

# Finite-Volume Simulations of Maxwell's Equations on Unstructured Grids

Ian Jeffrey

PhD Thesis  
Submitted in Partial Fulfillment  
of the Requirements for the Degree of  
Doctor of Philosophy

Department of Electrical and Computer Engineering  
University of Manitoba  
Winnipeg, Manitoba, Canada

Copyright ©2011 by Ian Jeffrey



# Abstract

---

Herein a fully parallel, upwind and flux-split Finite-Volume Time-Domain (FVTD) numerical engine for solving Maxwell's Equations on unstructured grids is developed. The required background theory for solving Maxwell's Equations using FVTD is given in sufficient detail, including a description of both the temporal and spatial approximations used. The details of the local-time stepping strategy of Fumeaux *et al.* is included. A global mesh-truncation scheme using field integration over a Huygens' surface is also presented.

The capabilities of the FVTD algorithm are augmented with thin-wire and subcell circuit models that permit very flexible and accurate simulations of circuit-driven wire structures. Numerical and experimental validation shows that the proposed models have a wide-range of applications. Specifically, it appears that the thin-wire and subcell circuit models may be very well suited to the simulation of radio-frequency coils used in magnetic resonance imaging systems.

A parallelization scheme for the volumetric field solver, combined with the local-time stepping, global mesh-truncation and subcell models is developed that theoretically provides both linear time- and memory scaling in a distributed parallel environment.

Finally, the FVTD code is converted to the frequency domain and the possibility of using different flux-reconstruction schemes to improve the iterative convergence of the Finite-Volume Frequency-Domain algorithm is investigated.

# Contributions

---

This dissertation focuses on the development of a fully parallel, three-dimensional electromagnetic field solver using the Finite-Volume Time-Domain (FVTD) and Finite-Volume Frequency-Domain (FVFD) algorithms. While the application of these algorithms to electromagnetic problems is not new, this work contains significant contributions to FVTD, FVFD and to electromagnetic field-modeling in general.

- A subcell thin-wire model is developed for FVTD that permits the wire to be arbitrarily oriented within the computational domain. While the coupling mechanism between thin-wires and electromagnetic fields is not a contribution of this work, the FVTD solution of the resulting equations, and a thorough explanation of the limitations of the algorithm, is.
- The introduction of a flexible lumped-element circuit model for driving the thin-wire structures allows for complicated circuitry to be introduced into the computational domain without affecting the volumetric mesh. Furthermore, circuit quantities can be used for the straight-forward extraction of engineering system parameters such as input impedance. In the absence of volumetric fields, the thin-wire and subcell models could be used to model multi-conductor transmission lines, though this is not investigated in this work.
- A fully parallel FVTD code has been developed. While parallelization of hyperbolic problems (such as the FVTD formulation of Maxwell's Equations) is not so difficult, this work includes general parallelization strategies for incorporating local-time-stepping, global mesh-truncation and the thin-wire models that permit linear scaling in both time and memory in distributed computing environments.
- A fully parallel FVFD code has been developed. Currently the code only supports volumetric field problems but allows for different flux-reconstruction schemes to be used to accelerate the convergence of the iterative system solver.

# Notation, Symbols and Acronyms

---

*As one learns the language of a subject, one is also learning what the subject is. ...what we call a subject consists mostly, if not entirely, of its language. If you eliminate all the words of a subject, you have eliminated the subject.*  
– Neil Postman

Included in the tables that follow is a summary of the most commonly used acronyms and notation in this work. There is a consistent need to cross-reference most of the derived equations and expressions. For emphasis, the most important equations, resulting from the end of a derivation, are “boxed”. Finally, a comment should be made on the use of the notation  $x \cdot y$ . If  $x = \vec{x}$  and  $y = \vec{y}$  are vectors, then this notation is used to represent a standard dot-product. It is also used, however, to denote scalar, matrix-vector and matrix-matrix products simply because, in some cases, without it equations are not as visually clear. The meaning of the operator should be clear from the context.

Symbols	Description
Volumetric Geometry	
$(x, y, z)$	Cartesian coordinates
$\vec{r}$	Position vector
$V$	Computational domain
$\partial V$	Boundary of $V$

Tab. 0.1: Common symbols and notation (continued next page)

Volumetric Geometry (Cont'd)	
$V_i$	3-dimensional finite-volume in $V$
$\{V_i\}$	Geometric partition of $V$
$\vec{r}_i$	Barycentre of $V_i$
$\partial V_i$	Boundary surface of $V_i$
$\partial V_{i,k}$	Facet $k$ of surface of $\partial V_i$
$\vec{r}_{i,k}$	Barycentre of $\partial V_{i,k}$
$ V_i $	Volume of $V_i$ [ $m^3$ ]
$ \partial V_{i,k} $	Area of $\partial V_{i,k}$ [ $m^2$ ]
$\vec{n}(\vec{r})$	Outer normal to a surface
$\vec{n}_{i,k}$	Unit outward normal to $\partial V_{i,k}$
Electromagnetic Fields	
$t$	Time [s]
$\vec{\mathcal{E}}(t, \vec{r})$	Electric field [V/m]
$\vec{\mathcal{H}}(t, \vec{r})$	Magnetic field [A/m]
$\vec{\mathcal{D}}(t, \vec{r})$	Electric flux density [C/m <sup>2</sup> ]
$\vec{\mathcal{B}}(t, \vec{r})$	Magnetic flux density [Wb/m <sup>2</sup> ]
$\vec{\mathcal{J}}(t, \vec{r})$	Electric current density [A/m <sup>2</sup> ]
$\vec{\mathcal{M}}(t, \vec{r})$	Magnetic current density [V/m <sup>2</sup> ]
$q_e(t, \vec{r})$	Electric charge density [C/m <sup>3</sup> ]
$q_m(t, \vec{r})$	Magnetic charge density [Wb/m <sup>3</sup> ]
$\nabla f(t, \vec{r})$	Gradient of the scalar function $f(t, \vec{r})$
$\nabla \cdot \vec{\mathcal{F}}(t, \vec{r})$	Divergence of the vector function $\vec{\mathcal{F}}(t, \vec{r})$
$\nabla \times \vec{\mathcal{F}}(t, \vec{r})$	Curl of the vector function $\vec{\mathcal{F}}(t, \vec{r})$
$\partial_t$	First partial derivative with respect to $t$
$d_t$	First ordinary derivative with respect to $t$
$\partial_{tt}$	Second partial derivative with respect to $t$
$\epsilon(\vec{r})$	Permittivity at $\vec{r}$ [F/m]
$\mu(\vec{r})$	Permeability at $\vec{r}$ [H/m]
$\sigma_e(\vec{r})$	Electrical conductivity at $\vec{r}$ [S/m]
$\sigma_m(\vec{r})$	Magnetic conductivity at $\vec{r}$ [ $\Omega$ /m]
$c(\vec{r})$	Electromagnetic velocity at $\vec{r}$ [m/s]

Volumetric FVTD	
$\vec{\mathcal{U}}(t, \vec{r})$	Solution vector $[\vec{\mathcal{E}}(t, \vec{r})^T \vec{\mathcal{H}}(t, \vec{r})^T]^T$
$\vec{\mathcal{U}}_{(i)}(t)$	Volumetric average of solution over $V_i$
$\vec{\mathcal{U}}_{i,k}(t, \vec{r})$	Solution on the surface $\partial V_{i,k}$
$\vec{\mathcal{U}}_{i,k}^o$	Boundary value of solution on $\partial V_{i,k}$
$\vec{\mathcal{S}}(t, \vec{r})$	Source vector.
$\vec{\alpha}(\vec{r})$	Constitutive matrix
$\vec{\sigma}(\vec{r})$	Conduction matrix
$\vec{\alpha}_b(\vec{r}), \vec{\sigma}_b(\vec{r})$	Background constitutive matrices
$\vec{L}(\vec{m})$	Rotation matrix for vector $\vec{m}$
$\vec{A}(\epsilon, \mu, \vec{m})$	Flux matrix for vector $\vec{m}$ and parameters $\epsilon$ and $\mu$
$\vec{A}_{i,k}^{\pm}$	Flux-split matrix on surface $\partial V_{i,k}$
$\vec{\mathcal{W}}(t, \vec{r})$	FVTD characteristic vector
$\mathcal{W}_p$	$p$ th component of the characteristic vector
$\vec{\mathcal{F}}(t, \vec{r})$	FVTD flux
$\vec{\mathcal{F}}_{i,k}^o$	Flux on $\partial V_{i,k}$
Time-Stepping and Global Mesh Truncation	
$\Delta t$	Time-step
$\vec{\mathcal{U}}^n$	Solution at time $n\Delta t$
$V^\ell$	Local time-stepping region $\ell$
$\partial V_\Sigma$	Huygen's surface bounded at infinity
$\partial V_S$	Finite Huygen's surface inside $V$
$\vec{r}'$	Integration source point
$\vec{R}$	$\vec{r} - \vec{r}'$
$R$	$ \vec{r} - \vec{r}' $
$[[ \cdot ]]$	Evaluation at the retarded time $t - R/c$

---

Thin-Wire Models	
$(\rho, \phi, \zeta)$	Cylindrical coordinates
$\mathcal{Q}(t, \zeta)$	Per-unit-length (PUL) charge [C/m <sup>2</sup> ]
$\mathcal{I}(t, \zeta)$	Current [A]
$\rho_0$	Field coupling radius
$g(\rho)$	Field coupling function
$L$	PUL inductance [H/m]
$C$	PUL capacitance [F/m]
$G$	PUL conductance [S/m]
$R$	PUL resistance [ $\Omega$ /m]
$Z$	Impedance [ $\Omega$ ]
$Y$	Admittance [S]
$\mathcal{V}(t, \vec{r})$	Voltage [V/m]
$W$	Continuous wire domain
$W_j$	One-dimensional finite-volume
$\zeta_j$	Barycentre of $W_j$
$ W_j $	Length of $W_j$ [m]
$T_{j,k}$	Terminal $k$ of wire $j$
$L_j, C_j$ etc.	PUL parameters associated with $W_j$
$\mathbf{u}(t, \zeta)$	Wire solution $[\mathcal{V}(t, \zeta) \ \mathcal{I}(t, \zeta)]^T$
$\mathbf{u}_{(j)}(t)$	Segment average of solution on $W_j$
$\bar{\mathbf{a}}(k)$	Wire flux matrix

---



<b>Acronym</b>	<b>Meaning</b>
<b>CEM</b>	Computational Electromagnetics
<b>CFD</b>	Computational Fluid Dynamics
<b>DGTD</b>	Discontinuous-Galerkin Time-Domain
<b>ERBC</b>	Exact Radiation Boundary Condition
<b>FDTD</b>	Finite-Difference Time-Domain
<b>FE</b>	Forward Euler
<b>FETD</b>	Finite-Element Time-Domain
<b>FIT</b>	Finite Integration Technique
<b>FV</b>	Finite-Volume
<b>FVFD</b>	Finite-Volume Frequency-Domain
<b>FVTD</b>	Finite-Volume Time-Domain
<b>EM</b>	Electromagnetic
<b>HG</b>	Hypergraph
<b>HPDE</b>	Hyperbolic Partial Differential Equation
<b>IVBP</b>	Initial Boundary Value Problem
<b>LTS</b>	Local Time Stepping
<b>MUSCL</b>	Monotone Upwind Scheme for Conservation Laws
<b>MPI</b>	Message Passing Interface
<b>MRI</b>	Magnetic Resonance Imaging
<b>NEC</b>	Numerical Electromagnetics Code
<b>ORB</b>	Orthogonal Recursive Bisection
<b>PC</b>	Predictor Corrector
<b>PEC</b>	Perfect Electric Conductor
<b>PMC</b>	Perfect Magnetic Conductor
<b>REA</b>	Reconstruct, Evolve, Average
<b>RK2</b>	Runge-Kutta, second order
<b>RK4</b>	Runge-Kutta, fourth order
<b>RF</b>	Radio Frequency
<b>SFC</b>	Space Filling Curve
<b>TLM</b>	Transmission Line Method

*Tab. 0.2:* Common acronyms.

# Acknowledgements

---

*Respect as a personal value implies, in any society, the public acknowledgements of justice and of due honor. – Jacob Bronowski*

I am indebted to Dr. Joe LoVetri who has guided, encouraged and pushed me beyond that which I thought I was capable. An advisor, a mentor, and a friend, Dr. LoVetri has shown me, in immeasurable hours through over a decade, the joys of research and education. I cannot thank you enough.

Thanks to Dr. Vladimir Okhmatovski for your expertise, friendship, and time.

To my friends and colleagues; Dr. Colin Gilmore, Dr. Puyan Mojabi, Amer Zakaria, Dustin Isleifson, Jon Aronsson, Majid Ostadrahimi, Gabriel Faucher, Cameron Kaye and Dr. Dmitry Firsov; I thank you for your support and for countless discussions regarding my work. Specifically, many thanks to Amer and Cam for conducting experimental measurements, Dustin for providing endless feedback on the functionality of the code, Gabriel for your help with mesh generation and Jon for your knowledge of all aspects of parallel computing.

I would like to thank my examining committee: Dr. Greg Bridges, Dr. Shaun Lui, and Dr. Poman So for their evaluation of this work on unfairly short notice.

I wish to extend my thanks to the National Science and Engineering Research Council of Canada, Mathematics of Information Technology and Complex Systems Canada, the University of Manitoba Graduate Students Association, Cadence Design Inc. and Defence Research and Development Canada for funding my research.

Last, to my family and friends... thank you.



# Contents

---

1. <i>Introduction</i> . . . . .	1
1.1 Purpose . . . . .	4
1.2 Outline . . . . .	4
1.3 Scope . . . . .	6
2. <i>The Conservative Form of Maxwell's Equations</i> . . . . .	7
2.1 Maxwell's Equations . . . . .	8
2.2 Constitutive Relationships . . . . .	10
2.3 Continuity Equations . . . . .	12
2.4 The Conservative Form of Maxwell's Equations . . . . .	12
2.5 The Flux Matrix . . . . .	15
2.6 The Similarity Transformation . . . . .	17
2.7 The Characteristic Solution of Maxwell's Equations . . . . .	20
2.8 Chapter Summary . . . . .	21
3. <i>The Method of Lines FVTD Discretization of Maxwell's Equations</i> . . . . .	23
3.1 Discretization of the Computational Domain . . . . .	24
3.2 The Volume-Surface Formulation of the Conservative Equations . . . . .	25
3.3 Godunov's Method for Linear Systems . . . . .	27
3.4 Vector Flux Splitting and Upwinding . . . . .	29
3.5 Variable Coefficient Fluxes and Discontinuous Constitutive Parameters . . . . .	32
3.6 Flux Reconstruction at Material Interfaces . . . . .	34
3.7 Spatial Reconstruction Schemes . . . . .	39
3.8 The Global, Semi-Discrete, FVTD System and Spatial Computational Complexity . . . . .	45
3.9 Chapter Summary . . . . .	47
4. <i>Time-Integration</i> . . . . .	48
4.1 Global Time-Stepping Schemes . . . . .	49
4.2 Time-Stepping Stability . . . . .	53
4.3 Local Time-Stepping . . . . .	54

4.4	Chapter Summary . . . . .	62
5.	<i>Mesh Truncation</i> . . . . .	63
5.1	Local Mesh Truncation: Silver-Müller Boundary Conditions . . . . .	65
5.2	Global Mesh Truncation: Exact Radiating Boundary Conditions . . . . .	66
5.3	Chapter Summary . . . . .	75
6.	<i>Implementation and Validation of the FVTD Volumetric Field Solver</i> . . . . .	76
6.1	Implementation . . . . .	76
6.2	Volumetric Scattering from a PEC Sphere . . . . .	77
6.3	LTS Validation . . . . .	80
6.4	Global Mesh Truncation Validation . . . . .	82
6.5	Chapter Summary . . . . .	87
7.	<i>An FVTD Thin-Wire Subcell Model</i> . . . . .	90
7.1	A Brief History of Time-Domain Thin-Wire Modeling . . . . .	92
7.2	FVTD Thin-Wire Formulation . . . . .	93
7.3	Thin-Wire Boundary Conditions . . . . .	104
7.4	Spatial Reconstruction Schemes . . . . .	110
7.5	Thin-Wire to Field-Solver Coupling . . . . .	113
7.6	Bent Wires, Multi-wire Junctions and Charge Buildup . . . . .	115
7.7	Time Integration . . . . .	117
7.8	Chapter Summary . . . . .	118
8.	<i>FVTD Thin-Wire Boundary Conditions by Subcell Lumped-Circuit Models</i> . . . . .	119
8.1	Circuit Solutions . . . . .	120
8.2	MNA System Construction by Branch Stamps . . . . .	126
8.3	Thin-Wire Subcell Models Terminated at Subcell Circuits . . . . .	130
8.4	Time-Synchronization . . . . .	137
8.5	Chapter Summary . . . . .	138
9.	<i>Subcell Modeling Validation</i> . . . . .	139
9.1	Example 1: A Thin-Wire Dipole . . . . .	139
9.2	Example 2: Straight and Bent Monopole Coupling . . . . .	141
9.3	Example 3: A Thin-Wire Loop Antenna . . . . .	144
9.4	Example 4: A Capacitively Tuned RF Loop for MRI . . . . .	149
9.5	Chapter Summary . . . . .	154
10.	<i>FVTD Parallelization</i> . . . . .	155
10.1	A Brief History of Parallel FVTD . . . . .	156
10.2	Distributed Parallel Environments - Message Passing Interface . . . . .	156
10.3	Volumetric Parallelization Strategy . . . . .	157
10.4	The Parallel Mesh Read and Domain Decomposition . . . . .	160
10.5	The Parallel Volumetric Solution . . . . .	167
10.6	Volumetric Parallelization and Local Time Stepping . . . . .	168

---

10.7 Volumetric Parallelization and Global Mesh Truncation . . . . .	169
10.8 Subcell Model Parallelization . . . . .	172
10.9 Determination of the Volumetric Weights . . . . .	175
10.10 Validation . . . . .	176
10.11 Chapter Summary . . . . .	181
<i>11. Finite-Volume Frequency Domain Solutions of Maxwell's Equations . . . . .</i>	<i>182</i>
11.1 The Global FVFD System . . . . .	184
11.2 Spatial Reconstruction and Electromagnetic Boundary Conditions . . .	185
11.3 Iterative Solutions, Parallelization and (a lack of) Preconditioners . . .	186
11.4 Numerical Results . . . . .	187
11.5 A Novel Iterative Approach . . . . .	190
11.6 Chapter Summary . . . . .	192
<i>12. Conclusions and Future Improvements . . . . .</i>	<i>193</i>
12.1 Summary of Accomplishments . . . . .	193
12.2 Future Work: Modeling Capabilities . . . . .	195
12.3 Future Work: Algorithmic Considerations . . . . .	197
 <i>Appendix</i>	 199
<i>A. Derivation Details for the Conservative Form of Maxwell's Equations . . . .</i>	<i>200</i>
<i>B. Time Domain Integral Solutions To Maxwell's Equations . . . . .</i>	<i>208</i>
B.1 The Vector and Scalar Potentials . . . . .	208
B.2 The Kirchoff Solution of the Scalar Wave Equation . . . . .	213
B.3 Time-Domain Integral Field Solutions . . . . .	217

# List of Tables

---

0.1	Notation and Symbols . . . . .	iii
0.2	Common Acronyms . . . . .	vii
8.1	Constitutive Equations for Common Circuit Branches . . . . .	121
8.2	Branch Stamps for some Common Linear Circuit Elements . . . . .	128

# List of Figures

---

3.1	Boundary Limits of the Solution at a Cell-Interface . . . . .	31
3.2	Characteristic Reconstruction of States at a Cell Interface . . . . .	33
3.3	Computation of the MUSCL Gradient . . . . .	43
3.4	Element Stencils . . . . .	45
4.1	General depiction of a global partitioning of $V$ into LTS partitions $V^\ell$ .	55
4.2	LTS Classification of an Element Based on its MUSCL Stencil . . . . .	56
4.3	Upper and Lower Elements in a 3-level LTS Partition of a 2D Mesh . .	58
4.4	LTS Synchronization between two levels for the Predictor-Corrector Scheme . . . . .	59
5.1	Illustration of the Silver-Müller Boundary Conditions . . . . .	64
5.2	The General Geometry of a Huygens' Surface Geometry . . . . .	67
5.3	Geometrical Configuration for Incorporating a Huygens' Surface into FVTD . . . . .	68
5.4	A Global Mesh Truncation Integration Surface . . . . .	70
5.5	Time Interpolation in the Global Mesh Truncation Scheme . . . . .	72



---

5.6	Problem Size Reduction Using Global Mesh Truncation . . . . .	74
6.1	Gaussian Sources for Volumetric PEC Scattering . . . . .	78
6.2	FVTD Solution to Scattering from a PEC Sphere - Coarse Mesh . . . . .	79
6.3	FVTD Solution to Scattering from a PEC Sphere - Fine Mesh . . . . .	80
6.4	(Colour) Volumetric Time-Domain Scattered Field Visualization . . . . .	81
6.5	(Colour) LTS Levels of an Unoptimized Mesh . . . . .	83
6.6	Comparison of LTS and non-LTS solutions for the time-domain scattered field . . . . .	84
6.7	(Colour) Sample Geometry for Validation of the Global Mesh Truncation Scheme . . . . .	86
6.8	(Colour) Propagation of Scattered Electric Fields Using Global Mesh Truncation . . . . .	88
6.9	Analytic Comparison of Scattered Fields using Global Mesh Truncation . . . . .	89
7.1	Geometric Description of the Thin-Wire Model . . . . .	94
7.2	The Wire-Field Coupling Function $g(\rho)$ . . . . .	96
7.3	Thin-Wire to Volume Coupling Influence . . . . .	116
8.1	A Current Driven RC Circuit . . . . .	123
8.2	A Voltage-Driven RLC Circuit . . . . .	124
8.3	A Circuit-Driven Thin-Wire Loop Antenna . . . . .	131
8.4	Interfacing of a Single Thin-Wire to Circuit . . . . .	133
8.5	Multiple Thin-Wire Segments Terminating at a Circuit . . . . .	134
9.1	Thin-Wire Dipole Geometry . . . . .	140

