)st component of JAU
dummy line (this line is skipped in the program)
1st component of IAU 2nd component of IAU : (n+1)th component of IAU

Table B.9. Matrix Solver: Format of the optional file related to the ILUT and ILU(k) pre-conditioners.

P [4 or 5] (ID no of the pre-conditioning technique)
nalu (size of the ALU and JLU matrices)
dummy line (this line is skipped in the program)
Real part of 1st component of ALU Imaginary part of 1st component of ALU Real part of 2nd component of ALU Imaginary part of 2nd component of ALU : Real part of (nalu)th component of ALU Imaginary part of (nalu)th component of ALU
dummy line (this line is skipped in the program)
1st component of JLU 2nd component of JLU : (nalu)st component of JLU
dummy line (this line is skipped in the program)
1st component of JU 2nd component of JU : (n)th component of JU

The third optional input text file, whose default name is *inx0.txt*, contains the initial solution matrix ($X0$). This file is used only if the `USE_X0_FILE` parameter is set equal to 1, otherwise the initial solution matrix is set to all zero matrix. If this file does not exist in the directory, the program prompts the user to write the name of another file or to press enter to set the initial solution matrix to all zero. The contents of this file are listed in Table B.10 line by line.

Table B.10. Matrix Solver: Format of the file related to the solution vector.

Real part of 1 st component of X	Imaginary part of 1 st component of X
Real part of 2 nd component of X	Imaginary part of 2 nd component of X
:	
Real part of (n) th component of X	Imaginary part of (n) th component of X

Output Parameters: This section describes the structure of both the compulsory and the optional output files. At the end of the program, the final solution vector X is written on the compulsory output file whose name is previously described in the command shell (e.g. *sol.txt*). This file must be in the same form in Table B.10.

In addition, the optional output file, whose default name is *outpc.txt*, is created only if the `WRITE_PC_FILE` parameter is set equal to 1, and the `PTYPE` parameter is different than zero. This file also contains the matrices related to the pre-conditioning techniques, and must be in the same form in Table B.7, Table B.8 or Table B.9, depending on the type of the technique.

B.3.2. Direct Matrix Solver

Direct Matrix Solver program has been developed to solve sparse and unsymmetric linear system equations ($AX = B$) using the LU decomposition. The matrix A must be stored in the CSR format. This execution file (*superlu.exe*)

requires a *compulsory* input text file and the name of a *compulsory* output text file. In order to execute the Direct Matrix Solver program, the following command must be written on the command shell:

```
superlu < [compulsory input file name] >[compulsory output file name]
```

```
Example:    > superlu <AB.txt >sol.txt
```

The compulsory input text file containing the entries of the matrix equation must be in the form of Table B.5 as described in Section B.2. At the end of the program, the final solution vector X is written on the compulsory output file whose name is previously described in the command shell (e.g. *sol.txt*). The entries of the solution vector are written in the form of Table B.10 as described in Section B.3.1.

B.4. Post-Processing

FEMS^{3D} Post-Processing program has been developed in the C language to calculate some far-field parameters from the edge field values, which are the solutions of global matrix equation ($AX = B$). The Post-Processing program calculates the following parameters for each θ and ϕ in spherical coordinates, and writes these parameters on separate output files:

- Radar Cross Section (RCS) normalized to λ^2 in dB scale.
- Magnitude and phase of the θ component of electric field (\vec{E}_θ)
- Magnitude and phase of the ϕ component of electric field (\vec{E}_ϕ)

How to Run: The Post-Processing program can be compiled using any C/C++ compiler. After the compilation of all files, the execution file *fem3dpost.exe* is created in the operating directory. This execution file uses three input text files (mesh input file, solution input file and analysis input file), which are *compulsory* to execute the program. While executing the program, an analysis

input file must exist in the operating directory. In order to execute the program, the following command must be written on the command shell.

fem3dpost [compulsory mesh input file name] [compulsory solution input file name]

Example: fem3dpost meshgen.txt sol.txt

Input Parameters: This section describes the structure of the compulsory input files. The first compulsory input text file containing the mesh parameters must be in the form of Table B.2 as described in Section B.1. The second compulsory input text file, which contains the solution (X) of the matrix equation, must be in the form of Table B.10 as defined in Section B.3.1. The third compulsory input text file, whose default name is *analysis.txt*, contains the input parameters related to the post-processing phase of the scattering problem. If this file does not exist in the directory, the program prompts the user to write the name of another file or to press enter to exit the program. The contents of the file *analysis.txt* are listed in Table B.11 line by line, and an example file is illustrated in Figure B.27.