---

9.2	Thin-Wire Dipole Simulation Results . . . . .	142
9.3	(Colour) Visualization of Simulated Dipole Fields . . . . .	143
9.4	(Colour) Geometry of Simulated Straight and L-Shaped Antennas . . .	145
9.5	(Colour) Simulation and Measurement Results: Coupling Between Two Straight Monopoles . . . . .	146
9.6	(Colour) Simulation and Measurement Results: Coupling Between Two L-Shaped Antennas . . . . .	147
9.7	Thin-Wire Loop Input Impedance . . . . .	150
9.8	(Colour) Thin-Wire Loop Field Visualization . . . . .	151
9.9	Geometry of a Capacitively Tuned MRI Thin-Wire Loop . . . . .	152
9.10	Simulation Results for an MRI-tuned RF Loop . . . . .	153
10.1	Continuous Partitioning of the Global Mesh and Halo Element Definitions	159
10.2	Visual description of the parallel mesh read. . . . .	163
10.3	An ORB Domain Decomposition. . . . .	165
10.4	A Visualization of Halo Layers on a Single Processor . . . . .	167
10.5	Visualization of the Parallelization of the Global Mesh-Truncation Scheme	171
10.6	Parallel Partitions Including Global Mesh-Truncation . . . . .	172
10.7	Visualization of the Subcell-Model Parallelization Strategy . . . . .	174
10.8	Parallel Efficiency Scaling of the Volumetric Solver . . . . .	177
10.9	Parallel Efficiency Scaling of the LTS scheme . . . . .	178
10.10	Parallel Efficiency Scaling of the Global Mesh-Truncation Scheme . . .	179
10.11	Parallel Efficiency Scaling of the Subcell Models . . . . .	180

11.1 FVFD Convergence for Different Flux Schemes. . . . .	188
11.2 FVFD Convergence for a Hybrid Flux Scheme . . . . .	189
11.3 (Colour) Visualization of the FVFD Solution for PEC Scattering. . . . .	191

# Introduction

---

*A science only advances with certainty, when the plan of inquiry and the object of our researches have been clearly defined.* – Jean-Baptiste Say

In 1873 James Clerk Maxwell published *A Treatise on Electricity & Magnetism* [1], providing a set of four partial differential equations coupling electric charge, electric current and both electric and magnetic fields in macroscopic media. The concise simplicity of Maxwell's equations has not limited their ability to predict electromagnetic phenomena. Perhaps the best example of the power of Maxwell's equations was demonstrated by Maxwell himself before publishing his treatise. In 1865, Maxwell used his equations to show that electric and magnetic fields satisfy the electromagnetic wave equation thereby predicting that light must be an electromagnetic wave [2]. While Maxwell's equations provide the theoretical framework for describing the behaviour of electromagnetic fields and their sources, it is our limited ability to analytically describe

the fields satisfying Maxwell's equations that often make obtaining exact, closed-form solutions to problems of interest impossible. Fortunately, the advent of the modern-day computer in the 1940s made it possible to approximately solve Maxwell's equations numerically. The ability of Maxwell's equations to predict electromagnetic behaviour, combined with an exponential increase in computational power over the last seven decades, has forever changed the field of electrical engineering. The fact that any design, no matter how complicated, can *theoretically* be validated before it is constructed justifiably makes numerical solutions to Maxwell's equations a large, and continually growing, field of research known as *computational electromagnetics* (CEM) [3].

Historically, robust algorithms for numerically solving Maxwell's equations did not begin appearing in publication until the 1960s, nearly two decades after the advent of the computer. It has been argued that the cause for this delay is that Maxwell's equations are linear, and that during the 1940s analytical progress in electromagnetics was still being made [3]. In contrast, scientists concerned with the dynamic behaviour of fluids were dealing with the Euler and Navier-Stokes equations [4]. The non-linearity of these equations limits the number of analytical tools that can be used to solve them, making numerical solutions immediately attractive. As a consequence, the field of *computational fluid dynamics* (CFD) was born nearly two decades before CEM [3].

Today, numerical techniques for solving the time-dependent Maxwell's equations include the *Finite-Difference Time-Domain* method (FDTD) [5], *Finite-Element Time-Domain* methods (FETD) [6], the *Transmission-Line Method* (TLM) [7], *Finite Integration Techniques* (FIT) [8] and more recently, the *Finite-Volume Time-Domain* (FVTD) [9] and *Discontinuous Galerkin Time-Domain* (DGTD) methods [10]. Each of these methods has its own advantages and disadvantages. For example, FDTD is incredibly efficient as its implementation (primarily) depends on a regular Cartesian

---

discretization of the computational domain, but suffers from geometric modeling error introduced by “stair-stepping” approximations in the Cartesian grid. FETD offers more geometry flexibility by allowing unstructured meshes, at the additional cost of a sparse matrix inversion at each time-step. FVTD also permits unstructured grids and does not require a matrix inversion but, in its simplest implementations, is limited to second order where the errors appear as numerical dissipation [11]. DGTD is the newest of these algorithms and attempts to balance the benefits of FVTD and FETD [10].

Herein, we focus on the FVTD algorithm. FVTD has its origins in CFD where it has been used, for decades, to solve the physics of fluid-flow [11, 4]. It is only within the last 20 odd years that FVTD has been applied to electromagnetics problems. In that time, FVTD has been implemented for structured, conformal meshes [12] and unstructured meshes [9, 13]. When used in unstructured meshes it provides excellent geometric modeling capabilities at the sacrifice of increased memory usage and computational time per step when compared to FDTD (for the same number of grid cells). Even with the burden of increased memory and reduced efficiency, FVTD can be used to accurately model highly intricate geometric features that would be incredibly difficult to model using FDTD [14, 15, 13].

In this thesis we deal exclusively with what is referred to as the “upwind” and “flux-split” FVTD formulation on unstructured grids [11, 9]. In this formulation, all electric and magnetic field components are collocated in time within each grid cell. As will be shown in this work, such a formulation permits great flexibility in introducing other numerical models into a volumetric FVTD code. As briefly mentioned above, the price of upwinding is that the errors introduced into the solution are dissipative. This is in contrast to the dispersive errors introduced by FDTD. With FVTD, fields traveling

large electrical distances can be highly dampened in amplitude (but produce excellent phase reconstruction) if the volumetric grid is not sufficiently refined. Alternative FVTD formulations that introduce dispersive, rather than dissipative, errors have recently been attempted and use a leap-frog update where the electric and magnetic fields are offset in time [16] (and the references therein). Unfortunately, this method is not yet robust enough on unstructured three-dimensional grids as it can suffer from local instabilities and grid-decoupling [17].

### *1.1 Purpose*

The purpose of this thesis is to provide the details of an upwind and flux-split FVTD numerical engine for solving Maxwell's Equations on unstructured grids. In addition to the basic volumetric formulation which has been previously developed in, for example, [9] or [14], very flexible thin-wire and subcell circuit models have been developed and implemented to expand the electromagnetic modeling capabilities of FVTD. In addition, as parallel computing becomes more and more accessible, this work provides a detailed parallelization strategy for running FVTD in a distributed computing environment. Finally, the FVTD algorithm is also converted to the frequency domain and a novel strategy for using different flux reconstruction schemes to improve the iterative convergence of the Finite-Volume Frequency-Domain (FVFD) solver is investigated.

### *1.2 Outline*

Although it hasn't been done explicitly, this thesis could be broken into two parts: the first being work required to develop an FVTD engine to the point where novel

---

advancement can be made, and the second being the improvements/additional features made to existing FVTD formulations. Herein, the first part is contained in Chapters 2 through 6. In Chapters 2 and 3 the necessary theory is developed to arrive at a semi-discrete FVTD formulation where continuous temporal derivatives remain. Chapter 4 provides details of explicit time-integration algorithms and a local-time-stepping (LTS) strategy previously developed in [15]. The LTS scheme has been included in this work as it provides significant performance increases on unstructured meshes. In Chapter 5, a local mesh-truncation scheme and a global mesh-truncation scheme are discussed. Finally, the FVTD implementation is validated in Chapter 6.

The second part of this thesis is contained in the remaining chapters and deals, in order, with subcell models, parallelization and the Finite-Volume Frequency-Domain (FVFD) method. In Chapters 7 to 9 an FVTD thin-wire and subcell circuit model are introduced and validated. The thin-wire model was originally developed for FDTD and FETD [18, 19], but is novel in the sense that it has been completely formulated for FVTD. The subcell-circuit models are used to drive and/or terminate the thin-wires and are both incredibly versatile and completely new. In Chapter 10, a parallelization strategy is presented, implemented and validated for the FVTD numerical engine. While parallelization of hyperbolic systems is relatively straight-forward for purely volumetric problems, the parallelization herein also manages the LTS, global mesh-truncation and subcell modeling schemes. Finally, in Chapter 11, the necessary steps required to convert the FVTD code to a frequency-domain FVFD solver are discussed. The focus therein is on the flexibility that upwinding provides and how different flux-reconstruction schemes can be used, in some cases, to improve the iterative convergence of the solution.



### 1.3 Scope

The scope of the work presented herein is limited to the upwind and flux-split FVTD formulation. While high-order reconstruction schemes are mentioned for completeness, a second-order MUSCL scheme is used exclusively in volumetric simulations. For this flavour of FVTD, it is unreasonable to simulate geometries that are many wavelengths in size and the validating examples presented adhere to moderate electrical sizes.

The dissipative nature of the upwind and flux-split scheme is not the focus of this work. As will be demonstrated, a large number of examples can be solved accurately despite this constraint. Additionally, high-order flux-reconstruction schemes can be used, at the cost of complexity, to overcome these dissipative effects [20].

Furthermore, outside of asymptotic limits, little focus is given to the memory and time required to simulate the examples. The code is a work in progress and improvements can always be made. Some details are provided at the end of Chapter 10. Nevertheless, the majority of the results that will be presented were obtained in minutes, not hours, on a moderate number of processors. A comparison of the computational efficiencies of FVTD and FIT can be found, for example, in [21] and a comparison to FDTD is available in [14].

The LTS scheme is presented in Chapter 4 with only slight differences to the founding work [15]. The FVTD global mesh-truncation scheme discussed in Chapter 5 was first published for FVTD by colleagues in [22] and is presented and implemented without modifications. It is computationally expensive and a rigorous analysis of this scheme is beyond the scope of this work. It has been included because it is both quite interesting and worth future consideration.

# The Conservative Form of Maxwell's Equations

---

*Ex nihilo nihil fit* – Lucretius, Translation: *From nothing, nothing comes.*

The Finite-Volume (FV) method is a numerical technique for solving systems of hyperbolic partial differential equations (HPDEs) and can, therefore, be applied to the class of HPDEs commonly referred to as conservation laws. A common approach for developing FV solutions to EM problems is to manipulate Maxwell's Equations into a *conservative form*. In this chapter Maxwell's Equations are presented; assumptions about the class of physical problems, including the description of the physical media, are made; the conservative form of Maxwell's Equations is developed, and is shown to be hyperbolic; and the solution to the system of equations when field variations occur along a single spatial direction is discussed. The developed theory serves as the

groundwork for the development of the FV solution to Maxwell's Equations developed in Chapter 3.

## 2.1 Maxwell's Equations

In SI units Maxwell's Equations relate six electromagnetic field quantities at time  $t$  and position  $\vec{r}$  in three-dimensional space  $\mathbb{R}^3$ :

$$\begin{aligned} \vec{D}(t, \vec{r}) & \text{ Electric Flux Density [C/m}^2\text{]} \in \mathbb{R}^3 \\ \vec{B}(t, \vec{r}) & \text{ Magnetic Flux Density [Wb/m}^2\text{]} \in \mathbb{R}^3 \\ \vec{E}(t, \vec{r}) & \text{ Electric Field [V/m]} \in \mathbb{R}^3 \\ \vec{H}(t, \vec{r}) & \text{ Magnetic Field [A/m]} \in \mathbb{R}^3 \\ \vec{J}(t, \vec{r}) & \text{ Electric Current Density [A/m}^2\text{]} \in \mathbb{R}^3 \\ q_e(t, \vec{r}) & \text{ Electric Charge Density [C/m}^3\text{]} \in \mathbb{R} \end{aligned}$$

Although the existence of magnetic charge has not been established, it is convenient to introduce two additional magnetic sources that can be used in equivalent mathematical models:

$$\begin{aligned} \vec{M}(t, \vec{r}) & \text{ Magnetic Current Density [V/m}^2\text{]} \in \mathbb{R}^3 \\ q_m(t, \vec{r}) & \text{ Magnetic Charge Density [Wb/m}^3\text{]} \in \mathbb{R} \end{aligned}$$

The introduction of magnetic current density and magnetic charge density completes a nice duality between electric and magnetic field quantities. We allow for impressed currents by decomposing the total current into the sum of impressed (superscript  $i$ )

and conduction (superscript  $c$ ) contributions:

$$\begin{aligned}\vec{\mathcal{J}}(t, \vec{r}) &= \vec{\mathcal{J}}^c(t, \vec{r}) + \vec{\mathcal{J}}^i(t, \vec{r}) \\ \vec{\mathcal{M}}(t, \vec{r}) &= \vec{\mathcal{M}}^c(t, \vec{r}) + \vec{\mathcal{M}}^i(t, \vec{r})\end{aligned}$$

where the impressed currents are to be interpreted as the sources of the electromagnetic fields (note that convection current is not considered herein). Similarly, we can introduce impressed charges  $q_e^i(t, \vec{r})$  and  $q_m^i(t, \vec{r})$  and conduction charges  $q_e^c(t, \vec{r})$ ,  $q_m^c(t, \vec{r})$ .

Wherever the electromagnetic fields are continuous and sufficiently differentiable they satisfy Maxwell's Equations:

$$\nabla \times \vec{\mathcal{E}}(t, \vec{r}) = -\partial_t \vec{\mathcal{B}}(t, \vec{r}) - \vec{\mathcal{M}}^c(t, \vec{r}) - \vec{\mathcal{M}}^i(t, \vec{r}) \quad (2.1a)$$

$$\nabla \times \vec{\mathcal{H}}(t, \vec{r}) = \partial_t \vec{\mathcal{D}}(t, \vec{r}) + \vec{\mathcal{J}}^c(t, \vec{r}) + \vec{\mathcal{J}}^i(t, \vec{r}) \quad (2.1b)$$

$$\nabla \cdot \vec{\mathcal{D}}(t, \vec{r}) = q_e^c(t, \vec{r}) + q_e^i(t, \vec{r}) \quad (2.1c)$$

$$\nabla \cdot \vec{\mathcal{B}}(t, \vec{r}) = q_m^c(t, \vec{r}) + q_m^i(t, \vec{r}) \quad (2.1d)$$

which are respectively referred to as *Faraday's Law*, *Ampère's Law*, *Gauss's Law* and *Gauss's Law of Magnetism*.

As the solution to a typical electromagnetic problem inside a volume  $V \subset \mathbb{R}^3$  we seek to determine the fields  $\vec{\mathcal{E}}(t, \vec{r})$ ,  $\vec{\mathcal{H}}(t, \vec{r})$ , the fluxes  $\vec{\mathcal{D}}(t, \vec{r})$ ,  $\vec{\mathcal{B}}(t, \vec{r})$ , the conduction currents,  $\vec{\mathcal{J}}^c(t, \vec{r})$ ,  $\vec{\mathcal{M}}^c(t, \vec{r})$ , and the conduction charges  $q_e^c(t, \vec{r})$  and  $q_m^c(t, \vec{r})$  resulting from the impressed sources  $\vec{\mathcal{J}}^i(t, \vec{r})$ ,  $\vec{\mathcal{M}}^i(t, \vec{r})$ ,  $q_e^i(t, \vec{r})$  and  $q_m^i(t, \vec{r})$ . This problem can be formally stated as an initial boundary-value problem (IBVP) where we seek the solution to Maxwell's equations in  $V$ , bounded by  $\partial V$ , over a time interval  $\mathcal{T} = (0, t_{\max}] \subset \mathbb{R}^+$  given initial conditions  $\vec{\mathcal{E}}(0, \vec{r})$ ,  $\vec{\mathcal{H}}(0, \vec{r})$ ,  $\vec{\mathcal{D}}(0, \vec{r})$ ,  $\vec{\mathcal{B}}(0, \vec{r})$ ,  $\vec{\mathcal{J}}^c(0, \vec{r})$ ,  $\vec{\mathcal{M}}^c(0, \vec{r})$ ,  $q_e^c(0, \vec{r})$ ,  $q_m^c(0, \vec{r}) \forall \vec{r} \in V$ . The impressed sources are assumed known  $\forall t \in \mathcal{T}, \vec{r} \in V$ . The

IBVP description is completed by providing appropriate boundary conditions on  $\partial V$  and relates 20 unknown scalar quantities (the components of 6 vector quantities and two scalars) through 8 scalar equations (two vector and two scalar equations).

## 2.2 Constitutive Relationships

Additional information is required in order to solve Maxwell's Equations (2.1a)-(2.1d).

Twelve additional equations are obtained by the so-called constitutive relationships:

$$\vec{\mathcal{D}}(t, \vec{r}) = \vec{\mathcal{D}}(t, \vec{r}, \vec{\mathcal{E}}, \vec{\mathcal{H}}) \quad (2.2a)$$

$$\vec{\mathcal{B}}(t, \vec{r}) = \vec{\mathcal{B}}(t, \vec{r}, \vec{\mathcal{E}}, \vec{\mathcal{H}}) \quad (2.2b)$$

$$\vec{\mathcal{J}}^c(t, \vec{r}) = \vec{\mathcal{J}}^c(t, \vec{r}, \vec{\mathcal{E}}, \vec{\mathcal{H}}) \quad (2.2c)$$

$$\vec{\mathcal{M}}^c(t, \vec{r}) = \vec{\mathcal{M}}^c(t, \vec{r}, \vec{\mathcal{E}}, \vec{\mathcal{H}}) \quad (2.2d)$$

For simple, stationary media, these relationships can be written as

$$\vec{\mathcal{D}}(t, \vec{r}) = \bar{\bar{\epsilon}}(\vec{r}) \cdot \vec{\mathcal{E}}(t, \vec{r}) \quad (2.3a)$$

$$\vec{\mathcal{B}}(t, \vec{r}) = \bar{\bar{\mu}}(\vec{r}) \cdot \vec{\mathcal{H}}(t, \vec{r}) \quad (2.3b)$$

$$\vec{\mathcal{J}}^c(t, \vec{r}) = \bar{\bar{\sigma}}_e(\vec{r}) \cdot \vec{\mathcal{E}}(t, \vec{r}) \quad (2.3c)$$

$$\vec{\mathcal{M}}^c(t, \vec{r}) = \bar{\bar{\sigma}}_m(\vec{r}) \cdot \vec{\mathcal{H}}(t, \vec{r}) \quad (2.3d)$$

The quantities  $\bar{\bar{\epsilon}}(\vec{r})$ ,  $\bar{\bar{\mu}}(\vec{r})$ ,  $\bar{\bar{\sigma}}_e(\vec{r})$  and  $\bar{\bar{\sigma}}_m(\vec{r}) \in \mathbb{R}^{3 \times 3}$  are respectively referred to as the permittivity dyadic, permeability dyadic, electric conductivity dyadic and magnetic conductivity dyadic and are determined by the physical medium in  $V$  at  $\vec{r}$ . In this work we limit ourselves to simple, isotropic, time-invariant material such that the

dyadics are scalar multiples of the three-dimensional identity matrix  $\bar{\bar{I}}$

$$\bar{\bar{\epsilon}}(\vec{r}) = \epsilon(\vec{r}) \cdot \bar{\bar{I}} \quad (2.4a)$$

$$\bar{\bar{\mu}}(\vec{r}) = \mu(\vec{r}) \cdot \bar{\bar{I}} \quad (2.4b)$$

$$\bar{\bar{\sigma}}_e(\vec{r}) = \sigma_e(\vec{r}) \cdot \bar{\bar{I}} \quad (2.4c)$$

$$\bar{\bar{\sigma}}_m(\vec{r}) = \sigma_m(\vec{r}) \cdot \bar{\bar{I}} \quad (2.4d)$$

with units  $\epsilon(\vec{r})$  [F/m],  $\mu(\vec{r})$  [H/m],  $\sigma_e(\vec{r})$  [S/m] and  $\sigma_m(\vec{r})$  [ $\Omega$ /m]. These twelve constitutive equations can be substituted into Maxwell's Equations to give:

$$\nabla \times \vec{\mathcal{E}}(t, \vec{r}) = -\mu(\vec{r}) \cdot \partial_t \vec{\mathcal{H}}(t, \vec{r}) - \sigma_m(\vec{r}) \cdot \vec{\mathcal{H}}(t, \vec{r}) - \vec{\mathcal{M}}^i(t, \vec{r}) \quad (2.5a)$$

$$\nabla \times \vec{\mathcal{H}}(t, \vec{r}) = \epsilon(\vec{r}) \cdot \partial_t \vec{\mathcal{E}}(t, \vec{r}) + \sigma_e(\vec{r}) \cdot \vec{\mathcal{E}}(t, \vec{r}) + \vec{\mathcal{J}}^i(t, \vec{r}) \quad (2.5b)$$

$$\nabla \cdot \left( \epsilon(\vec{r}) \cdot \vec{\mathcal{E}}(t, \vec{r}) \right) = q_e^c(t, \vec{r}) + q_e^i(t, \vec{r}) \quad (2.5c)$$

$$\nabla \cdot \left( \mu(\vec{r}) \cdot \vec{\mathcal{H}}(t, \vec{r}) \right) = q_m^c(t, \vec{r}) + q_m^i(t, \vec{r}) \quad (2.5d)$$

In this form, Maxwell's Equations represent 8 equations relating 8 unknown quantities  $\vec{\mathcal{E}}(t, \vec{r})$ ,  $\vec{\mathcal{H}}(t, \vec{r})$ ,  $q_e^c(t, \vec{r})$  and  $q_m^c(t, \vec{r})$ . What is interesting is that if we seek the solution to the vector fields in the absence of imposed charge, then equations (2.5a), (2.5b), combined with initial and boundary conditions, are sufficient as they are six scalar equations for six scalar functions.