Table B.11. Post-Processing: Format of the input text file for the analysis.

FREQ [value]	Frequency in Hertz
ku [value] [value] [value]	x-y-z components of the unit-direction vector \vec{a}_k
E0inc [value] [value] [value] [value] [value] [value]	Real and imaginary parts of the x-component of the incident plane wave Real and imaginary parts of the y-component of the incident plane wave Real and imaginary parts of the z-component of the incident plane wave
THETA_MIN_MAX_STEP [value] [value] [value]	Minimum, maximum and step values of θ angles at the far-field in spherical coordinates.
PHI_MIN_MAX_STEP [value] [value] [value]	Minimum, maximum and step values of ϕ angles at the far-field in spherical coordinates.

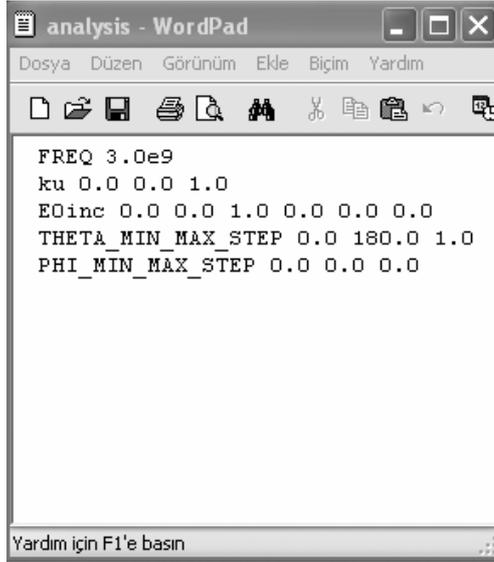


Figure B.27. Post Processing: Example file for the input analysis file.

Output Parameters: At the end of the program, the far-field parameters are written on the following output files:

- RCS normalized to λ^2 in dB scale (*RCS.txt*).
- Magnitude of the θ component of the electric field (\vec{E}_θ) (*Ethetamag.txt*)
- Phase of the θ component of the electric field (\vec{E}_θ) in radian (*Ethetaphase.txt*)
- Magnitude of the ϕ component of the electric field (\vec{E}_ϕ) (*Ephimag.txt*)
- Phase of the ϕ component of the electric field (\vec{E}_ϕ) in radian (*Ephiphase.txt*)

These files are all in the same form, which is shown in Table B.12 where D is a dummy variable indicating one of the five far-field parameters (i.e., RCS, magnitude of \vec{E}_θ , phase of \vec{E}_θ , magnitude of \vec{E}_ϕ or phase of \vec{E}_ϕ).

Table B.12. Post-Processing: Format of the output text files.

$D(\phi_1, \theta_1)$	$D(\phi_1, \theta_2)$	$D(\phi_1, \theta_2)$...	$D(\phi_1, \theta_{last})$
$D(\phi_2, \theta_2)$	$D(\phi_2, \theta_2)$	$D(\phi_2, \theta_2)$...	$D(\phi_2, \theta_{last})$
:				
$D(\phi_{last}, \theta_2)$	$D(\phi_{last}, \theta_2)$	$D(\phi_{last}, \theta_2)$...	$D(\phi_{last}, \theta_{last})$

VITA

PERSONAL INFORMATION

Surname, Name : ÖZGÜN, Özlem
Nationality : Turkish (TC)
Date & Place of Birth : July 08, 1976, Gelibolu