### 2.3 Continuity Equations

If we take the divergence of the two curl equations (2.5a) and (2.5b) we obtain:

$$\nabla \cdot \left( \mu(\vec{r}) \cdot \partial_t \vec{\mathcal{H}}(t, \vec{r}) \right) = -\nabla \cdot \left( \sigma_m \cdot \vec{\mathcal{H}}(t, \vec{r}) + \vec{\mathcal{M}}^i(t, \vec{r}) \right) \quad (2.6a)$$

$$\nabla \cdot \left( \epsilon(\vec{r}) \cdot \partial_t \vec{\mathcal{E}}(t, \vec{r}) \right) = -\nabla \cdot \left( \sigma_e \cdot \vec{\mathcal{E}}(t, \vec{r}) + \vec{\mathcal{J}}^i(t, \vec{r}) \right) \quad (2.6b)$$

Interchanging the temporal and spatial derivatives and substituting Gauss's Laws into the above equations gives:

$$\partial_t \left( q_m^c(t, \vec{r}) + q_m^i(t, \vec{r}) \right) + \nabla \cdot \left( \sigma_m \cdot \vec{\mathcal{H}}(t, \vec{r}) + \vec{\mathcal{M}}^i(t, \vec{r}) \right) = 0 \quad (2.7a)$$

$$\partial_t \left( q_e^c(t, \vec{r}) + q_e^i(t, \vec{r}) \right) + \nabla \cdot \left( \sigma_e \cdot \vec{\mathcal{E}}(t, \vec{r}) + \vec{\mathcal{J}}^i(t, \vec{r}) \right) = 0 \quad (2.7b)$$

which are respectively referred to as the *electric* and *magnetic equations of continuity*. These equations state that electric/magnetic charge is conserved and can be interpreted as *physical laws*. What is important here is that the continuity equations above put strict limitations on the relationship between total currents and charge at  $\vec{r} \in V$ . Provided that these conditions are enforced Maxwell's Equations can be solved by considering only the two curl equations (2.5a) and (2.5b).

### 2.4 The Conservative Form of Maxwell's Equations

Using notation similar to that in [9], we obtain the conservative form of Maxwell's Equations by combining the time-domain differential form of Ampère's and Faraday's

laws into a single system of six partial differential equations:

$$\boxed{\bar{\alpha}(\vec{r}) \cdot \partial_t \vec{\mathcal{U}}(t, \vec{r}) + \vec{\mathcal{K}}(t, \vec{r}) = \bar{\sigma}(\vec{r}) \cdot \vec{\mathcal{U}}(t, \vec{r}) + \vec{\mathcal{S}}(t, \vec{r})}, \quad (2.8)$$

where

$$\vec{\mathcal{U}}(t, \vec{r}) = \begin{bmatrix} \vec{\mathcal{E}}(t, \vec{r}) \\ \vec{\mathcal{H}}(t, \vec{r}) \end{bmatrix}, \quad \vec{\mathcal{K}}(t, \vec{r}) = \begin{bmatrix} -\nabla \times \mathcal{H}(t, \vec{r}) \\ \nabla \times \mathcal{E}(t, \vec{r}) \end{bmatrix}, \quad \vec{\mathcal{S}}(t, \vec{r}) = \begin{bmatrix} -\vec{\mathcal{J}}^i(t, \vec{r}) \\ -\vec{\mathcal{M}}^i(t, \vec{r}) \end{bmatrix} \quad (2.9)$$

The simple, stationary, and isotropic physical medium is accounted for by the  $6 \times 6$  diagonal matrices  $\bar{\alpha}(\vec{r})$  and  $\bar{\sigma}(\vec{r})$ :

$$\begin{aligned} \bar{\alpha}(\vec{r}) &= \text{diag}(\epsilon(\vec{r}), \epsilon(\vec{r}), \epsilon(\vec{r}), \mu(\vec{r}), \mu(\vec{r}), \mu(\vec{r})) \\ \bar{\sigma}(\vec{r}) &= \text{diag}(-\sigma_e(\vec{r}), -\sigma_e(\vec{r}), -\sigma_e(\vec{r}), -\sigma_m(\vec{r}), -\sigma_m(\vec{r}), -\sigma_m(\vec{r})) \end{aligned} \quad (2.10)$$

If a scattered-field formulation is desired, we decompose the total fields  $\vec{\mathcal{U}}(t, \vec{r})$  into the sum of a scattered field  $\vec{\mathcal{U}}^{scat}(t, \vec{r})$  and an incident field  $\vec{\mathcal{U}}^{inc}(t, \vec{r})$ . The incident field is assumed to be supported by the impressed sources  $\vec{\mathcal{S}}(t, \vec{r})$  in some background medium  $\bar{\alpha}_b(\vec{r})$  and  $\bar{\sigma}_b(\vec{r})$  and, from (2.8), satisfy

$$\bar{\alpha}_b(\vec{r}) \cdot \partial_t \vec{\mathcal{U}}(t, \vec{r}) + \vec{\mathcal{K}}^{inc}(t, \vec{r}) = \bar{\sigma}_b(\vec{r}) \cdot \vec{\mathcal{U}}^{inc}(t, \vec{r}) + \vec{\mathcal{S}}(t, \vec{r}) \quad (2.11)$$

Subtracting (2.11) from (2.8) and simplifying produces:

$$\bar{\alpha}(\vec{r}) \cdot \partial_t \vec{\mathcal{U}}^{scat}(t, \vec{r}) + \vec{\mathcal{K}}^{scat}(t, \vec{r}) = \bar{\sigma}(\vec{r}) \cdot \vec{\mathcal{U}}^{scat}(t, \vec{r}) + \vec{\mathcal{S}}^{scat}(t, \vec{r}) \quad (2.12)$$



where the scattered-field source term is defined as:

$$\vec{\mathcal{S}}^{scat}(t, \vec{r}) \triangleq (\bar{\vec{\sigma}}(\vec{r}) - \bar{\vec{\sigma}}_b(\vec{r})) \cdot \vec{\mathcal{U}}(t, \vec{r}) - (\bar{\vec{\alpha}}(\vec{r}) - \bar{\vec{\alpha}}_b(\vec{r})) \cdot \partial_t \vec{\mathcal{U}}^{inc}(t, \vec{r}) \quad (2.13)$$

A comparison of (2.12) and (2.8) shows that both the total- and scattered-fields satisfy the same system of partial differential equations, but with different source terms. Any theory developed for one, holds for the other. (As will be shown in Chapter 3 this is not quite true as part of the FVTD algorithm requires an incident-field-dependent flux-term on perfect conducting surfaces.)

We now show that equation (2.8) is in fact a conservation law for the quantity  $\vec{\mathcal{U}}(t, \vec{r})$ . The vector  $\vec{\mathcal{K}}(t, \vec{r})$  is a function of the components of  $\vec{\mathcal{U}}(t, \vec{r})$ , and can be written in Cartesian coordinates  $(x, y, z)$  as:

$$\vec{\mathcal{K}}(t, \vec{r}) = \partial_x \begin{bmatrix} 0 \\ \mathcal{H}_z \\ -\mathcal{H}_y \\ 0 \\ -\mathcal{E}_z \\ \mathcal{E}_y \end{bmatrix} + \partial_y \begin{bmatrix} -\mathcal{H}_z \\ 0 \\ \mathcal{H}_x \\ \mathcal{E}_z \\ 0 \\ -\mathcal{E}_x \end{bmatrix} + \partial_z \begin{bmatrix} \mathcal{H}_y \\ -\mathcal{H}_x \\ 0 \\ -\mathcal{E}_y \\ \mathcal{E}_x \\ 0 \end{bmatrix} = \partial_x \vec{\mathcal{G}}_x(t, \vec{r}) + \partial_y \vec{\mathcal{G}}_y(t, \vec{r}) + \partial_z \vec{\mathcal{G}}_z(t, \vec{r}) \quad (2.14)$$

where the explicit dependence of the field components on space and time has been dropped for brevity. As a consequence of (2.14), if we assume that there is no loss or sources at  $\vec{r}$ , (2.8) becomes

$$\begin{aligned} \partial_t \left( \bar{\vec{\alpha}}(\vec{r}) \cdot \vec{\mathcal{U}}(t, \vec{r}) \right) + \partial_x \vec{\mathcal{G}}_x(t, \vec{r}) + \partial_y \vec{\mathcal{G}}_y(t, \vec{r}) + \partial_z \vec{\mathcal{G}}_z(t, \vec{r}) = \\ \partial_t \left( \bar{\vec{\alpha}}(\vec{r}) \cdot \vec{\mathcal{U}}(t, \vec{r}) \right) + \nabla \cdot \vec{\mathcal{G}}(t, \vec{r}) = \vec{0} \end{aligned} \quad (2.15)$$

The  $6 \times 3$  matrix  $\vec{\mathcal{G}}(t, \vec{r})$  is composed of three columns  $\vec{\mathcal{G}}_x(t, \vec{r})$ ,  $\vec{\mathcal{G}}_y(t, \vec{r})$  and  $\vec{\mathcal{G}}_z(t, \vec{r})$ . The divergence is applied to each row of  $\vec{\mathcal{G}}(t, \vec{r})$  producing a vector in  $\mathbb{R}^6$  [13].

For the FV algorithm developed in Chapter 3, the material parameters  $\bar{\alpha}(\vec{r}) = \bar{\alpha}$  will be assumed piece-wise constant. Thus, we currently assume that the medium is constant for all  $\vec{r}$  under consideration and (2.15) becomes:

$$\partial_t \vec{\mathcal{U}}(t, \vec{r}) + \nabla \cdot \left( \bar{\alpha}^{-1} \cdot \vec{\mathcal{G}}(t, \vec{r}) \right) = \vec{\mathbf{0}} \quad (2.16)$$

Although we have not yet shown it, this system is hyperbolic and is interpreted as the conservative form of Maxwell's Equations (in the source-free/lossless case) analogous to a scalar conservation law *e.g.* (2.7a), (2.7b).

## 2.5 The Flux Matrix

To formulate the FV solution to the conservative form of Maxwell's two curl equations (2.16), we will require some additional concepts and notation. As shown in Appendix A, it is possible to rewrite (2.16) as

$$\partial_t \vec{\mathcal{U}}(t, \vec{r}) + \sum_{k=1}^K \left( \bar{\mathbf{A}}(\epsilon, \mu, \vec{m}_k) \cdot \partial_{\vec{m}_k} \vec{\mathcal{U}}(t, \vec{r}) \right) = \vec{\mathbf{0}} \quad (2.17)$$

where  $\{\vec{m}_k\}$ ,  $k = 1, \dots, K$  is a set of  $K$  unit vectors (possibly dependent) that span  $\mathbb{R}^3$  and where

$$\partial_{\vec{m}_k} \vec{\mathcal{U}}(t, \vec{r}) \triangleq \vec{m}_k \cdot \partial_{\vec{r}} \vec{\mathcal{U}}(t, \vec{r})$$

is the spatial rate of change of  $\vec{\mathcal{U}}(t, \vec{r})$  in the  $\vec{m}_k$  direction. The validity of this result is far from obvious and rigorous analysis is required to develop it [9]. The  $6 \times 6$  matrix

$\bar{\bar{\mathbf{A}}}(\epsilon, \mu, \vec{m}_k)$  is given by:

$$\bar{\bar{\mathbf{A}}}(\epsilon, \mu, \vec{m}) \triangleq \bar{\bar{\boldsymbol{\alpha}}}^{-1} \cdot \begin{bmatrix} 0 & 0 & 0 & 0 & m_z & -m_y \\ 0 & 0 & 0 & -m_z & 0 & -m_x \\ 0 & 0 & 0 & m_y & m_x & 0 \\ 0 & -m_z & m_y & 0 & 0 & 0 \\ m_z & 0 & -m_x & 0 & 0 & 0 \\ -m_y & m_x & 0 & 0 & 0 & 0 \end{bmatrix} = \bar{\bar{\boldsymbol{\alpha}}}^{-1} \cdot \bar{\bar{\mathbf{L}}}(\vec{m}) \quad (2.18)$$

Notational simplification can be obtained by considering the three-by-three antisymmetric matrix:

$$\bar{\bar{\mathbf{s}}}(\vec{m}) = \begin{bmatrix} 0 & -m_z & m_y \\ m_z & 0 & -m_x \\ -m_y & m_x & 0 \end{bmatrix} \quad (2.19)$$

The result of the matrix-vector-product  $\bar{\bar{\mathbf{s}}}(\vec{m}) \cdot \vec{x} = \vec{m} \times \vec{x}$  for any vector  $\vec{x}$ . The matrix  $\bar{\bar{\mathbf{L}}}(\vec{m})$  can then be written as:

$$\bar{\bar{\mathbf{L}}}(\vec{m}) = \begin{bmatrix} \bar{\bar{\mathbf{0}}} & -\bar{\bar{\mathbf{s}}}(\vec{m}) \\ \bar{\bar{\mathbf{s}}}(\vec{m}) & \bar{\bar{\mathbf{0}}} \end{bmatrix} \quad (2.20)$$

where  $\bar{\bar{\mathbf{0}}}$  is a  $3 \times 3$  matrix of zeros. Therefore,

$$\bar{\bar{\mathbf{L}}}(\vec{m}) \cdot \vec{\mathcal{U}}(t, \vec{r}) = \begin{bmatrix} -\vec{m} \times \vec{\mathcal{H}}(t, \vec{r}) \\ \vec{m} \times \vec{\mathcal{E}}(t, \vec{r}) \end{bmatrix}$$

For reasons that will become clear shortly, the matrix  $\bar{\bar{\mathbf{A}}}(\epsilon, \mu, \vec{m}) = \bar{\bar{\boldsymbol{\alpha}}}^{-1} \cdot \bar{\bar{\mathbf{L}}}(\vec{m})$  is referred to as the *flux matrix* in the  $\vec{m}$  direction. The flux matrix can be compactly

written as:

$$\bar{\bar{\mathbf{A}}}(\epsilon, \mu, \vec{m}) = \begin{bmatrix} \bar{\bar{\mathbf{0}}} & -\epsilon^{-1} \cdot \bar{\bar{s}}(\vec{m}) \\ \mu^{-1} \cdot \bar{\bar{s}}(\vec{m}) & \bar{\bar{\mathbf{0}}} \end{bmatrix} \quad (2.21)$$

## 2.6 The Similarity Transformation

The concept of “flux-splitting” is often used for developing “upwind” FV solutions to hyperbolic systems [11, 9]. Although we will not define flux-splitting and upwinding until Chapter 3, the foundational work required for introducing an upwind and flux-split FVTD algorithm is presented in this section.

We make the assumption that the only variation in the solution is in the  $\vec{m} \in \{\vec{m}_k\}$  direction. This would hold, for example, in the case of plane-wave propagation. Under this assumption the conservative form of Maxwell’s Equations expanded in the basis  $\{\vec{m}_k\}$  reduces to a one-dimensional system of partial-differential-equations:

$$\partial_t \vec{\mathbf{u}}(t, \vec{r}) + \bar{\bar{\mathbf{A}}}(\epsilon, \mu, \vec{m}) \cdot \partial_m \vec{\mathbf{u}}(t, \vec{r}) = \vec{\mathbf{0}} \quad (2.22)$$

Equation (2.22) shows that the matrix  $\bar{\bar{\mathbf{A}}}(\epsilon, \mu, \vec{m})$  is aptly referred to as the flux matrix: if it is taken inside the spatial derivative we can interpret its product with  $\vec{\mathbf{u}}(t, \vec{r})$  as the flux in the  $\vec{m}$  direction. Not surprisingly, the flux-matrix is linear for the linear Maxwell’s Equations. We now seek a direct solution to the one-dimensional system (2.22). We know that if the system is hyperbolic, a solution is easily obtained by the method of characteristics [11]. To prove that the system is hyperbolic it suffices to show that the flux matrix is diagonalizable by means of a similarity transformation

with real eigenvalues [11]. Therefore we seek a transformation of the form:

$$\bar{\mathbf{A}}(\epsilon, \mu, \vec{m}) = \bar{\mathbf{P}}(\epsilon, \mu, \vec{m}) \cdot \bar{\mathbf{\Lambda}}(\epsilon, \mu, \vec{m}) \cdot \bar{\mathbf{P}}^{-1}(\epsilon, \mu, \vec{m}) \quad (2.23)$$

where  $\bar{\mathbf{P}}(\epsilon, \mu, \vec{m})$  is the  $6 \times 6$  matrix of right eigenvectors,  $\bar{\mathbf{P}}^{-1}(\epsilon, \mu, \vec{m})$  is the matrix of left eigenvectors and the diagonal matrix  $\bar{\mathbf{\Lambda}}(\epsilon, \mu, \vec{m})$  contains the eigenvalues. The matrices satisfying the transformation (2.23) are provided below. The diagonal matrix of eigenvalues is:

$$\begin{aligned} \bar{\mathbf{\Lambda}}(\epsilon, \mu, \vec{m}) &= \bar{\mathbf{\Lambda}}(\epsilon, \mu) = \text{diag} \left( 0, 0, (\epsilon\mu)^{-1/2}, (\epsilon\mu)^{-1/2}, -(\epsilon\mu)^{-1/2}, -(\epsilon\mu)^{-1/2} \right) \\ &= \text{diag} (0, 0, c, c, -c, -c) = \text{diag} (\Lambda_0, \Lambda_1, \Lambda_2, \Lambda_3, \Lambda_4, \Lambda_5) \end{aligned} \quad (2.24)$$

where  $c = \sqrt{\epsilon\mu}^{-1}$  is the electromagnetic speed in the region with constant material parameters  $\epsilon$  and  $\mu$  and where  $\Lambda_p$  denotes the  $p^{\text{th}}$  eigenvalue. The eigenvector matrices are:

$$\bar{\mathbf{P}}(\epsilon, \mu, \vec{m}) = \begin{bmatrix} m_x & 0 & m_x m_z Z & -m_x m_y Z & -m_x m_z Z & m_x m_y Z \\ m_y & 0 & m_y m_z Z & (m_x^2 + m_y^2) Z & -m_y m_z Z & -(m_x^2 + m_y^2) Z \\ m_z & 0 & -(m_x^2 + m_y^2) Z & -m_y m_x Z & (m_x^2 + m_y^2) Z & m_y m_z Z \\ 0 & m_x & -m_y & -m_z & -m_y & -m_z \\ 0 & m_y & m_x & 0 & m_x & 0 \\ 0 & m_z & 0 & m_x & 0 & m_x \end{bmatrix} \quad (2.25)$$

$$\bar{\bar{\mathbf{P}}}^{-1}(\epsilon, \mu, \vec{m}) = \frac{1}{2} \begin{bmatrix} 2m_x & 2m_y & 2m_z & 0 & 0 & 0 \\ 0 & 0 & 0 & 2m_x & 2m_y & 2m_z \\ m_x^{-1}m_z Y & 0 & -Y & -m_y & m_x^{-1}(m_x^2 + m_z^2) & -m_x^{-1}m_y m_z \\ -m_x^{-1}m_y Y & Y & 0 & -m_z & -m_x^{-1}m_y m_z & m_x^{-1}(m_x^2 + m_y^2) \\ -m_x^{-1}m_z Y & 0 & Y & -m_y & m_x^{-1}(m_x^2 + m_z^2) & -m_x^{-1}m_y m_z \\ m_x^{-1}m_y Y & -Y & 0 & -m_z & -m_x^{-1}m_y m_z & m_x^{-1}(m_x^2 + m_y^2) \end{bmatrix} \quad (2.26)$$

where  $Z = \sqrt{\mu/\epsilon} [\Omega]$  is the impedance of the medium and  $Y = Z^{-1} [S]$  is the admittance. Clearly the eigenvalues of the similarity transformation are real and the conservative form of Maxwell's Equations is hyperbolic. We denote the six columns of the matrix  $\bar{\bar{\mathbf{P}}}(\epsilon, \mu, \vec{m})$  by  $\bar{\mathbf{r}}_p \in \mathbb{R}^{6 \times 1}$   $p = 0, \dots, 5$  and the rows of  $\bar{\bar{\mathbf{P}}}^{-1}(\epsilon, \mu, \vec{m})$  by  $\bar{\mathbf{l}}_p \in \mathbb{R}^{1 \times 6}$  such that

$$\bar{\bar{\mathbf{P}}} = \begin{bmatrix} \bar{\mathbf{r}}_0 & \bar{\mathbf{r}}_1 & \bar{\mathbf{r}}_2 & \bar{\mathbf{r}}_3 & \bar{\mathbf{r}}_4 & \bar{\mathbf{r}}_5 \end{bmatrix} \quad \bar{\bar{\mathbf{P}}}^{-1} = \begin{bmatrix} \bar{\mathbf{l}}_0^T & \bar{\mathbf{l}}_1^T & \bar{\mathbf{l}}_2^T & \bar{\mathbf{l}}_3^T & \bar{\mathbf{l}}_4^T & \bar{\mathbf{l}}_5^T \end{bmatrix}^T \quad (2.27)$$

Consider  $\bar{\bar{\mathbf{P}}}^{-1} \cdot \bar{\bar{\mathbf{A}}} = \bar{\bar{\mathbf{P}}}^{-1} \cdot \bar{\bar{\mathbf{P}}} \cdot \bar{\bar{\mathbf{\Lambda}}} \cdot \bar{\bar{\mathbf{P}}}^{-1}$ . Omitting the physical and spatial dependence of the diagonalization procedure for brevity, we have

$$\begin{bmatrix} \bar{\mathbf{l}}_0 \\ \bar{\mathbf{l}}_1 \\ \bar{\mathbf{l}}_2 \\ \bar{\mathbf{l}}_3 \\ \bar{\mathbf{l}}_4 \\ \bar{\mathbf{l}}_5 \end{bmatrix} \cdot \bar{\bar{\mathbf{A}}} = \bar{\bar{\mathbf{\Lambda}}} \cdot \begin{bmatrix} \bar{\mathbf{l}}_0 \\ \bar{\mathbf{l}}_1 \\ \bar{\mathbf{l}}_2 \\ \bar{\mathbf{l}}_3 \\ \bar{\mathbf{l}}_4 \\ \bar{\mathbf{l}}_5 \end{bmatrix} = \begin{bmatrix} \Lambda_0 \bar{\mathbf{l}}_0 \\ \Lambda_1 \bar{\mathbf{l}}_1 \\ \Lambda_2 \bar{\mathbf{l}}_2 \\ \Lambda_3 \bar{\mathbf{l}}_3 \\ \Lambda_4 \bar{\mathbf{l}}_4 \\ \Lambda_5 \bar{\mathbf{l}}_5 \end{bmatrix} \quad (2.28)$$

Thus we see that due to the diagonal nature of  $\bar{\bar{\mathbf{\Lambda}}}$  the rows of  $\bar{\bar{\mathbf{P}}}^{-1}$  are the left eigen-

vectors of the flux matrix. In a similar manner, we consider the equality  $\bar{\bar{\mathbf{A}}} \cdot \bar{\bar{\mathbf{P}}} = \bar{\bar{\mathbf{P}}} \cdot \bar{\bar{\mathbf{\Lambda}}} \cdot \bar{\bar{\mathbf{P}}}^{-1} \cdot \bar{\bar{\mathbf{P}}}$  leading to:

$$\begin{aligned} \bar{\bar{\mathbf{A}}} \cdot \begin{bmatrix} \bar{r}_0 & \bar{r}_1 & \bar{r}_2 & \bar{r}_3 & \bar{r}_4 & \bar{r}_5 \end{bmatrix} &= \begin{bmatrix} \bar{r}_0 & \bar{r}_1 & \bar{r}_2 & \bar{r}_3 & \bar{r}_4 & \bar{r}_5 \end{bmatrix} \cdot \bar{\bar{\mathbf{\Lambda}}} \\ &= \begin{bmatrix} \Lambda_0 \bar{r}_0 & \Lambda_1 \bar{r}_1 & \Lambda_2 \bar{r}_2 & \Lambda_3 \bar{r}_3 & \Lambda_4 \bar{r}_4 & \Lambda_5 \bar{r}_5 \end{bmatrix} \end{aligned} \quad (2.29)$$

demonstrating that the columns of  $\bar{\bar{\mathbf{P}}}$  are the right eigenvectors of the flux matrix.

## 2.7 The Characteristic Solution of Maxwell's Equations

The system (2.22) can be solved by multiplying both sides by  $\bar{\bar{\mathbf{P}}}^{-1}(\epsilon, \mu, \vec{m})$ :

$$\begin{aligned} \bar{\bar{\mathbf{P}}}^{-1} \partial_t \vec{\mathbf{U}}(t, \vec{r}) + \bar{\bar{\mathbf{\Lambda}}} \cdot \bar{\bar{\mathbf{P}}}^{-1} \cdot \partial_m \vec{\mathbf{U}}(t, \vec{r}) &= \\ \partial_t \left( \bar{\bar{\mathbf{P}}}^{-1} \cdot \vec{\mathbf{U}}(t, \vec{r}) \right) + \bar{\bar{\mathbf{\Lambda}}} \cdot \partial_m \left( \bar{\bar{\mathbf{P}}}^{-1} \cdot \vec{\mathbf{U}}(t, \vec{r}) \right) &= \\ \partial_t \vec{\mathbf{W}}(t, \vec{r}) + \bar{\bar{\mathbf{\Lambda}}} \cdot \partial_m \vec{\mathbf{W}}(t, \vec{r}) &= \vec{\mathbf{0}} \end{aligned} \quad (2.30)$$

The interchange of derivatives and multiplication by  $\bar{\bar{\mathbf{P}}}^{-1}(\epsilon, \mu, \vec{m})$  is legitimized by the assumption of constant material parameters in a neighbourhood of  $\vec{r}$ . The quantity:

$$\vec{\mathbf{W}}(t, \vec{r}) \triangleq \bar{\bar{\mathbf{P}}}^{-1} \cdot \vec{\mathbf{U}} = \begin{bmatrix} \mathcal{W}_0(t, \vec{r}) & \mathcal{W}_1(t, \vec{r}) & \mathcal{W}_2(t, \vec{r}) & \mathcal{W}_3(t, \vec{r}) & \mathcal{W}_4(t, \vec{r}) & \mathcal{W}_5(t, \vec{r}) \end{bmatrix}^T \quad (2.31)$$

satisfies an uncoupled set of six partial differential equations and is referred to as the *characteristic vector* of the conservative form of Maxwell's Equations [11]. Each component of the characteristic vector is referred to as a *characteristic variable*. In terms of the eigenvectors of the previous section the  $p^{th}$  characteristic variable can be

written as:

$$\mathcal{W}_p(t, \vec{r}) = \bar{\ell}_p \cdot \vec{\mathcal{U}}(t, \vec{r}) \quad (2.32)$$

The last form of (2.30) is referred to as the *system of characteristic equations* consisting of six scalar partial differential equations of the form:

$$\partial_t \mathcal{W}_p(t, \vec{r}) + \Lambda_p \partial_m \mathcal{W}_p(t, \vec{r}). \quad (2.33)$$

Each of these equations is a one-dimensional advection equation in the coordinate  $\vec{m} \cdot \vec{r}$ .

The solution to this equation is easily obtained as:

$$\mathcal{W}_p(t, \vec{r}) = \mathcal{W}_p(0, (\vec{m} \cdot \vec{r}) - \Lambda_p t) \quad (2.34)$$

According to equation (2.34) the characteristic variables propagate unchanged along the  $p^{\text{th}}$  *characteristic curve*  $(\vec{m} \cdot \vec{r}) - \Lambda_p t = \text{constant}$  with velocity  $\Lambda_p$ . The fundamental principle that, for hyperbolic systems, characteristic variables propagate unchanged along characteristic curves will be used in the development of a stable FV solution to Maxwell's Equations in the next chapter.

## 2.8 Chapter Summary

In this chapter the analytic ground-work required to develop the FV solution to Maxwell's Equations has been laid out in some detail. Maxwell's Equations were presented; the medium was assumed to be simple, stationary and isotropic; the conservative form of Maxwell's Equations was developed; and the characteristic solution to Maxwell's Equations was introduced for the case where the only field variation is



along a single spatial dimension.

# The Method of Lines FVTD Discretization of Maxwell's Equations

---

*“Young man, in mathematics you don’t understand things. You just get used to them.”* – John von Neumann in response to Felix T. Smith who had said *“I’m afraid I don’t understand the method of characteristics.”*

In this chapter the method of lines FVTD discretization of the electromagnetic initial boundary-value problem (IBVP) in a computational domain  $V$  bounded by  $\partial V$  is developed using an upwind, flux-split finite-volume discretization of the spatial operators. Numerical approximation of the temporal derivative is discussed in Chapter 4. Furthermore, we assume that the computational domain  $V$  is either enclosed by ideal conducting surfaces or that  $\partial V$  recedes to infinity. Methods for truncating the computational domain are provided in Chapter 5.

### 3.1 Discretization of the Computational Domain

To solve the IBVP in a computational domain  $V$ , the domain is partitioned into a set  $\{V_i\}$  of  $N_V$  finite-volumes, or cells,  $i = 1, \dots, N_V$  each with an associated volume  $|V_i|$  and boundary  $\partial V_i$ . The outer unit normal to  $V_i$  on  $\partial V_i$  is  $\vec{n}_i(\vec{r}), \vec{r} \in \partial V_i$ . A fundamental assumption of the partition  $\{V_i\}$  is that the physical constitutive parameters in each  $V_i$  are constant:  $\epsilon(\vec{r}) = \epsilon_i$ ,  $\mu(\vec{r}) = \mu_i$  and  $\sigma(\vec{r}) = \sigma_i$ ,  $\forall \vec{r} \in V_i$ . Corresponding to these constant physical parameters, the material matrices  $\bar{\bar{\alpha}}(\vec{r})$  and  $\bar{\bar{\sigma}}(\vec{r})$  must also be constant in the  $i^{\text{th}}$  volume and are denoted as  $\bar{\bar{\alpha}}_i$  and  $\bar{\bar{\sigma}}_i$ . For a scattered-field formulation the background matrices must also be constant in each  $V_i$ , and are denoted  $\bar{\bar{\alpha}}_{b_i}$  and  $\bar{\bar{\sigma}}_{b_i}$ .

Beyond the assumption of constant material parameters for each cell,  $\{V_i\}$  can be structured or unstructured and the cells can have arbitrary geometric order. In the case of first-order polyhedral elements, the partition can consist of a mixture of any first-order elements including hexahedrons, tetrahedrons and prisms.

For the remainder of this work we consider a partition consisting of first-order polyhedral elements each with barycentre  $\vec{r}_i$ . Each polyhedral volume is bounded by a set of  $K_i$  flat facets symbolized by  $\partial V_{i,k}, k = 1, \dots, K_i$ , each having a surface area  $|\partial V_{i,k}|$  and a constant outward unit-normal  $\vec{n}_{i,k}$ . The barycentres of each facet are symbolized by  $\vec{r}_{i,k}$ .

### 3.2 The Volume-Surface Formulation of the Conservative Equations

According to the results of the previous chapter we can write the conservative form of Maxwell's Equations in terms of a set of vectors  $\{\vec{m}_k\}$ ,  $k = 1, \dots, K$  as in (2.17).

When sources and loss are present this equation becomes:

$$\partial_t \vec{\mathbf{u}}(t, \vec{r}) + \sum_{k=1}^K \left( \bar{\bar{\mathbf{A}}}(\epsilon, \mu, \vec{m}_k) \cdot \partial_{m_k} \vec{\mathbf{u}}(t, \vec{r}) \right) = \bar{\alpha}^{-1}(\vec{r}) \cdot \left( \bar{\sigma}(\vec{r}) \cdot \vec{\mathbf{u}}(t, \vec{r}) + \vec{\mathcal{S}}(t, \vec{r}) \right) \quad (3.1)$$

We consider this equation in a single first-order polyhedral cell  $V_i$  and let the basis  $\{\vec{m}_k\}$  be the set of outward unit normals to the cell facets  $\{\vec{n}_{i,k}\}$ ,  $k = 1, \dots, K_i$ . As the material parameters  $\bar{\alpha}(\vec{r})$  and  $\bar{\sigma}(\vec{r})$  are constants over  $V_i$  we define the abbreviated notation  $\bar{\bar{\mathbf{A}}}_{i,k} \triangleq \bar{\bar{\mathbf{A}}}(\epsilon_i, \mu_i, \vec{n}_{i,k})$ . In this case the system of equations (3.1) becomes

$$\partial_t \vec{\mathbf{u}}(t, \vec{r}) + \sum_{k=1}^{K_i} \left( \bar{\bar{\mathbf{A}}}_{i,k} \cdot \partial_{n_{i,k}} \vec{\mathbf{u}}(t, \vec{r}) \right) = \bar{\alpha}_i^{-1} \cdot \left( \bar{\sigma}_i \cdot \vec{\mathbf{u}}(t, \vec{r}) + \vec{\mathcal{S}}(t, \vec{r}) \right), \quad \vec{r} \in V_i \quad (3.2)$$

We denote the volumetric average of  $\vec{\mathbf{u}}$  and  $\vec{\mathcal{S}}$  over a finite-volume using a single subscript in parentheses. In this way  $\vec{\mathbf{u}}_{(i)}(t)$  denotes the volumetric average of the solution over  $V_i$ . We define the cell-average residual  $\vec{\mathcal{R}}_{(i)}(t)$  of equation (3.2) when  $\vec{r} \in V_i$  as [10]:

$$\begin{aligned} \vec{\mathcal{R}}_{(i)}(t) &\triangleq d_t \vec{\mathbf{u}}_{(i)}(t) \\ &+ \frac{1}{|V_i|} \iiint_{V_i} \sum_{k=1}^{K_i} \left( \bar{\bar{\mathbf{A}}}_{i,k} \cdot \partial_{n_{i,k}} \vec{\mathbf{u}}(t, \vec{r}) \right) dV - \bar{\alpha}_i^{-1} \cdot \left( \bar{\sigma}_i \cdot \vec{\mathbf{u}}_{(i)}(t) + \vec{\mathcal{S}}_{(i)}(t) \right) \end{aligned} \quad (3.3)$$

where the partial derivative  $\partial_t$  becomes an ordinary derivative  $d_t$  operating on the cell-average solution. Assuming sufficient differentiability of the solution, the volume integral of the spatial derivatives of  $\vec{\mathbf{u}}(t, \vec{r})$  is converted to a surface integral by the divergence theorem. While the result of this operation may not be obvious given the

form (3.3), it can be easily shown by direct integration of (2.8) that it results in:

$$\vec{\mathcal{R}}_{(i)}(t) = d_t \vec{\mathbf{u}}_{(i)}(t) + \frac{1}{|V_i|} \sum_{k=1}^{K_i} \iint_{\partial V_{i,k}} \bar{\mathbf{A}}_{i,k} \cdot \vec{\mathbf{u}}^o(t, \vec{r}) dS + \bar{\alpha}_i^{-1} \cdot \left( \bar{\sigma}_i \cdot \vec{\mathbf{u}}_{(i)}(t) - \vec{\mathcal{S}}_{(i)}(t) \right) \quad (3.4)$$

where  $\vec{\mathbf{u}}^o(t, \vec{r})$ ,  $\vec{r} \in \partial V_{i,k}$  is the boundary limit of the solution. In the FVTD algorithm we demand that the residual on each cell goes to zero resulting in the local system of equations on each cell:

$$d_t \vec{\mathbf{u}}_{(i)}(t) = - \frac{1}{|V_i|} \sum_{k=1}^{K_i} \iint_{\partial V_{i,k}} \bar{\mathbf{A}}_{i,k} \cdot \vec{\mathbf{u}}^o(t, \vec{r}) dS + \bar{\alpha}_i^{-1} \cdot \left( \bar{\sigma}_i \cdot \vec{\mathbf{u}}_{(i)}(t) + \vec{\mathcal{S}}_{(i)}(t) \right) \quad (3.5)$$

Some comments regarding the local system of equations (3.5) are warranted and pertain to the interpretation of the equation on  $V_i$ . In one regard, it is tempting to state that this system of equations is *exact* in the sense that no approximations have been made beyond the discretization of the computational domain. Therefore, if  $\{V_i\}$  models  $V$  and its constitutive parameters exactly, Maxwell's Equations in  $V$  are modeled precisely by (3.5) [15]. On the other hand, the application of the divergence theorem requires continuity constraints on the solution  $\vec{\mathbf{u}}(t, \vec{r})$  on  $V_i$  and its boundary. This is somewhat bothersome as discontinuities in the fields result from discontinuities in the constitutive parameters and therefore exist exactly at cell boundaries [23].

An alternative interpretation of (3.5) can be made by assuming that we have a local solution  $\left( \vec{\mathbf{u}}(t, \vec{r}) \right)_i$  (not to be confused with the volumetric average) in each  $V_i$  and that this solution is defined in  $V_i$  and on its boundary  $\partial V_i$  [10]. This reasoning permits two solutions on the boundary  $\partial V_{i,k}$  of each first-order polyhedral element: one from  $V_i$  and one from  $V_j$  with  $\partial V_{j,k'} = \partial V_{i,k}$ . The global solution can only be recovered by

ensuring the appropriate relationship between these two solutions, that is, by properly determining  $\vec{\mathbf{u}}^o(t, \vec{r})$  on  $\partial V_{i,k}$  that numerically satisfies Maxwell's Equations.

### 3.3 Godunov's Method for Linear Systems

There are many ways to globally solve (3.5). The method of Godunov, originally proposed for solving Euler's non-linear gas dynamics equations has "revolutionized the field of computational fluid dynamics, by overcoming many of the difficulties that had plagued earlier numerical methods for compressible flow" [11].

The basic idea behind Godunov's method is to associate with each  $V_i$  the cell-average of the solution. That is, for the numerical solution of (3.7) we let  $\vec{\mathbf{u}}_{(i)}(t)$  be the unknown quantities. With this in mind Godunov's method for solving the conservative form of Maxwell's Equations can be summarized by a three-step procedure referred to as the *Reconstruct, Evolve, Average*, or REA algorithm:

1. *Reconstruct* the global solution  $\vec{\mathbf{u}}(t, \vec{r})$ ,  $\vec{r} \in V$  from the cell-average values  $\vec{\mathbf{u}}_{(i)}(t)$ .
2. *Evolve* the global solution  $\vec{\mathbf{u}}(t, \vec{r})$  over a time interval  $\Delta t$  by evaluating the boundary flux integrals and integrating the time derivative.
3. *Average* the global solution  $\vec{\mathbf{u}}(t, \vec{r})$  to obtain the new local volumetric average  $\vec{\mathbf{u}}_{(i)}(t + \Delta t)$  in each cell.

In this chapter we are concerned with Step 1 (the reconstruction step), that is, how to compute  $\vec{\mathbf{u}}_{i,k}^o(t, \vec{r}) = \vec{\mathbf{u}}^o(t, \vec{r})$  for  $\vec{r} \in \partial V_{i,k}$ . We will have reason to consistently use the subscript pair notation  $\vec{\mathbf{u}}_{i,k}(t, \vec{r})$ , for any quantity, to indicate that  $\vec{r} \in \partial V_{i,k}$ . Before solving this problem, we must choose the flavour of the FV method to be used, as

there are many [11]. As mentioned briefly in Chapter 2, we use a flux-split, upwind formulation. At this point it important to note the following:

- **Note:** The volumetric averages  $\vec{\mathbf{u}}_{(i)}(t)$  and  $\vec{\mathbf{S}}_{(i)}(t)$  will only be computed (known) to some desired spatial and temporal accuracy. Henceforth, we assume that these quantities are numerical approximations to the true values.

For notational simplicity in what follows we symbolically denote the flux of the solution  $\vec{\mathcal{F}}_{i,k}^o$  over the partial boundary  $\partial V_{i,k}$  as:

$$\vec{\mathcal{F}}_{i,k}^o(t, \vec{r}) = \vec{\mathcal{F}}_{i,k}^o(\epsilon_i, \mu_i, \vec{n}_{i,k}, \vec{\mathbf{u}}^o(t, \vec{r})) \triangleq \bar{\bar{\mathbf{A}}}_{i,k} \cdot \vec{\mathbf{u}}^o(t, \vec{r}), \quad \vec{r} \in \partial V_{i,k} \quad (3.6)$$

such that (3.5) can be written as:

$$d_t \vec{\mathbf{u}}_{(i)}(t) = - \frac{1}{|V_i|} \sum_{k=1}^{K_i} \iint_{\partial V_{i,k}} \vec{\mathcal{F}}_{i,k}^o(t, \vec{r}) dS + \bar{\bar{\alpha}}_i^{-1} \cdot \left( \bar{\bar{\sigma}}_i \cdot \vec{\mathbf{u}}_{(i)}(t) + \vec{\mathbf{S}}_{(i)}(t) \right), \quad \vec{r} \in \partial V_{i,k} \quad (3.7)$$

- **Note:** According to Godunov's method, the flux  $\vec{\mathcal{F}}_{i,k}^o(t, \vec{r})$  is to be reconstructed from the *approximate* cell-averages  $\vec{\mathbf{u}}_{(i)}(t)$ . Due to this assumption, and upwinding approximations made in the next section, it is common to refer to this flux as a *numerical* flux function [11].

Given a reconstruction of the boundary flux, the surface integrals can be evaluated to the desired precision using numerical quadrature reducing (3.7) to a function of time only.

### 3.4 Vector Flux Splitting and Upwinding

A highly-versatile Finite-Volume method can be formulated by using the so-called flux-split, upwind schemes originally introduced by Van Leer [24], and now ubiquitous in Finite-Volume solutions for hyperbolic problems [11]. In these schemes the eigenmatrix (2.24) is decomposed into two matrices containing the positive and negative eigenvalues respectively

$$\bar{\bar{\Lambda}}^\pm(\epsilon, \mu) = \frac{1}{2} \left( \bar{\bar{\Lambda}}(\epsilon, \mu) \pm \text{diag}(|\Lambda_0|, |\Lambda_1|, |\Lambda_2|, |\Lambda_3|, |\Lambda_4|, |\Lambda_5|) \right) \quad (3.8)$$

where the eigenvalues  $\Lambda_p$  are defined according to (2.24). The flux matrix is then split into a matrix  $\bar{\bar{\mathbf{A}}}_{i,k}^+$  and a matrix  $\bar{\bar{\mathbf{A}}}_{i,k}^-$

$$\bar{\bar{\mathbf{A}}}_{i,k}^\pm = \bar{\bar{\mathbf{P}}}(\epsilon_i, \mu_i, \vec{n}_{i,k}) \cdot \bar{\bar{\Lambda}}^\pm(\epsilon_i, \mu_i) \cdot \bar{\bar{\mathbf{P}}}^{-1}(\epsilon_i, \mu_i, \vec{n}_{i,k}). \quad (3.9)$$

In  $V_i$ , these matrices are explicitly:

$$\bar{\bar{\mathbf{A}}}_{i,k}^\pm = \frac{1}{2} \begin{bmatrix} \mp c_i \bar{\bar{s}}_{i,k}^2 & -\epsilon_i^{-1} \bar{\bar{s}}_{i,k} \\ \mu_i^{-1} \bar{\bar{s}}_{i,k} & \mp c_i \bar{\bar{s}}_{i,k}^2 \end{bmatrix} \quad (3.10)$$

where  $c_i = \sqrt{\epsilon_i \mu_i}^{-1}$  is the velocity of field propagation in the volume  $V_i$  and the notation  $\bar{\bar{s}}_{i,k} = \bar{\bar{s}}(\vec{n}_{i,k})$  has been used. From the definitions above it is easily verified that:

$$\begin{aligned} \bar{\bar{\mathbf{A}}}_{i,k} &= \bar{\bar{\mathbf{A}}}_{i,k}^+ + \bar{\bar{\mathbf{A}}}_{i,k}^- \\ \bar{\bar{\mathbf{A}}}_{i,k}^+ &= -\bar{\bar{\mathbf{A}}}_{i,-k}^- \end{aligned} \quad (3.11)$$

where the subscript  $-k$  is used to denote reversal of the normal  $\vec{n}_{i,k}$ .