EDUCATION

Degree	Institution	Year of Graduation
MS	Bilkent University Electrical and Electronics Engineering	January 2001
BS	Bilkent University Electrical and Electronics Engineering	June 1998

WORK EXPERIENCE

Year	Place	Enrollment
2004-2005	ASELSAN, Microwave and System Technologies (MST) Division - Radar Electronic Warfare Department	System Engineer
2000-2004	Scientific and Technical Research Council of Turkey – National Electronics and Cryptography Research Institute (TUBITAK-UEKAE) Electronic Warfare Department	Research Scientist
1998-2000	Bilkent University, Department of Electrical and Electronics Engineering	Research Assistant

PUBLICATIONS

1. International Journals

[1] O. Ozgun, M. Kuzuoglu, “Non-Maxwellian locally-conformal PML absorbers for finite element mesh truncation,” IEEE Transactions on Antennas and Propagation, vol. 55, no. 3, pp. 931-937, March 2007.

- [2] O. Ozgun, M. Kuzuoglu, "Locally-conformal Perfectly Matched Layer Implementation for Finite Element Mesh Truncation," *Microwave and Optical Technology Letters*, vol. 48, no. 9, pp. 1836-1839, Sept. 2006.
- [3] O. Ozgun, M. Kuzuoglu, "Multi-center perfectly matched layer implementation for finite element mesh truncation," *Microwave and Optical Technology Letters*, vol. 49, no. 4, pp. 827-832, April 2007.
- [4] O. Ozgun, and M. Kuzuoglu, "Near-field performance analysis of locally-conformal perfectly matched absorbers via Monte Carlo simulations," *Journal of Computational Physics*, (under review).
- [5] O. Ozgun, and M. Kuzuoglu, "Iterative Leap-Field Domain Decomposition Method for Finite Element Solution of Electromagnetic Boundary Value Problems," *IEEE Transactions on Antennas and Propagation*, (under review).
- [6] O. Ozgun, and M. Kuzuoglu, "Finite Element Analysis of Electromagnetic Scattering Problems via Iterative Leap-Field Domain Decomposition Method," *Microwave and Optical Technology Letters*, (under review).
- [7] O. Ozgun, and M. Kuzuoglu, "Forward-Backward Domain Decomposition Method for Finite Element Solution of Boundary Value Problems," *Microwave and Optical Technology Letters*, (under review).
- [8] O. Ozgun, S. Mutlu, M.I. Aksun, L. Alatan, "Design of Dual-Frequency Probe-Fed Microstrip Antennas with Genetic Optimization Algorithm," *IEEE Transactions on Antennas and Propagation*, vol. 51, no: 8, pp. 1947-1954, August 2003.

2. International / National Conferences

- [1] Ö. Özgün, M. Kuzuoğlu, "Sonlu Elemanlar Yöntemi Ağ Sonlandırılmasında Kullanılan Yerel-uyumlu ve Çok-merkezli Tamamen Eşlenmiş Katman Yaklaşımları," *URSI-Türkiye 2006 3. Bilimsel Kongresi*, 6-8 Eylül 2006, Hacettepe Üniv., Ankara.
- [2] O. Ozgun, M. Kuzuoglu, "Locally-conformal and Multi-center Perfectly Matched Layer Implementations for Finite Element Mesh Truncation," *2006 IEEE AP-S International Symposium and USNC/URSI National Radio Science Meeting*, July 9-14 2006, Albuquerque, New Mexico, USA.
- [3] Ö. Özgün, S.G. Tanyer, "Troposferdeki elektromanyetik yayılımın hesaplanmasında Fourier adımlama yönteminin başarımının incelenmesi," *URSI-2002 1. Ulusal Kongresi*, 18-20 Eylül 2002, İTÜ, İstanbul.
- [4] O. Ozgun, S.G. Tanyer, "An examination of the Fourier split-step method of representing electromagnetic propagation in the troposphere," *2002 IEEE International Geoscience and Remote Sensing Symposium and the 24th Canadian Symposium on Remote Sensing, IGARS'2002*, June 24-28 2002, Toronto, Canada.