To see the benefits of this flux-matrix splitting, we will recall the characteristic solution (2.30) developed for the one-dimensional problem (2.22). While the assumption of a one-dimensional propagation does not, in general, hold for the three-dimensional case, we can assume that the fields in  $V_i$  can be written in terms of a plane-wave decomposition in each of the  $\vec{n}_{i,k}$  directions. Each of these plane waves would then satisfy a one-dimensional system of the form (2.22) and the following derivations will hold. A rigorous analysis of this plane-wave assumption is beyond the scope of this thesis.

With the above in mind we can define characteristics propagating in the  $\pm\vec{n}_{i,k}$  direction as:

$$\vec{\mathcal{W}}_{i,k}^{\pm}(t, \vec{r}) \triangleq \frac{1}{\pm c_i} \bar{\bar{\Lambda}}^{\pm}(\epsilon_i, \mu_i) \cdot \bar{\bar{P}}^{-1}(\epsilon_i, \mu_i, \vec{n}_{i,k}) \cdot \vec{\mathcal{U}}(t, \vec{r}). \quad (3.12)$$

where the subscripts  $i, k$  on  $\vec{\mathcal{W}}_{i,k}^{\pm}(t, \vec{r})$  associates the characteristics with  $\vec{n}_{i,k}$  (and does not necessarily imply evaluation on  $\partial V_{i,k}$ ). Introducing the notation  $\bar{\bar{\Lambda}}_i^{\pm} = \bar{\bar{\Lambda}}^{\pm}(\epsilon_i, \mu_i)$  and  $\bar{\bar{P}}_{i,k} = \bar{\bar{P}}(\epsilon_i, \mu_i, \vec{n}_{i,k})$ , we have:

$$\begin{aligned} \pm c_i \bar{\bar{P}}_{i,k} \cdot \vec{\mathcal{W}}_{i,k}^{\pm}(t, \vec{r}) &= \frac{\pm c_i}{\pm c_i} \bar{\bar{P}}_{i,k} \cdot \bar{\bar{\Lambda}}_i^{\pm} \cdot \bar{\bar{P}}_{i,k}^{-1} \cdot \vec{\mathcal{U}}(t, \vec{r}) \\ &= \bar{\bar{A}}_{i,k}^{\pm} \cdot \vec{\mathcal{U}}(t, \vec{r}) \end{aligned} \quad (3.13)$$

implying that the positive and negative flux-matrix  $\bar{\bar{A}}_{i,k}^{\pm}$  are also associated with propagation in the  $\pm\vec{n}_{i,k}$  direction. We consider, therefore, two boundary limits of the solution:  $\vec{\mathcal{U}}_{i,k}^{*}(t, \vec{r})$  is the boundary limit of the solution from inside  $V_i$  and  $\vec{\mathcal{U}}_{i,k}^{**}(t, \vec{r})$  is the boundary limit of the solution outside  $\partial V_{i,k}$  [9] as shown in Figure 3.1. The flux defined in equation (3.7) is a linear function of  $\vec{\mathcal{U}}_{i,k}^o(t, \vec{r})$  and we can decompose the

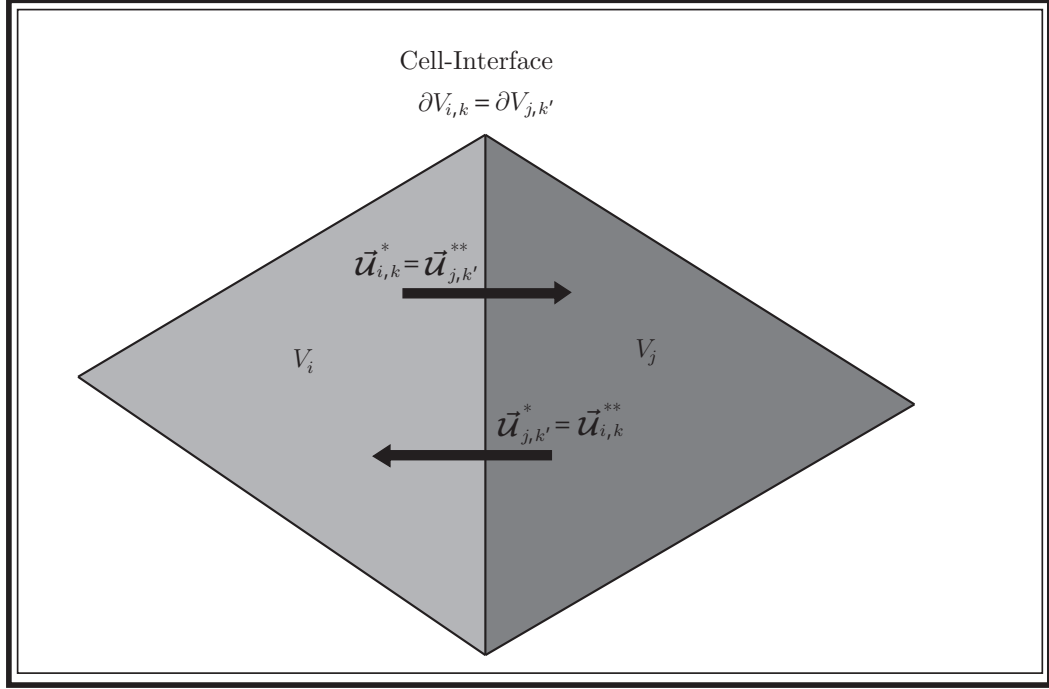


Fig. 3.1: Boundary limits of the solution at a volumetric cell interface (depicted in two dimensions): For  $V_i$  the interior solution is transported into  $V_j$  while the exterior solution is transported into  $V_i$  over the  $k$ th facet  $\partial V_{i,k} = \partial V_{j,k'}$ .

flux according to the decomposition of the flux-matrix:

$$\begin{aligned} \vec{\mathcal{F}}_{i,k}^o(t, \vec{r}) &= \bar{\mathbf{A}}_{i,k}^+ \cdot \vec{\mathbf{u}}_{i,k}^*(t, \vec{r}) + \bar{\mathbf{A}}_{i,k}^- \cdot \vec{\mathbf{u}}_{i,k}^{**}(t, \vec{r}) \\ &= \vec{\mathcal{F}}_{i,k}^*(t, \vec{r}) + \vec{\mathcal{F}}_{i,k}^{**}(t, \vec{r}) \end{aligned} \quad (3.14)$$

Physically, the decomposition (3.14) says that the flux of the solution flowing out of  $V_i$  is to be computed from knowledge of the solution inside  $V_i$ . Conversely, the flux of the solution flowing into  $V_i$  should be computed from knowledge of the solution outside  $V_i$ . Intuitively, such a decomposition makes use of causality, and is commonly referred to as *upwinding*. When upwinding is used in time-marching schemes, it can be shown that the resulting numerical scheme expresses the solution at the next time-step as a convex combination of upwind cell-averages at the current time-step and, provided that a limit is imposed on the time-step increment, the scheme is stable [11]. Determining

the boundary states  $\vec{\mathbf{u}}_{i,k}^*(t, \vec{r})$  and  $\vec{\mathbf{u}}_{i,k}^{**}(t, \vec{r})$  that satisfy Maxwell's Equations are the subject of the following section.

### 3.5 Variable Coefficient Fluxes and Discontinuous Constitutive Parameters

For the numerical solution of Maxwell's Equations using the Finite-Volume method we are considering a partition of the computational domain  $V$  in which each cell  $V_i$  has constant constitutive parameters. This implies that, globally, the flux matrix  $\bar{\mathbf{A}}(\epsilon(\vec{r}), \mu(\vec{r}), \vec{m})$  is both a function of position and discontinuous across cell interfaces corresponding to a jump in the constitutive parameters. Even in the absence of a material jump across a cell boundary, we are required to enforce the appropriate electromagnetic boundary conditions to ensure accuracy of the numerical scheme. For example, across a facet  $\partial V_{i,k}$  between two finite-volumes having the same physical parameters, the solution should be continuous according to standard electromagnetic boundary conditions, *i.e.*  $\vec{\mathbf{u}}_{i,k}^*(t, \vec{r}) = \vec{\mathbf{u}}_{i,k}^{**}(t, \vec{r}) = \vec{\mathbf{u}}_{i,k}^o(t, \vec{r})$ . In practice, however, we will only have numerical approximations to these boundary limits and continuity will not hold due to numerical error. Fortunately, modifying the upwind, flux-split scheme to account for electromagnetic boundary conditions is straight-forward.

Given a facet between two cells, as shown in Figure 3.2, we consider the one-dimensional reconstruction problem with coordinate  $n$  in the direction of the normal  $\vec{n}_{i,k}$ . We assume that we can reconstruct the fields  $\vec{\mathbf{u}}^+(t, \vec{r})$ ,  $\vec{r} \in V_i$  and  $\vec{\mathbf{u}}^-(t, \vec{r})$ ,  $\vec{r} \in V_j$  from the cell average values. The actual mechanism for performing this reconstruction is provided in Section 3.7. It is then possible to use characteristic propagation along the characteristic curves to determine  $\vec{\mathbf{u}}_{i,k}^*(t, \vec{r})$  and  $\vec{\mathbf{u}}_{i,k}^{**}(t, \vec{r})$  from  $\vec{\mathbf{u}}^\pm(t, \vec{r})$  where we

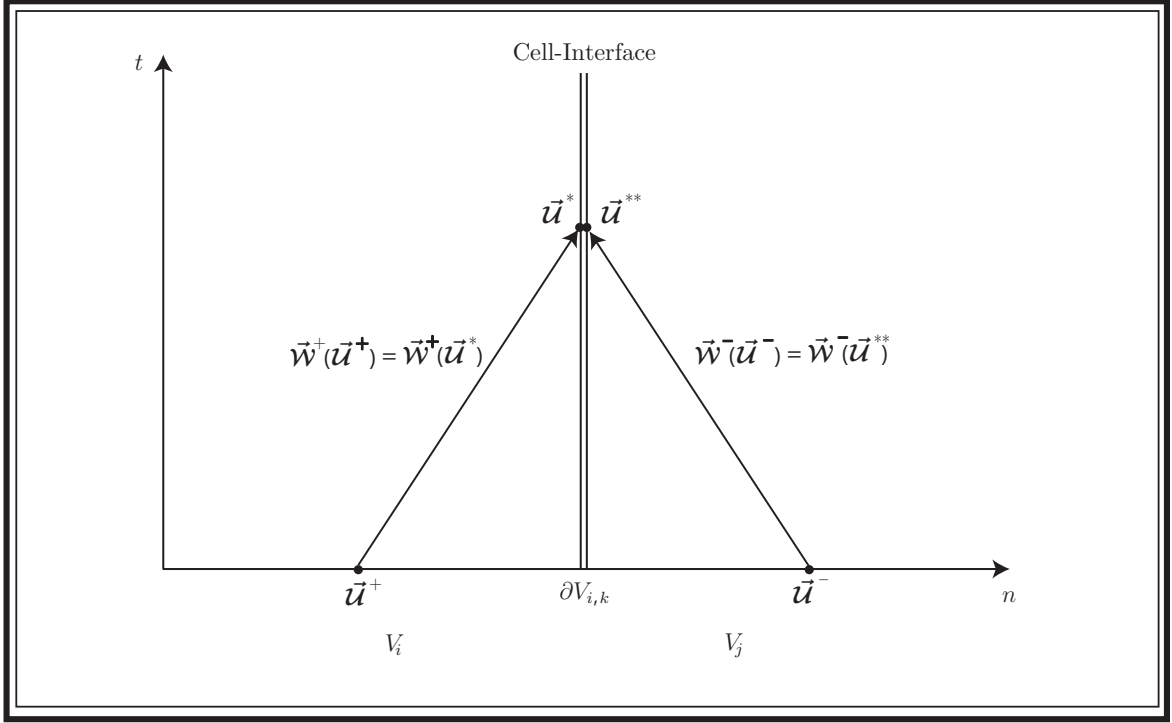


Fig. 3.2: Characteristic reconstruction of the boundary limits at a cell interface. The characteristic vectors propagate unchanged in space and time along the characteristic curves (shown as arrows)

momentarily assume that cell sizes are small enough that the propagation time can be neglected. Using the characteristics in the  $\pm \vec{n}_{i,k}$  direction defined in (3.12), the following relationships hold along the characteristic curves:

$$\begin{aligned} \frac{1}{c_i} \bar{\Lambda}_i^+ \cdot \bar{P}_{i,k}^{-1} \cdot \vec{u}^+ &= \frac{1}{c_i} \bar{\Lambda}_i^+ \cdot \bar{P}_{i,k}^{-1} \cdot \vec{u}_{i,k}^* \\ \frac{1}{-c_j} \bar{\Lambda}_j^- \cdot \bar{P}_{j,-k'}^{-1} \cdot \vec{u}^- &= \frac{1}{-c_j} \bar{\Lambda}_j^- \cdot \bar{P}_{j,-k'}^{-1} \cdot \vec{u}_{i,k}^{**} \end{aligned} \quad (3.15)$$

where the notation  $\bar{P}_{j,-k'}^{-1}$  is used to note that this matrix is being constructed using the negative boundary normal from the perspective of  $V_j$ , and where the explicit time and spatial dependence have been dropped for brevity. Provided that the electromagnetic boundary condition relating  $\vec{u}_{i,k}^*$  and  $\vec{u}_{i,k}^{**}$  are known, the system of equations (3.15) can be solved for, say,  $\vec{u}_{i,k}^*$  in terms of  $\vec{u}^\pm(t, \vec{r})$  as will be illustrated shortly for the most

common electromagnetic boundary conditions. These are simplified by multiplying the first set of equations in (3.15) by  $c_i \bar{\bar{\mathbf{P}}}_{i,k}$  and the second set by  $-c_j \bar{\bar{\mathbf{P}}}_{j,-k'}$  analogous to (3.12):

$$\begin{aligned}\bar{\bar{\mathbf{A}}}_{i,k}^+ \bar{\mathbf{u}}^+ &= \bar{\bar{\mathbf{A}}}_{i,k}^+ \bar{\mathbf{u}}_{i,k}^* \\ \bar{\bar{\mathbf{A}}}_{j,k'}^+ \bar{\mathbf{u}}^- &= \bar{\bar{\mathbf{A}}}_{j,k'}^+ \bar{\mathbf{u}}_{i,k}^{**}\end{aligned}\tag{3.16}$$

where, in the second set of equations, conversion from  $\bar{\bar{\mathbf{A}}}_{j,-k'}^-$  to  $\bar{\bar{\mathbf{A}}}_{j,k'}^+$  is accomplished by applying the second identity in (3.11) to both sides of the equality and canceling the negative signs.

If we again consider Figure 3.2 we can note that if the states  $\bar{\mathbf{u}}^\pm$  are reconstructed infinitesimally near to the interface at time  $t$ , then the characteristic propagation time is infinitesimally small but the relationships (3.16) still hold. This implies that we can reconstruct the boundary states from the cell averages of the solution at time  $t$  by spatial reconstruction alone. Henceforth, therefore, we are interested in the reconstruction of  $\bar{\mathbf{u}}_{i,k}^\pm(t, \vec{r}) = \bar{\mathbf{u}}^\pm(t, \vec{r})$  with  $\vec{r} \in \partial V_{i,k}$  such that (3.16) becomes:

$$\boxed{\begin{aligned}\bar{\bar{\mathbf{A}}}_{i,k}^+ \bar{\mathbf{u}}_{i,k}^+ &= \bar{\bar{\mathbf{A}}}_{i,k}^+ \bar{\mathbf{u}}_{i,k}^* \\ \bar{\bar{\mathbf{A}}}_{j,k'}^+ \bar{\mathbf{u}}_{i,k}^- &= \bar{\bar{\mathbf{A}}}_{j,k'}^+ \bar{\mathbf{u}}_{i,k}^{**}\end{aligned}}\tag{3.17}$$

### 3.6 Flux Reconstruction at Material Interfaces

The general numerical boundary conditions (3.16) are the starting point for reconstructing interface states. The following symmetry is useful on the facet  $\partial V_{i,k}$ :

$$\bar{\mathbf{u}}_{i,k}^-(t, \vec{r}) = \bar{\mathbf{u}}_{j,k'}^+(t, \vec{r}) = \bar{\mathbf{u}}_{j,-k}^+(t, \vec{r})\tag{3.18}$$

We omit the subscript notation and assume we are concerned with the reconstruction at  $\partial V_{i,k} = \partial V_{j,k'}$ . Algebraic manipulations and the use of (3.18) show that the boundary relationships (3.16) are equivalent to:

$$\boxed{\begin{aligned} \bar{s}^2 \cdot (\vec{\mathcal{E}}^* - \vec{\mathcal{E}}^{**}) + \bar{s} \cdot (Z_i \vec{\mathcal{H}}^* + Z_j \vec{\mathcal{H}}^{**}) &= \bar{s}^2 \cdot (\vec{\mathcal{E}}^+ - \vec{\mathcal{E}}^-) + \bar{s} \cdot (Z_i \vec{\mathcal{H}}^+ + Z_j \vec{\mathcal{H}}^-) \\ \bar{s}^2 \cdot (\vec{\mathcal{H}}^* - \vec{\mathcal{H}}^{**}) - \bar{s} \cdot (Y_j \vec{\mathcal{E}}^{**} + Y_i \vec{\mathcal{E}}^*) &= \bar{s}^2 \cdot (\vec{\mathcal{H}}^+ - \vec{\mathcal{H}}^-) - \bar{s} \cdot (Y_j \vec{\mathcal{E}}^- + Y_i \vec{\mathcal{E}}^+). \end{aligned}} \quad (3.19)$$

where the abbreviated notation  $\bar{s}(\vec{n}_{i,k}) = \bar{s}$  has been used. These simplified forms are quite handy for deriving specific boundary conditions and will be used in what follows. Before proceeding, we note that the flux integration in equation (3.7) only involves tangential field components due to the definition of the flux matrix.

### 3.6.1 Flux Reconstruction at Inhomogeneous Dielectrics

For dielectric discontinuities, the boundary conditions for Maxwell's equations require that the tangential field components are continuous across the discontinuity. In terms of the operator  $\bar{s}(\vec{n}_{i,k})$  at the surface  $\partial V_{i,k}$  these conditions can be written as:

$$\begin{aligned} \bar{s} \cdot \vec{\mathcal{E}}^* &= \bar{s} \cdot \vec{\mathcal{E}}^{**} \\ \bar{s} \cdot \vec{\mathcal{H}}^* &= \bar{s} \cdot \vec{\mathcal{H}}^{**} \end{aligned} \quad (3.20)$$

Substituting these constraints into the two equations (3.19) gives:

$$\begin{aligned} \bar{s} \cdot \vec{\mathcal{H}}^* (Z_i + Z_j) &= \bar{s}^2 \cdot (\vec{\mathcal{E}}^+ - \vec{\mathcal{E}}^-) + \bar{s} \cdot (Z_i \vec{\mathcal{H}}^+ + Z_j \vec{\mathcal{H}}^-) \\ \bar{s} \cdot \vec{\mathcal{E}}^* (Y_i + Y_j) &= -\bar{s}^2 \cdot (\vec{\mathcal{H}}^+ - \vec{\mathcal{H}}^-) + \bar{s} \cdot (Y_j \vec{\mathcal{E}}^- + Y_i \vec{\mathcal{E}}^+) \end{aligned} \quad (3.21)$$

which can be re-written as:

$$\begin{aligned} -\bar{s} \cdot \vec{\mathcal{H}}^* &= (Z_i + Z_j)^{-1} \left( -\bar{s}^2 \cdot \vec{\mathcal{E}}^+ - \bar{s} \cdot Z_i \vec{\mathcal{H}}^+ + \bar{s}^2 \cdot \vec{\mathcal{E}}^- - \bar{s} \cdot Z_j \vec{\mathcal{H}}^- \right) \\ \bar{s} \cdot \vec{\mathcal{E}}^* &= (Y_i + Y_j)^{-1} \left( -\bar{s}^2 \cdot \vec{\mathcal{H}}^+ + \bar{s} \cdot Y_i \vec{\mathcal{E}}^+ + \bar{s}^2 \cdot \vec{\mathcal{H}}^- + \bar{s} \cdot Y_j \vec{\mathcal{E}}^- \right) \end{aligned} \quad (3.22)$$

This system can be written in terms of the split flux-matrices as

$$\begin{aligned} \begin{bmatrix} -\bar{s} \cdot \vec{\mathcal{H}}^* \\ \bar{s} \cdot \vec{\mathcal{E}}^* \end{bmatrix} &= \bar{\alpha}_i \cdot \begin{bmatrix} \frac{2Z_i}{Z_i+Z_j} \bar{I} & \bar{0} \\ \bar{0} & \frac{2Y_i}{Y_i+Y_j} \bar{I} \end{bmatrix} \cdot \frac{1}{2} \begin{bmatrix} -c_i \bar{s}^2 & -\epsilon_i^{-1} \bar{s} \\ \mu_i^{-1} \bar{s} & -c_i \bar{s}^2 \end{bmatrix} \cdot \begin{bmatrix} \vec{\mathcal{E}}^+ \\ \vec{\mathcal{H}}^+ \end{bmatrix} + \\ &\bar{\alpha}_j \cdot \begin{bmatrix} \frac{2Z_j}{Z_i+Z_j} \bar{I} & \bar{0} \\ \bar{0} & \frac{2Y_j}{Y_i+Y_j} \bar{I} \end{bmatrix} \cdot \frac{1}{2} \begin{bmatrix} c_j \bar{s}^2 & -\epsilon_j^{-1} \bar{s} \\ \mu_j^{-1} \bar{s} & c_j \bar{s}^2 \end{bmatrix} \cdot \begin{bmatrix} \vec{\mathcal{E}}^- \\ \vec{\mathcal{H}}^- \end{bmatrix} \end{aligned} \quad (3.23)$$

or in a compact form:

$$\bar{\mathbf{L}}_{i,k} \cdot \vec{\mathbf{u}}_{i,k}^* = \bar{\alpha}_i \cdot \bar{\mathbf{T}}_{i,j} \cdot \bar{\mathbf{A}}_{i,k}^+ \cdot \vec{\mathbf{u}}_{i,k}^+ - \bar{\alpha}_j \cdot \bar{\mathbf{T}}_{j,i} \cdot \bar{\mathbf{A}}_{j,k'}^+ \cdot \vec{\mathbf{u}}_{j,k'}^+ \quad (3.24)$$

where  $\bar{\mathbf{T}}_{i,k}$  is the facet transmission matrix:

$$\bar{\mathbf{T}}_{i,j} = \begin{bmatrix} \frac{2Z_i}{Z_i+Z_j} \bar{I} & \bar{0} \\ \bar{0} & \frac{2Y_i}{Y_i+Y_j} \bar{I} \end{bmatrix} \quad (3.25)$$

Finally, the evaluation of the upwind and split flux, equivalent to (3.6) at an inhomogeneous dielectric (noted by the subscript ID) is given by

$$\begin{aligned} \vec{\mathcal{F}}_{i,k|ID}^o &= \bar{\mathbf{A}}_{i,k} \cdot \vec{\mathbf{u}}_{i,k}^o = \bar{\alpha}_i^{-1} \cdot \bar{\mathbf{L}}_{i,k} \cdot \vec{\mathbf{u}}_{i,k}^o \\ &= \bar{\mathbf{T}}_{i,j} \cdot \bar{\mathbf{A}}_{i,k}^+ \cdot \vec{\mathbf{u}}_{i,k}^+ - \bar{\alpha}_i^{-1} \cdot \bar{\alpha}_j \cdot \bar{\mathbf{T}}_{j,i} \cdot \bar{\mathbf{A}}_{j,k'}^+ \cdot \vec{\mathbf{u}}_{j,k'}^+ \end{aligned} \quad (3.26)$$

As a final comment we note that in a scattered-field formulation, both the total- and scattered-fields satisfy the same dielectric boundary conditions so the final form

equations (3.26) also holds for scattering problems.

### 3.6.2 Flux Reconstruction at Homogeneous Dielectrics

When the material parameters across the facet  $\partial V_{i,k}$  are homogeneous dielectrics (HD), the transmission matrices in (3.26) become unity and the constitutive matrices are equal reducing the boundary conditions to:

$$\boxed{\vec{\mathcal{F}}_{i,k|HD}^o = \bar{\mathbf{A}}_{i,k}^+ \cdot \vec{\mathbf{u}}_{i,k}^+ - \bar{\mathbf{A}}_{j,k'}^+ \cdot \vec{\mathbf{u}}_{j,k'}^+ = \bar{\mathbf{A}}_{i,k}^+ \cdot \vec{\mathbf{u}}_{i,k}^+ + \bar{\mathbf{A}}_{i,k}^- \cdot \vec{\mathbf{u}}_{j,k'}^+} \quad (3.27)$$

where the second identity in (3.11) has been used in converting  $-\bar{\mathbf{A}}_{j,k'}^+$  to  $\bar{\mathbf{A}}_{i,k}^-$ .

### 3.6.3 Total-Field Flux Reconstruction at PEC and PMC Boundaries

In the case where finite volume  $V_i$  is bounded on a facet  $k$  by a perfect electric conductor (PEC), the appropriate boundary condition can be obtained by noting that if the facet  $\partial V_{i,k}$  is PEC, then from the perspective of  $V_i$  the fields  $\vec{\mathbf{u}}_{j,k'}^+$  are zero, as is the impedance  $Z_j$ . This leads to the expression:

$$\boxed{\vec{\mathcal{F}}_{i,k|PEC}^o = \bar{\mathbf{T}}_i^{\text{PEC}} \cdot \bar{\mathbf{A}}_{i,k}^+ \cdot \vec{\mathbf{u}}_{i,k}^+, \quad \text{with } \bar{\mathbf{T}}_i^{\text{PEC}} = \lim_{Z_j \rightarrow 0} \bar{\mathbf{T}}_{i,j} = \begin{bmatrix} 2\bar{\mathbf{I}} & \bar{\mathbf{0}} \\ \bar{\mathbf{0}} & \bar{\mathbf{0}} \end{bmatrix}} \quad (3.28)$$

If, instead of a PEC boundary, volume  $V_i$  is bounded on a facet  $k$  by a perfect magnetic conductor (PMC), we can obtain the appropriate boundary conditions by noting that



that from the perspective of  $V_i$  the fields  $\vec{\mathbf{u}}_{j,k'}^+$  are zero as is the admittance  $Y_j$ :

$$\boxed{\vec{\mathcal{F}}_{i,k|PMC}^o = \bar{\mathbf{T}}_i^{PMC} \cdot \bar{\mathbf{A}}_{i,k}^+ \cdot \vec{\mathbf{u}}_{i,k}^+, \quad \text{with } \bar{\mathbf{T}}_i^{PMC} = \lim_{Y_j \rightarrow 0} \bar{\mathbf{T}}_{i,j} = \begin{bmatrix} \bar{0} & \bar{0} \\ \bar{0} & 2\bar{I} \end{bmatrix}} \quad (3.29)$$

Both the PEC and PMC flux-reconstructions above can also be used to incorporate thin PEC/PMC screens separating volumetric cells.

#### 3.6.4 Scattered-Field Flux Reconstruction at PEC and PMC boundaries

A scattered-field reconstruction equation at a PEC or PMC facet can be developed by considering the scattered-field equivalent of the first set of equations in (3.16) which can be written as:

$$\begin{aligned} \bar{s}^2 \cdot \vec{\mathcal{E}}^{*,scat} + \bar{s} \cdot Z_i \vec{\mathcal{H}}^{*,scat} &= \bar{s}^2 \cdot \vec{\mathcal{E}}^{+,scat} + \bar{s} \cdot Z_i \vec{\mathcal{H}}^{+,scat} \\ \bar{s}^2 \cdot \vec{\mathcal{H}}^{*,scat} - \bar{s} \cdot Y_i \vec{\mathcal{E}}^{*,scat} &= \bar{s}^2 \cdot \vec{\mathcal{H}}^{+,scat} - \bar{s} \cdot Y_i \vec{\mathcal{E}}^{+,scat} \end{aligned} \quad (3.30)$$

At a PEC facet under a scattered field formulation we have the relationship:

$$\bar{s} \cdot \vec{\mathcal{E}}^{*,scat} = -\bar{s} \cdot \vec{\mathcal{E}}^{*,inc} \quad (3.31)$$

where the boundary limit of the incident field is assumed to be a known function for all time and space. Substituting this equation into the first of equations (3.30) allows us to solve for the boundary state  $-\bar{s} \cdot \vec{\mathcal{H}}^{*,scat}$ . We obtain

$$\begin{bmatrix} -\bar{s} \cdot \vec{\mathcal{H}}^{*,scat} \\ \bar{s} \cdot \vec{\mathcal{E}}^{*,scat} \end{bmatrix} = \begin{bmatrix} -Y_i \bar{s}^2 \cdot \vec{\mathcal{E}}^{*,inc} - Y_i \bar{s}^2 \cdot \vec{\mathcal{E}}^{+,scat} - \bar{s} \cdot \vec{\mathcal{H}}^{+,scat} \\ -\bar{s} \cdot \vec{\mathcal{E}}^{*,inc} \end{bmatrix} \quad (3.32)$$

which can be written concisely as:

$$\begin{bmatrix} -\bar{s} \cdot \vec{\mathcal{H}}_{i,k}^{*,scat} \\ \bar{s} \cdot \vec{\mathcal{E}}_{i,k}^{*,scat} \end{bmatrix} = \bar{\alpha}_i \cdot \bar{\mathbf{T}}_i^{PEC} \cdot \bar{\mathbf{A}}_{i,k}^+ \cdot \vec{\mathbf{u}}_{i,k}^{+,scat} + \bar{\alpha}_i \cdot 2\bar{\mathbf{A}}_{i,k}^- \cdot \begin{bmatrix} -\vec{\mathcal{E}}_{i,k}^{*,inc} \\ 0 \end{bmatrix} \quad (3.33)$$

such that the flux becomes

$$\boxed{\vec{\mathcal{F}}_{i,k|PEC}^{o,scat} = \bar{\mathbf{T}}_i^{PEC} \cdot \bar{\mathbf{A}}_{i,k}^+ \cdot \vec{\mathbf{u}}_{i,k}^{+,scat} - 2\bar{\mathbf{A}}_{i,k}^- \cdot \begin{bmatrix} \vec{\mathcal{E}}_{i,k}^{*,inc} \\ 0 \end{bmatrix}} \quad (3.34)$$

For a PMC boundary condition under a scattered-field formulation we have the condition that

$$\bar{s} \cdot \vec{\mathcal{H}}^{*,scat} = -\bar{s} \cdot \vec{\mathcal{H}}^{*,inc} \quad (3.35)$$

Substituting this condition into the second set of equations in (3.30) and following a procedure analogous to the one for PEC facets above we arrive at the scattered-field PMC flux reconstruction equations

$$\boxed{\vec{\mathcal{F}}_{i,k|PMC}^{o,scat} = \bar{\mathbf{T}}_i^{PMC} \cdot \bar{\mathbf{A}}_{i,k}^+ \cdot \vec{\mathbf{u}}_{i,k}^{+,scat} - 2\bar{\mathbf{A}}_{i,k}^- \cdot \begin{bmatrix} 0 \\ \vec{\mathcal{H}}_{i,k}^{*,inc} \end{bmatrix}} \quad (3.36)$$

where again the incident magnetic field is assumed to be a known function.

### 3.7 Spatial Reconstruction Schemes

The method of lines FVTD formulation of Maxwell's Equations is not complete without detailing how  $\vec{\mathbf{u}}_{i,k}^+$  and  $\vec{\mathbf{u}}_{i,k}^-$  should be reconstructed from the cell averages. Depending

on the desired spatial accuracy, different reconstruction schemes can be developed. In what follows we will assume that volume  $V_j$  bounds  $V_i$  on facet  $\partial V_{i,k} = \partial V_{j,k'}$

A small comment regarding terminology should be made. In approximation theory, we state that we have an  $n^{\text{th}}$ -order *approximation* to a function if the approximation contains the first  $n$  terms of the functions' Taylor's Expansion. This is in contrast to the order of *accuracy* of a numerical method, where we say that a numerical method is accurate to order  $n$  if the error terms are order  $n$ . Herein we are concerned only with the order of accuracy of the FVTD method given a reconstruction scheme, and references to "order" should be understood in this context.

Before discussing the reconstruction schemes, we present some theory. Consider the volumetric average of a spatially varying function  $f(\vec{r})$  over  $V_i$ , to be used in a second-order accurate algorithm, and represented by its Taylor's expansion about the barycentre  $\vec{r}_i$  of  $V_i$ :

$$\begin{aligned} f_{(i)} &= \frac{1}{|V_i|} \iiint_{V_i} f(\vec{r}) dV \\ &= \frac{1}{|V_i|} \iiint_{V_i} \left( f(\vec{r}_i) + (\vec{r} - \vec{r}_i) \cdot (\nabla f(\vec{r}))|_{\vec{r}=\vec{r}_i} + O(|\Delta R|^2) \right) dV \\ &= f(\vec{r}_i) + O(|\Delta R|^2) \end{aligned} \quad (3.37)$$

where  $\Delta R = |\vec{r} - \vec{r}_i|$  such that  $O(\Delta R)$  can be interpreted as being on the order of cell-size. This equality holds because

$$\vec{r}_i \triangleq \frac{1}{|V_i|} \iiint_{V_i} \vec{r} dV$$

From (3.37), we can conclude that any volume (surface) average quantity is a second-order accurate approximation to the quantity at the volume (surface) barycentre. As

a corollary we also have:

$$\iiint_{V_i} f(\vec{r}) dV = |V_i| f_{(i)} = |V_i| f(\vec{r}_i) + O(|\Delta R|^2) \quad (3.38)$$

providing a method for evaluating volume (surface) integrals to second-order using only the value at the barycentre of the volume (surface).

### 3.7.1 First-Order Spatial Reconstruction

The simplest form of spatial reconstruction available for computing the boundary values of the fields from the cell-averages is to make the first-order assumption that  $\vec{u}_{i,k}^+(t, \vec{r})$ , is equal to the barycentric value  $\vec{u}(t, \vec{r}_i)$ . From (3.37) a first-order reconstruction scheme can be written as:

$$\boxed{\vec{u}_{i,k}^+(t, \vec{r}) = \vec{u}_{(i)}(t) + O(\Delta R) \quad \text{and} \quad \vec{u}_{i,k}^-(t, \vec{r}) = \vec{u}_{(j)}(t) + O(\Delta R)} \quad (3.39)$$

where the reconstruction on  $\partial V_{i,k}$  is independent of position. If we reconstruct the solution at the barycentre  $\vec{r}_{i,k}$  of the facet  $\partial V_{i,k}$  to first order, the flux integral in (3.7) can be evaluated using the surface equivalent of (3.38) to maintain first-order accuracy. In practice, this type of reconstruction suffers from unacceptable numerical dissipation and a higher-order reconstruction must be obtained [11].

### 3.7.2 Second-Order Spatial Reconstruction: MUSCL

The *Monotone Upwind Scheme for Conservation Laws* or MUSCL uses the flux-split, upwind formulation in conjunction with a second-order boundary state reconstruction. The MUSCL scheme was first proposed by Van Leer [25] and is second-order accurate

in space. As a consequence of (3.38), the integral of the flux over the facet  $\partial V_{i,k}$  can be reconstructed by multiplying the flux at the barycentre  $\vec{r}_{i,k}$  by the area  $|\partial V_{i,k}|$  of the facet. That is:

$$\iint_{\partial V_{i,k}} \vec{\mathcal{F}}_{i,k}^o(t, \vec{r}) = |\partial V_{i,k}| \vec{\mathcal{F}}_{i,k}^o(t, \vec{r}_{i,k}) + O(\Delta R^2) \quad (3.40)$$

Therefore, for second-order accuracy in the flux integral, we require second-order accuracy of the flux at the barycentre of the facet. We consider a single component of  $\vec{\mathcal{U}}_{i,k}^+(t, \vec{r})$ , say the  $x$ -component of the electric field  $\mathcal{E}_x(t, \vec{r})$ , at the facet barycentre  $\vec{r}_{i,k}$  of the facet  $\partial V_{i,k}$ . Writing this facet-centered component as a Taylor's series expanded about the volumetric barycentre  $\vec{r}_i$  of  $V_i$  gives:

$$\mathcal{E}_x(t, \vec{r}_{i,k}) = \mathcal{E}_x(t, \vec{r}_i) + (\vec{r}_{i,k} - \vec{r}_i) \cdot \nabla \mathcal{E}_x(t, \vec{r}_i) + O(|\Delta R|^2) \quad (3.41)$$

From (3.37), we can substitute the cell-average  $\mathcal{E}_{x(i)}(t)$  into the above expansion while maintaining the desired accuracy:

$$\mathcal{E}_x^+(t, \vec{r}_{i,k}) = \mathcal{E}_{x(i)}(t) + (\vec{r}_{i,k} - \vec{r}_i) \cdot \nabla \mathcal{E}_x(t, \vec{r}_i) + O(|\Delta R|^2) \quad (3.42)$$

and so, we require a method for evaluating the gradient of the field at the barycenter of a volume to *first-order* accuracy; the gradient is effectively multiplied by  $\Delta R$  resulting in second-order accuracy overall. From the results of (3.37) and an application of the divergence theorem, we have the following second-order relationships:

$$\begin{aligned} \nabla \mathcal{E}_x(t, \vec{r}_i) &= \frac{1}{|V_i|} \iiint_{V_i} \nabla \mathcal{E}_x(t, \vec{r}) dV + O(|\Delta R|^2) \\ &= \frac{1}{|V_i|} \sum_{k=1}^{K_i} \iint_{\partial V_{i,k}} \mathcal{E}_x(t, \vec{r}) \vec{n}_{i,k} dS + O(|\Delta R|^2) \end{aligned} \quad (3.43)$$

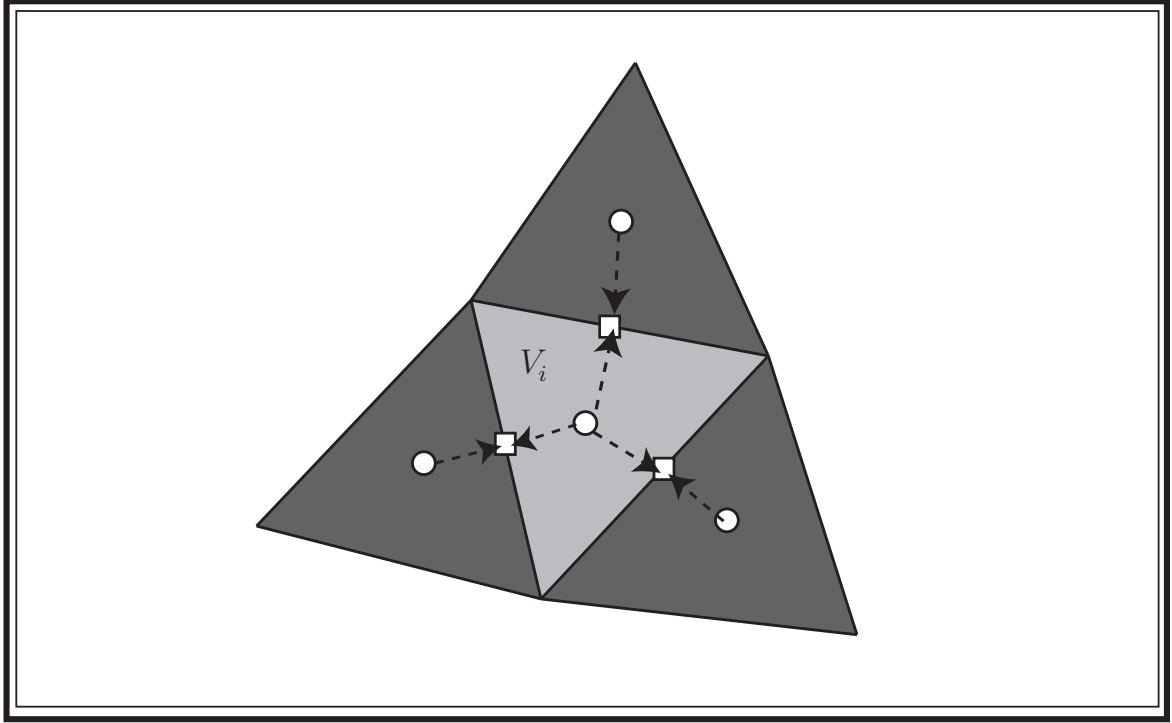


Fig. 3.3: Computation of the MUSCL gradient for volume  $V_i$  depicted as a two-dimensional volume. The facet-centroid fields (square boxes) are computed as a weighted average of the solutions at the volume centroids (circular boxes) across each facet.

It follows from (3.40) that

$$\nabla \mathcal{E}_x(t, \vec{r}_i) = \frac{1}{|V_i|} \sum_{k=1}^{K_i} |\partial V_{i,k}| \mathcal{E}_x(t, \vec{r}_{i,k}) \vec{n}_{i,k} + O(|\Delta R|^2) \quad (3.44)$$

providing an explicit equation for the gradient. The only unknowns in the right-hand-side of (3.44) are the facet-centroid values  $\mathcal{E}_x(t, \vec{r}_{i,k})$ , *the same quantities that we are seeking in order to evaluate (3.40)*! The major difference, however, is that we only require the gradient to first order. Therefore, we seek an initial first-order estimate to the facet-centroid values  $\mathcal{E}_x(t, \vec{r}_{i,k})$  which can be obtained by a simple weighted average:

$$\mathcal{E}_x(t, \vec{r}_{i,k}) = \frac{d_{j,k'} \mathcal{E}_x(t, \vec{r}_i) + d_{i,k} \mathcal{E}_x(t, \vec{r}_j)}{d_{j,k'} + d_{i,k}} + O(|\Delta R|) \quad (3.45)$$

where  $d_{i,k} = |\vec{r}_{i,k} - \vec{r}_i|$ . Using these values to compute (3.44) computes the gradient with second-order accuracy, which in turn provides second-order accurate reconstructed facet-centre values. The computation of the gradient is shown pictorially in Figure 3.3

### 3.7.3 Higher-Order Spatial Reconstruction

It is possible to develop higher-order schemes for spatially reconstructing the solution from each side of a cell boundary. The *Essentially Non-Oscillatory* (ENO)-based high-order polynomial reconstruction scheme presented in [26] and the so-called *Weighted-ENO* (WENO) schemes [20] are two such methods. By using high-order reconstruction we can integrate the surface flux to the desired precision by using an appropriately selected numerical quadrature rule. While mentioned briefly for completeness, these schemes are beyond the scope of this work.

### 3.7.4 The Spatial Stencil of an Element

Corresponding to the selected spatial reconstruction scheme, it is apparent that the flux-integration on  $V_i$  at time  $t$  is dependent on the solution of elements surrounding  $V_i$  as shown in Figure 3.4. We refer to the collection of all elements contributing to the flux-integration of  $V_i$  as the *stencil* for  $V_i$ . In the case of first-order flux-reconstruction, the integration on  $V_i$  is only dependent on the immediate neighbours. In MUSCL reconstruction, the flux integration is dependent on the reconstructed solution in  $V_i$  and each of its neighbours, but the gradient computation extends the stencil to the neighbours of neighbours. For higher-order flux-reconstruction schemes, this trend continues, requiring an increasingly broader stencil in the vicinity of  $V_i$ . The flux-reconstruction stencils are important when considering the local-time-stepping scheme

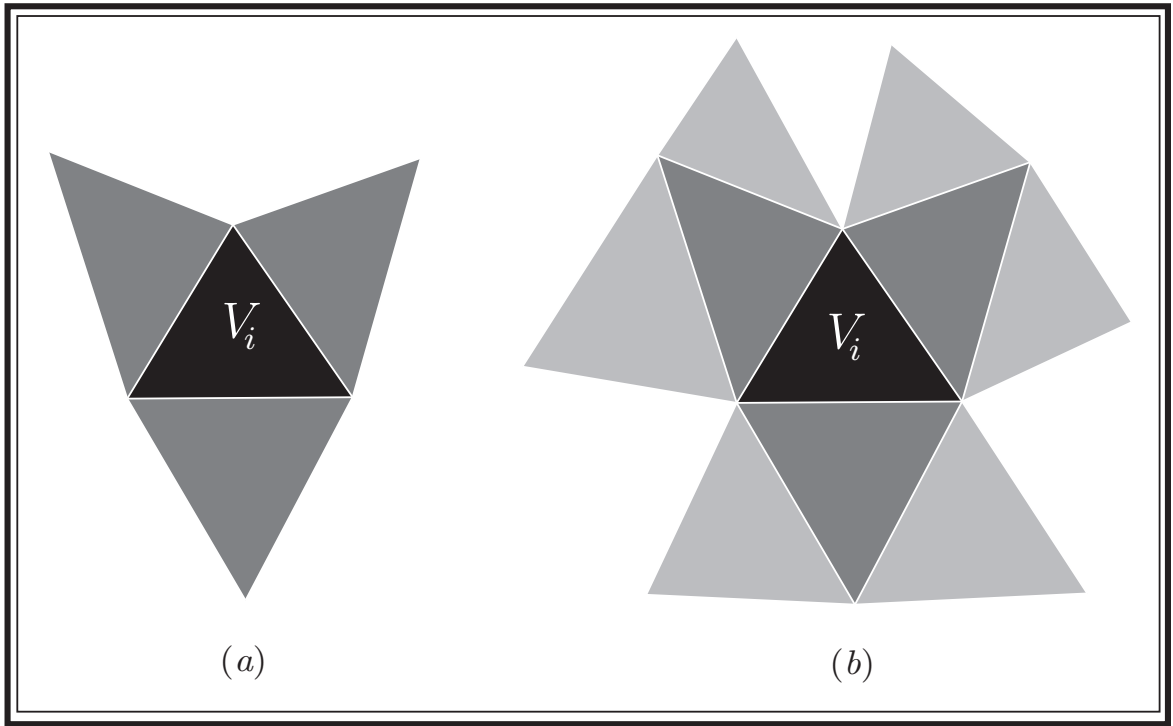


Fig. 3.4: Spatial reconstruction stencils on an element  $V_i$ . The first-order stencil (left) requires knowledge of the solution in neighbouring cells. The MUSCL stencil (right) requires knowledge of both the solution and its gradient in neighbouring cells. The gradient in a neighbour requires the neighbour's neighbours.

discussed in Chapter 4, as well as the parallelization of the algorithm presented in Chapter 10.

### 3.8 The Global, Semi-Discrete, FVTD System and Spatial Computational Complexity

From the formulation presented in the preceding sections of this chapter we can now formulate a global, semi-discrete system for the FVTD solution to Maxwell's Equations. We associate with each cell  $V_i$  the volumetric average solution  $\vec{\mathbf{u}}_{(i)}(t)$  which can be written globally as a vector  $\vec{\mathbf{u}}(t)$  of length  $6N_V$  where  $N_V$  is the number of



cells. Similarly we introduce the global source vector  $\underline{\underline{\mathcal{S}}}$ . Then, abstractly, the global semi-discrete system can be described as

$$\boxed{d_t \underline{\underline{\mathcal{U}}}(t) = -\underline{\underline{\mathcal{F}}} \cdot \underline{\underline{\mathcal{R}}} \cdot \underline{\underline{\mathcal{U}}}(t) + \underline{\underline{\Sigma}} \cdot \underline{\underline{\mathcal{U}}}(t) + \underline{\underline{\mathcal{S}}}(t)} \quad (3.46)$$

The sparse global matrix  $\underline{\underline{\mathcal{F}}}$  is understood to account for the flux integration and operates on the boundary states obtained by operating with the sparse reconstruction matrix  $\underline{\underline{\mathcal{R}}}$  on the cell-averages. The diagonal matrix  $\underline{\underline{\Sigma}}$  accounts for the conductive losses.

The memory and computational complexity of this system at a single time-step can be summarized for a computational domain consisting of  $N_V$  volumes as follows:

- Assuming we are storing the geometric configuration of the mesh, each element can be represented by a constant amount of memory and the mesh storage requires  $O(N_V)$  memory.
- For each  $V_i$ , spatial reconstruction requires knowledge of the geometric properties of a constant number of neighbouring cells. Therefore the memory storage for the reconstruction matrix  $\underline{\underline{\mathcal{R}}}$  and the computational complexity for evaluating the reconstructed states is  $O(N_V)$  (as the number of facets in a volumetric partition scales with the number of volumes). The leading constant grows with the order of the spatial reconstruction scheme in two ways, first the reconstruction is required at a larger number of points, and second the reconstruction at each point is dependent on a greater number of neighbouring elements.
- For each  $V_i$ , flux integration requires knowledge of the geometric properties of  $V_i$  itself and memory scales as  $O(N_V)$ . Evaluation of the flux integral, given the reconstructed states, requires a constant amount of time on each cell and the

complexity scales as  $O(N_V)$ .

- At time  $t$ ,  $O(N_V)$  memory is required to store the solution. Any conductive loss terms can be computed directly from the solution in  $V_i$ .

From the above summary it is clear that the semi-discrete scheme is  $O(N_V)$  in both time- and memory. It should also be clear that provided the flux and reconstruction matrices are stored in memory, the *system* (3.46) no longer requires knowledge of the mesh. It is not, however, desirable to remove the mesh from memory – post-processing, visualization and parallelization strategies are highly mesh-dependent. In practice, we choose not to store the flux- and reconstruction-matrices as, for large problems involving millions of volumes, the additional storage of both the mesh and the matrices can become prohibitive. Instead, each block of 6 rows in (3.46) can be evaluated by performing the reconstruction and flux integration from scratch in constant time at each time-step.

### 3.9 Chapter Summary

In this chapter an upwind, flux-split, semi-discrete (method of lines) formulation of FVTD for Maxwell's Equations has been presented including spatial reconstruction schemes and flux integration that accounts for electromagnetic boundary conditions. To complete the description of the FVTD algorithm for computing approximate solutions to field problems it remains to integrate the temporal derivative and truncate the computational domain. Solutions to these problems are the subject of Chapters 4 and 5 respectively.

# Time-Integration

---

*I want to go ahead of Father Time with a scythe of my own.* – H.G. Wells

The development presented thus far constitutes a semi-discrete numerical scheme for solving Maxwell's Equations using the FVTD method. It remains to numerically integrate the resulting system of ordinary differential equations (3.46) (repeated here for convenience)

$$d_t \vec{\underline{\mathbf{u}}}(t) = -\vec{\underline{\mathcal{F}}}(\vec{\underline{\mathbf{u}}}(t)) + \underline{\underline{\Sigma}} \cdot \vec{\underline{\mathbf{u}}}(t) + \vec{\underline{\mathcal{S}}}(t) \quad (4.1)$$

where we are using the abbreviated notation  $\vec{\underline{\mathcal{F}}}(\vec{\underline{\mathbf{u}}}(t)) \triangleq \underline{\underline{\mathcal{F}}} \cdot \underline{\underline{\mathcal{R}}} \cdot \vec{\underline{\mathbf{u}}}(t)$ . Herein we consider only explicit time-integration schemes as, for hyperbolic systems, they are generally more efficient than implicit methods [11]. In this chapter we provide the details for select global explicit time-marching algorithms. We also provide the details of a local time-stepping (LTS) strategy that exploits local stability conditions on unstructured

grids. The LTS scheme adopted is that of Fumeaux *et. al.* [15], [13]. Details of the LTS scheme are provided such that additional FVTD features, developed in the following chapters, can be formulated within the LTS framework.

## 4.1 Global Time-Stepping Schemes

A global time-stepping scheme maintains synchronization of the time-evolving solution on each element: that is, the solution in each element is computed for the same instant of time. Depending on the desired order of accuracy for approximating the time-derivative in (4.1) above, different time-marching schemes can be formulated. The superscript notation  $n$  is reserved for function evaluation at time  $t = n\Delta t$  for integer values  $n$  and some global time-step  $\Delta t$ . For the sake of brevity we will derive the update equations without source terms, as in both a total- and scattered-field formulation, the sources are assumed to be known over the computational domain for all time.

### 4.1.1 Explicit Forward-Euler Time-Integration

The simplest explicit scheme is the first-order, Forward-Euler time-integration scheme with global time-step  $\Delta t$ . When loss is present, a backward difference of the loss term is required for numerical stability as discussed below. The first-order time-integration can then be written as

$$\underline{\vec{u}}^{n+1} = \underline{\vec{u}}^n - \Delta t \underline{\vec{F}}(\underline{\vec{u}}^n) + \Delta t \underline{\underline{\Sigma}} \cdot \underline{\vec{u}}^{n+1} + O(|\Delta t|^2) \quad (4.2)$$

which can be simplified to

$$\boxed{\text{Forward Euler: } \underline{\vec{u}}^{n+1} = (\underline{\mathbf{I}} - \Delta t \underline{\Sigma})^{-1} \cdot (\underline{\vec{u}}^n - \Delta t \underline{\mathcal{F}}(\underline{\vec{u}}^n)) + O(|\Delta t|^2)} \quad (4.3)$$

where  $\underline{\mathbf{I}}$  is the global  $6N_V \times 6N_V$  identity matrix and where the matrix to be inverted is diagonal. Evaluation of the conductive terms on the right-hand-side of (4.2) at time  $n + 1$  ensures numerical stability. To elucidate this point we consider the scheme without conductive losses:

$$\underline{\vec{u}}^{n+1} = \underline{\vec{u}}^n - \Delta t \underline{\mathcal{F}}(\underline{\vec{u}}^n) + O(|\Delta t|^2) \quad (4.4)$$

It can be shown that this system is stable if the time-step-scaled flux-reconstruction matrix has eigenvalues less than unity [27]. We will assume this to be the case but will also allow the eigenvalues to be infinitesimally close to one. Keeping in mind that the entries of the matrix  $\underline{\Sigma}$  are, by definition, negative (see the definition (2.10) of the conduction matrix  $\bar{\sigma}(\vec{r})$ ), the effect of the loss approximation at  $n + 1$  divides the diagonal of the system (4.4) by a number larger than zero, ensuring that the eigenvalues remain below unity. Consequently, if the loss-free system (4.4) is stable, so is (4.3).

#### 4.1.2 Two-Step Explicit Predictor Corrector Time-Integration

A second-order-accurate-in-time Predictor-Corrector scheme can be constructed by first computing the solution at an intermediate time-step  $n + 1/2$  using the Forward-Euler scheme and then using this estimate as a correction to a full step over an interval of  $\Delta t$ . The Forward-Euler step is referred to as the predictor:

$$\boxed{\text{Predictor: } \underline{\vec{u}}^{n+\frac{1}{2}} = \left( \underline{\mathbf{I}} - \frac{\Delta t}{2} \underline{\Sigma} \right)^{-1} \cdot \left( \underline{\vec{u}}^n - \frac{\Delta t}{2} \underline{\mathcal{F}}(\underline{\vec{u}}^n) \right) + O(|\Delta t|^2)} \quad (4.5)$$

These values are used to numerically evaluate the fluxes at the intermediate step  $n+1/2$  to be used in a centered-difference over a full time-step

$$\underline{\vec{u}}^{n+1} = \underline{\vec{u}}^n - \Delta t \underline{\vec{F}}(\underline{\vec{u}}^{n+\frac{1}{2}}) + \Delta t \underline{\underline{\Sigma}} \cdot \frac{(\underline{\vec{u}}^{n+1} + \underline{\vec{u}}^n)}{2} + O(|\Delta t|^3)$$

Once again, the loss term is handled carefully, this time using a trapezoidal rule. This last equation can judiciously be written in terms of the difference between the solutions at step  $n+1$  and  $n$ :

$$\underline{\vec{u}}^{n+1} - \underline{\vec{u}}^n = -\Delta t \underline{\mathcal{F}}(\underline{\vec{u}}^{n+\frac{1}{2}}) + \Delta t \underline{\underline{\Sigma}} \cdot \frac{(\underline{\vec{u}}^{n+1} - \underline{\vec{u}}^n)}{2} + \Delta t \underline{\underline{\Sigma}} \cdot \underline{\vec{u}}^n + O(|\Delta t|^3)$$

Collecting terms involving  $\underline{\vec{u}}^{n+1} - \underline{\vec{u}}^n$  gives:

$$\left( \underline{\underline{\mathbf{I}}} - \frac{\Delta t}{2} \underline{\underline{\Sigma}} \right) \cdot (\underline{\vec{u}}^{n+1} - \underline{\vec{u}}^n) = -\Delta t \underline{\vec{F}}(\underline{\vec{u}}^{n+\frac{1}{2}}) + \Delta t \underline{\underline{\Sigma}} \cdot \underline{\vec{u}}^n + O(|\Delta t|^3)$$

and the explicit second-order update can be solved as

$$\boxed{\begin{aligned} \text{Corrector: } \underline{\vec{u}}^{n+1} &= \left( \underline{\underline{\mathbf{I}}} + \Delta t \left( \underline{\underline{\mathbf{I}}} - \frac{\Delta t}{2} \underline{\underline{\Sigma}} \right)^{-1} \cdot \underline{\underline{\Sigma}} \right) \cdot \underline{\vec{u}}^n \\ &\quad - \Delta t \left( \underline{\underline{\mathbf{I}}} - \frac{\Delta t}{2} \underline{\underline{\Sigma}} \right)^{-1} \cdot \underline{\vec{F}}(\underline{\vec{u}}^{n+\frac{1}{2}}) + O(|\Delta t|^3) \end{aligned}} \quad (4.6)$$

### 4.1.3 Second Order Runge-Kutta

An alternate second-order time-stepping scheme is the second-order Runge-Kutta (RK2) scheme [28]. Let us assume that the computational domain  $V$  is source-free

and lossless such that (4.1) becomes:

$$d_t \vec{\mathbf{u}}(t) = -\vec{\mathcal{F}}(\vec{\mathbf{u}}(t))$$

The time-rate-of-change, or slope, of  $\vec{\mathbf{u}}_i(t)$  is given by right-hand-side of this equation.

The RK2 scheme computes two slopes as follows:

$$\begin{aligned} \vec{\mathcal{F}}_{\text{RK2},1}^n &= -\vec{\mathcal{F}}(\vec{\mathbf{u}}^n) \\ \vec{\mathcal{F}}_{\text{RK2},2}^n &= -\vec{\mathcal{F}}(\vec{\mathbf{u}}^n - \Delta t \vec{\mathcal{F}}_{\text{RK2},1}^n) \end{aligned} \quad (4.7)$$

and constructs a update scheme by averaging these slopes over  $\Delta t$

$$\boxed{\text{RK2: } \vec{\mathbf{u}}^{n+1} = \vec{\mathbf{u}}^n + \frac{\Delta t}{2} \left( \vec{\mathcal{F}}_{\text{RK2},1}^n + \vec{\mathcal{F}}_{\text{RK2},2}^n \right) + O(|\Delta t|^3)} \quad (4.8)$$

#### 4.1.4 Fourth Order Runge-Kutta

The RK2 scheme presented above can be generalized to higher-orders as detailed in [28]. A fourth-order scheme (RK4) uses the slopes

$$\begin{aligned} \vec{\mathcal{F}}_{\text{RK4},1}^n &= -\vec{\mathcal{F}}(\vec{\mathbf{u}}^n) \\ \vec{\mathcal{F}}_{\text{RK4},2}^n &= -\vec{\mathcal{F}}(\vec{\mathbf{u}}^n - (\Delta t/2)\vec{\mathcal{F}}_{\text{RK4},1}^n) \\ \vec{\mathcal{F}}_{\text{RK4},3}^n &= -\vec{\mathcal{F}}(\vec{\mathbf{u}}^n - (\Delta t/2)\vec{\mathcal{F}}_{\text{RK4},2}^n) \\ \vec{\mathcal{F}}_{\text{RK4},4}^n &= -\vec{\mathcal{F}}(\vec{\mathbf{u}}^n - \Delta t\vec{\mathcal{F}}_{\text{RK4},3}^n) \end{aligned} \quad (4.9)$$

and constructing the update scheme as:

$$\boxed{RK4: \quad \vec{u}^{n+1} = \vec{u}^n - \frac{\Delta t}{6} \left( \vec{\mathcal{F}}_{\text{RK4},1}^n + 2\vec{\mathcal{F}}_{\text{RK4},2}^n + 2\vec{\mathcal{F}}_{\text{RK4},3}^n + \vec{\mathcal{F}}_{\text{RK4},4}^n \right) + O(|\Delta t|^5)}$$
(4.10)

## 4.2 Time-Stepping Stability

For each of the global time-marching schemes the time-step  $\Delta t$  must be selected to ensure global stability of the solution. It is shown in [29] that a necessary condition for stability on element  $V_i$  is:

$$\boxed{\Delta t_i \leq \frac{2}{c_i} \frac{|V_i|}{|\partial V_i|}}$$
(4.11)

where we note that for a cubical cell this condition reduces to

$$\Delta t_i \leq \frac{2}{c_i} \frac{\Delta x^3}{6\Delta x^2} \leq \frac{\Delta x}{3c_i}$$

where  $\Delta x$  is the length of the edge of the cube. Therefore we see that this condition is in fact more stringent (by a factor of  $\sqrt{3}$ ) than the standard Courant-Friedrichs-Lewy (CFL) condition [11]. A less stringent stability condition, that reduces to the CFL condition on cubical meshes, has been developed in [30] but is beyond the scope of this work.

A necessary condition for global stability is the adherence to the local condition (4.11)



on each cell. Therefore, the global time-step is selected as:

$$\boxed{\Delta t \leq \min_i \Delta t_i} \quad (4.12)$$

### 4.3 Local Time-Stepping

So far, in this chapter, we have provided the details for explicit time-integration schemes using a globally defined time-step  $\Delta t$ . This time-step is not, however, dependent on the global geometry of the partition  $\{V_i\}$  but is limited by some worst-case geometric properties of the individual cells as illustrated by (4.12) and (4.11). This fact motivates the use of local time-stepping (LTS) schemes where the global partition  $V_i$  is divided into geometrically connected subsets containing elements that have approximately the same time-step stability limits. One successful and straightforward LTS scheme is the “power-of-two scheme” of Fumeaux *et. al.* presented in [15], [13]. This LTS scheme is attractive as it is not algorithmically complicated, incurs minimal computational overhead, and provides a significant increase in computation efficiency in strongly inhomogeneous meshes. Even in the case of relatively homogeneous volume sizes, poorly structured volumes can limit the global time-step and the LTS scheme offers significant computational speedup. The algorithm can be decomposed into two parts: first is a preprocessing step that assigns to each element a local time-step and a classification depending on the time-step of its neighbours; and second is the global time-integration scheme that is performed using the local time-step classification.

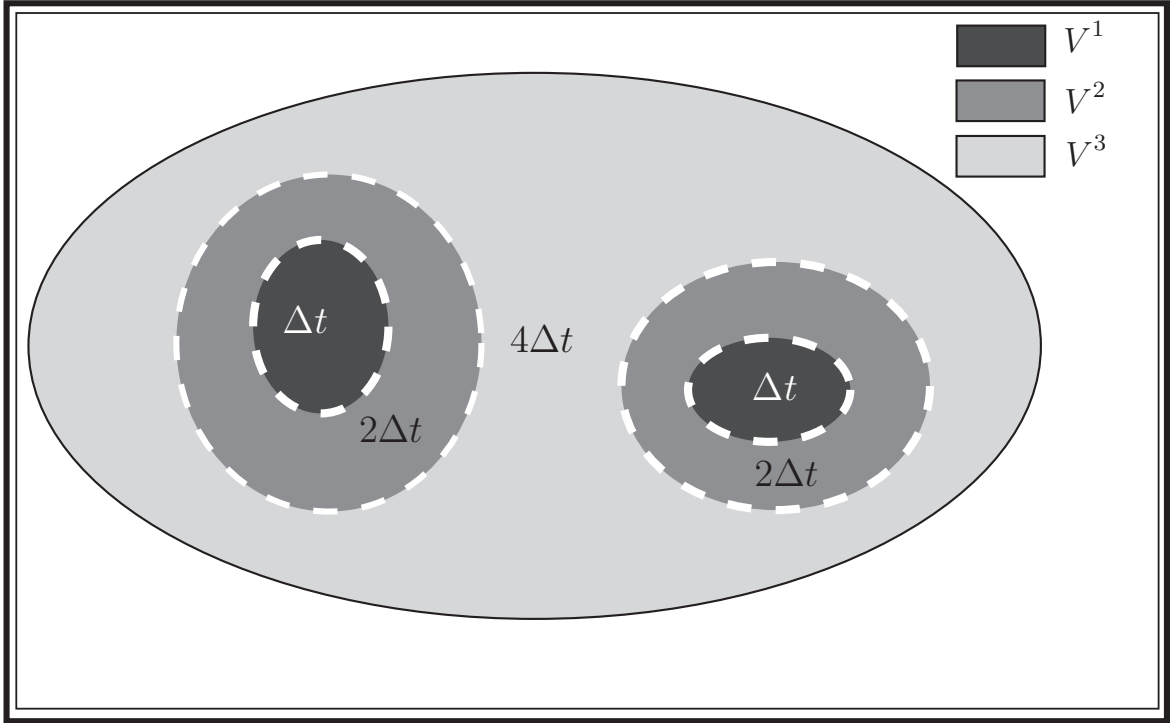


Fig. 4.1: General depiction of a global partitioning of  $V$  into LTS partitions  $V^\ell$ . A three-level LTS scheme is shown in  $V$ . The boundaries between LTS levels are denoted by the white dashed curves.

#### 4.3.1 The LTS Preprocessing Algorithm

The function of the LTS preprocessing algorithm is to subdivide the partition  $\{V_i\}$  of  $V$  into sub-domains  $V^\ell$ ,  $\ell = 1, \dots, \ell_{max}$  that are locally stable at a time-step of  $2^{\ell-1}\Delta t$ , where  $\Delta t$  is the globally computed minimum time-step (4.12). This condition is referred to as the “local-stability condition” of the LTS scheme in [15]. The general idea is depicted in Figure 4.1. The level 1 region  $V^1$  is evolved at  $\Delta t$ , level 2 at  $2\Delta t$ , level 3 at  $4\Delta t$  and so on. The only complication that arises is in synchronizing elements that have stencils that cross the boundaries  $\partial V^\ell$  of the LTS domains (recall that the stencil of an element is the collection of elements required to compute the flux-integration at time  $t$ ). In Figure 4.1, the LTS boundaries are denoted by the dashed curves. As it turns out, the LTS scheme is quite flexible and applies to various orders of flux-

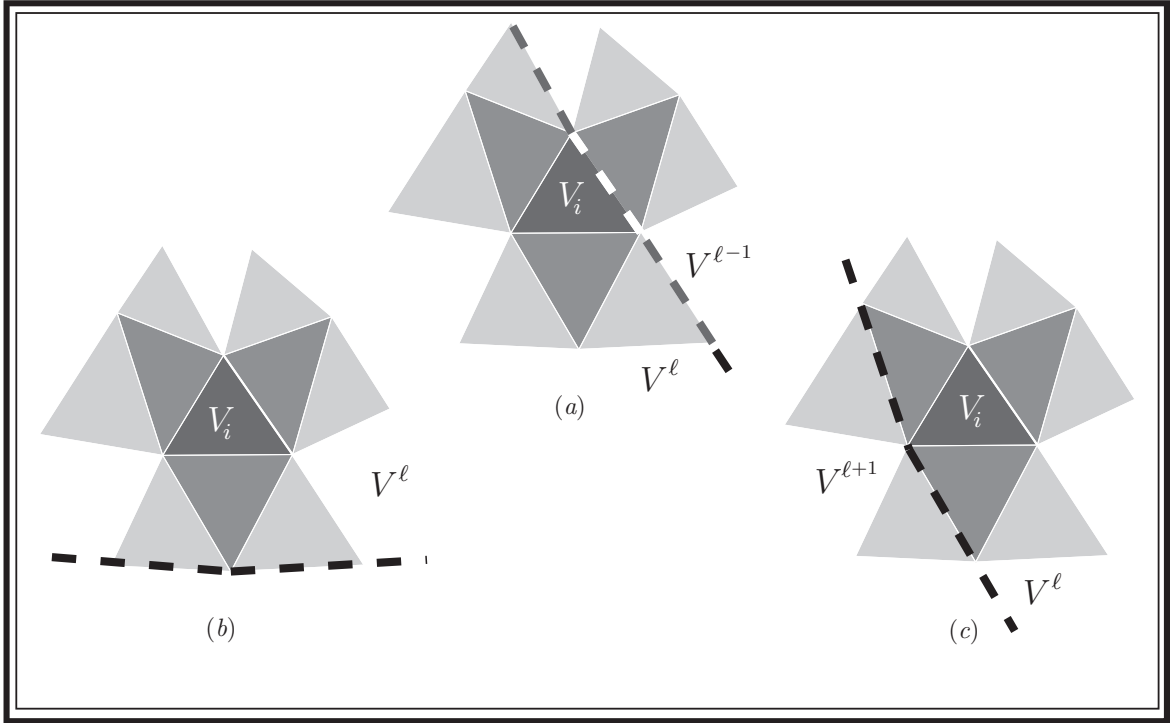


Fig. 4.2: LTS classification of an element  $V_i$  in  $V^\ell$  based on its MUSCL stencil (illustrated in 2D). The stencil is assumed to cross, at most, one LTS level boundary (denoted by the dashed curves). In  $V^\ell$  we have the following three cases: (a)  $V_i$  is an upper element, (b)  $V_i$  is a bulk element, (c)  $V_i$  is lower element

integration and time-integration schemes. It is, however, somewhat difficult to explain clearly in the general sense. On one hand, the method could simply be illustrated for a particular flux/time scheme but this hides some of the implementation details. Instead, we will approach the method generally using two assumptions:

- **LTS Assumption 1:** Given two elements  $V_i$  and  $V_j$ , if  $V_i$  is in the stencil of  $V_j$ ,  $V_j$  is in the stencil of  $V_i$ .
- **LTS Assumption 2:** The stencil of an element crosses at most one LTS boundary.

Using these two assumptions, the LTS scheme can be presented in a straightforward way. Any given element  $V_i$  in  $V^\ell$  can now be classified as one of three types as shown in Figure 4.2:

- Case (a): The stencil of element  $V_i$  in  $V^\ell$  contains elements in  $V^{\ell-1}$ .  $V_i$  is then referred to as an *upper* element of  $V^\ell$ . The collection of upper elements in  $V^\ell$  is denoted as  $Upper(\ell)$ .
- Case (b): The stencil of element  $V_i$  in  $V^\ell$  is contained entirely in  $V^\ell$ .  $V_i$  is referred to as a *bulk* element of  $V^\ell$ . The collection of bulk elements in  $V^\ell$  is denoted  $Bulk(\ell)$ .
- Case (c): The stencil of element  $V_i$  in  $V^\ell$  contains elements in  $V^{\ell+1}$ .  $V_i$  is then referred to as a *lower* element of  $V^\ell$ . The collection of lower elements is referred to as  $Lower(\ell)$ .

This classification of elements will be all that is required to perform the LTS update as discussed in Section 4.3.2. An example classification for a simple 2D mesh is shown in Figure 4.3 for first-order stencils.

The pre-processing algorithm is responsible for determining the domains  $V^\ell$  that satisfy Assumption 2. The partitioning algorithm presented in [15] is:

- Compute the local  $\Delta t_i$  for each element  $V_i$  according to the local condition (4.11).
- Compute the global  $\Delta t = \min_i \Delta t_i$ .
- Assign to each element  $V_i$  the level  $\ell_i = 1$ .
- Recursively increase the level of each  $V_i$  to  $\ell_i + 1$  provided that:

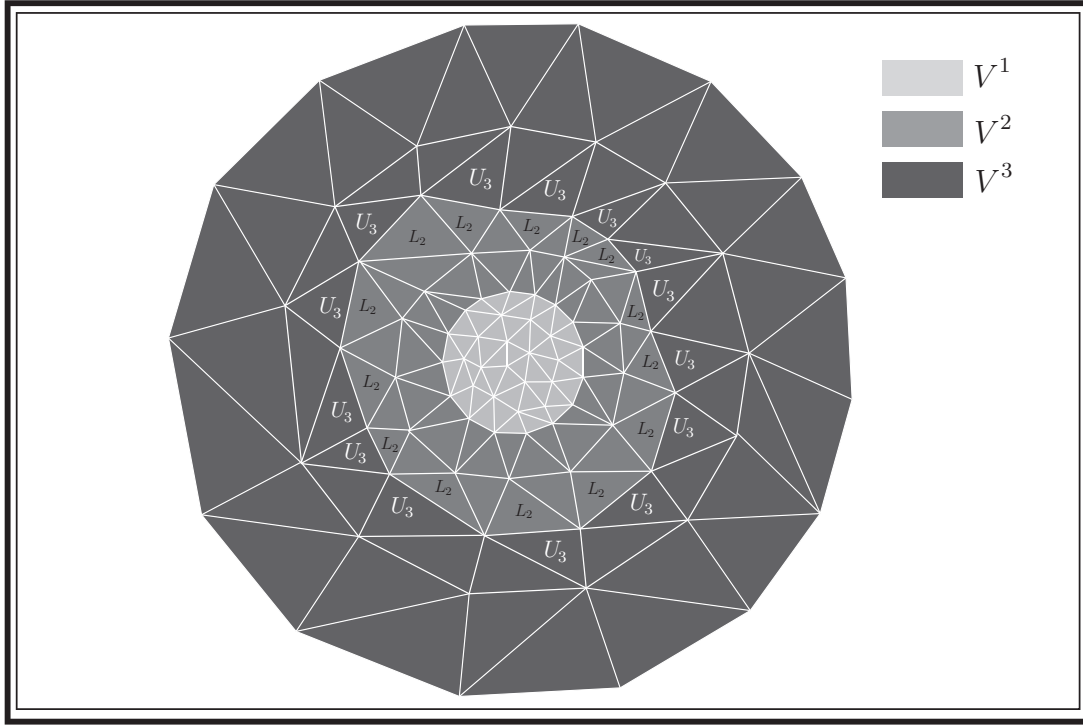


Fig. 4.3: A 3-level LTS partition of a 2D mesh. The lower elements ( $L_\ell$  in the figure) for  $V^2$  and the upper elements ( $U_\ell$  in the figure) for  $V^3$  are shown for a first-order stencil.

- $\Delta t_i > 2^{\ell_i} \Delta t$  (“local stability”)
- This local stability condition holds for all elements sharing a vertex with  $V_i$ , even though one of these elements may not be increased to the next level (“no double jumps”)
- No element sharing a vertex with  $V_i$  has a neighbour restricted by  $\Delta t_i < 2^{\ell_i-1} \Delta t$  (“disjoint boundaries”)
- Halt the algorithm once there are either no level increases for any element in  $V_i$  or some preset maximum level is reached.

Upon completion of the partitioning algorithm, a level  $\ell_i$  is associated with each element. The collection of these elements comprises  $V^\ell$ . Once  $V^\ell$  has been determined, it is straightforward to search each subdomain to determine  $Bulk(\ell)$ ,  $Lower(\ell)$  and

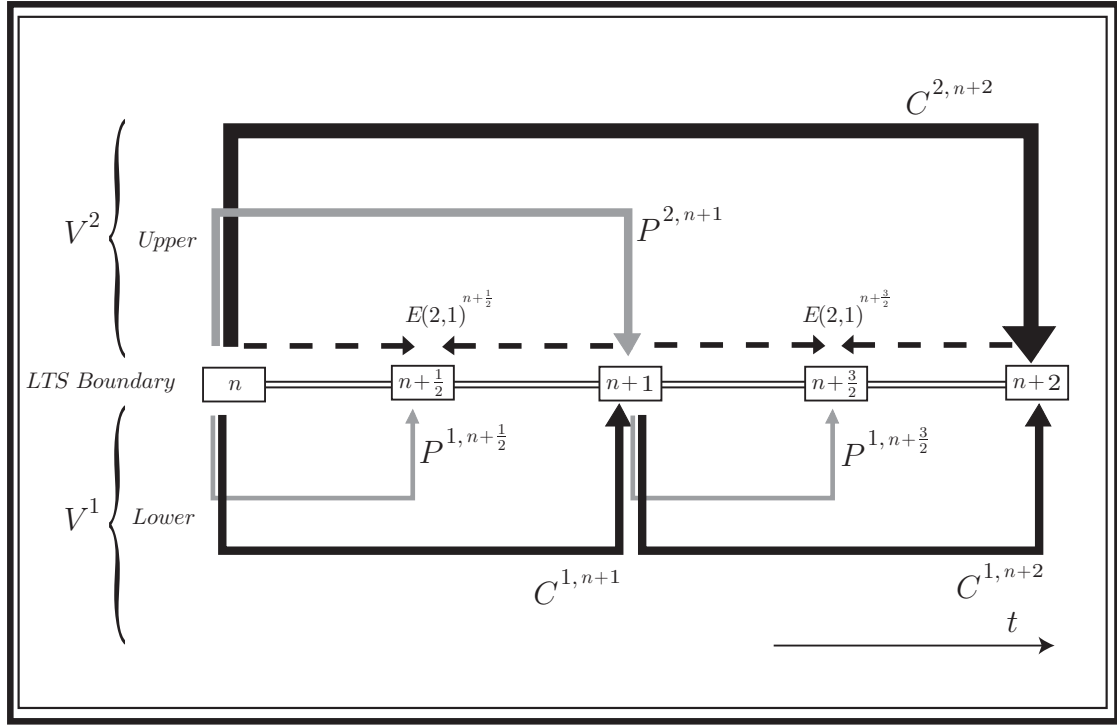


Fig. 4.4: LTS synchronization for the Predictor-Corrector scheme. Level 1 operates at  $\Delta t$ , Level 2 operates at  $2\Delta t$ . The scheme assumes that a first-order approximation is available for  $Lower(1)$  from  $Upper(2)$ .

$Upper(\ell)$ . The “disjoint-boundaries” constraint was introduced in [15] to “displace the boundaries toward larger cells”, but it also has the added benefit of ensuring that Approximation 2 holds for the MUSCL scheme. Although we do not consider it herein, we point out that by simply expanding the “disjoint boundary” condition to consider the neighbours of neighbouring cells and so on, we can ensure the classification into bulk, lower and upper elements for larger stencils required in high-order schemes. This would imply that the spatial transition from one LTS region to another would occur more slowly, but the LTS classification into bulk, lower and upper elements would still hold in each region.

## 4.3.2 The LTS Time-Evolution Schemes

The pre-processing algorithm described in the previous section produces subdomains  $V^\ell$  of  $V$  that are locally stable at time-step  $2^{\ell-1}\Delta t$ . In addition we have the classifications  $Upper(\ell)$ ,  $Bulk(\ell)$  and  $Lower(\ell)$ . We can now describe the LTS time-marching scheme, compatible with any of the explicit, global, time-marching schemes discussed in Section 4.1, and illustrated by example for the Predictor-Corrector (PC) method (4.6). We consider a domain  $V$  with two LTS levels  $V^1$  and  $V^2$ . In this way,  $V^1$  contains only bulk and lower elements while  $V^2$  contains only bulk and upper elements. We assume that the fields are initially synchronized across  $V^1$  and  $V^2$  (say at  $n = 0$ ). The fundamental time-step in  $V^1$  is  $\Delta t$  and in  $V^2$  is  $2\Delta t$ .

The general idea is shown in Figure 4.4. Level 1 (the lower part of the figure), operates at the fundamental time-step  $\Delta t$ , while level 2 (the upper portion of the figure), operates at  $2\Delta t$ . The update proceeds top-down, starting from the highest level (in this case  $\ell = 2$ ). As will be made clear shortly, the lower elements at level 1 will require a first-order estimate of the solution in the upper elements of  $V^2$  at every discrete step that  $V^1$  requires a flux integration. We denote this estimate as  $E(2, 1)^n$  which may be read as “the estimate in  $Upper(2)$  to be used by  $Lower(1)$ ” for flux evaluation at  $t = n\Delta t$ . We assume that  $E(2, 1)^n$  is initialized to the current solution for all elements in  $Upper(2)$ . Before discussing why it works, we present the algorithm:

- **Time step  $n$**  (fields assumed synchronized)
  - Compute the predictor  $P^{2,n+1}$  (denoted by a grey arrow) in  $V^2$ .  $Upper(2)$  uses the current solution in  $V^1$ .
  - Compute the predictor  $P^{1,n+1/2}$  (denoted by a grey arrow) in  $V^1$ .  $Lower(2)$  uses  $E(2, 1)^n$

- Set  $E(2, 1)^{n+1/2}$  (denoted in the figure by the dashed black arrows) equal to the average of the solution at time  $n$  and the computed predictor  $P^{2,n+1}$  in  $V^2$ .
- **Time step  $n + 1/2$** 
  - Compute the corrector  $C^{1,n+1}$  (denoted by a black arrow) in  $V^1$ . *Lower(2)* uses  $E(2, 1)^{n+1/2}$ .
  - Set  $E(2, 1)^{n+1}$  equal to the predictor  $P^{2,n+1}$  in  $V^2$ .
- **Time step  $n + 1$**  (the current solution in  $V^1$  is now second order accurate)
  - Compute the corrector step  $C^{2,n+2}$  in  $V^2$ . *Upper(2)* uses the current solution  $C^{1,n+1}$  in  $V^1$ .
  - Compute the predictor step  $P^{1,n+3/2}$  in  $V^1$ . *Lower(1)* uses  $E(2, 1)^{n+1}$
  - Set  $E(2, 1)^{n+3/2}$  equal to the average of the corrector  $C^{2,n+2}$  and predictor  $C^{2,n+1}$  in  $V^2$
- **Time step  $n + 3/2$** 
  - Compute the corrector step  $C^{1,n+2}$  in  $V^1$ . *Lower(1)* uses  $E(2, 1)^{n+3/2}$
  - Set  $E(2, 1)^{n+2}$  equal to the corrector  $C^{n+2}$  in  $V^2$  (the fully updated solution at  $n + 2$ ).
- **Time step  $n + 2$**  (all fields are second order accurate and synchronized - repeat the above steps).

The key step to understanding how this algorithm works is to realize that:

- In the general case for any of the explicit schemes under consideration, every time-step is a combination of first-order (in time) flux-evaluations.



- For  $Upper(\ell)$  we always have a first-order time approximation to the flux integral using the most recently computed (possibly intermediate) solution in  $Lower(\ell - 1)$ .
- For  $Lower(\ell)$  we can always *construct* a first-order time-approximation to the fluxes using the two most recent (possibly intermediate) solutions in  $Upper(\ell)$ .

Consequently, this algorithm easily extends to multiple levels in a straight-forward manner. Although we will not provide the details herein, it is a straight-forward task to formulate analogous LTS schemes for the RK2 and RK4 time-integration methods given in (4.8) and (4.10).

#### 4.4 Chapter Summary

In this chapter both global and local time-stepping schemes have been presented providing numerical techniques for integrating the time-derivative in the method-of-lines FVTD formulation. The LTS scheme adopted is that of Fumeaux *et. al.* [15] and details have been provided so that its functionality within the context of additional numerical features presented throughout the remainder of this work can be understood.

## Mesh Truncation

---

*We know then the existence and nature of the finite, because we also are finite and have extension. We know the existence of the infinite, and are ignorant of its nature, because it has extension like us, but not limits like us. – Blaise Pascal*

The volumetric FVTD formulation described in the previous chapters provides a rigorous mechanism for solving field problems when the computational domain  $V$  has infinite extent or is completely enclosed by perfect conductors. The numerical solution of Maxwell's field equations using FVTD requires that  $V$  must be finite. In this chapter both local and global mesh-truncation schemes – the latter implemented using Huygens' surfaces – are presented for FVTD. On one hand, local mesh truncation schemes provide a mechanism for updating  $V_i$  on the boundary  $\partial V$  from local information propagating at finite speed in the vicinity of  $V_i$ . Typically this is achieved by making assumptions about the field-behaviour on  $\partial V$ , and designing conditions to ab-

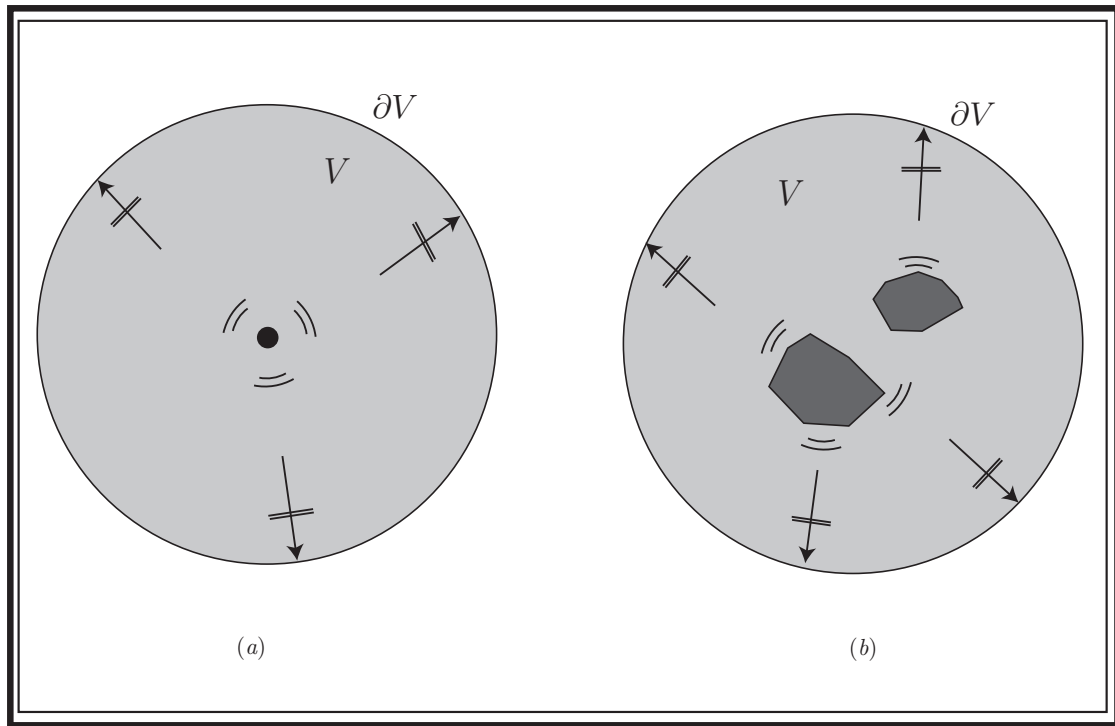


Fig. 5.1: 2D Illustration of the Silver-Müller Boundary Conditions: (a) For a point source, the fields at the boundary  $\partial V$  become spherical, or locally planar, provided  $\partial V$  is sufficiently far removed. (b) For general sources, these conditions can be met approximately provided  $\partial V$  is sufficiently far from the nearest sources.

sorb these assumed field. The benefit is that, in general, the computational overhead for these methods is low. On the other hand, when the assumed field behaviour is not satisfied, numerical reflections occur.

In contrast to local mesh-truncation, global mesh-truncation schemes avoid such assumptions at the price of increased computational effort but can be used in innovative ways to reduce the volumetric problem size.

### 5.1 Local Mesh Truncation: Silver-Müller Boundary Conditions

One major benefit of the upwind, flux-split FVTD formulation, is that the split flux-matrix provides a natural mechanism for handling incoming and outgoing propagation across the boundary of a finite-volume. This enables the implementation of the simple Silver-Müller boundary condition that assume the incoming flux is zero [9, 13]. Therefore, if volume  $V_i$  has facet  $k$  on  $\partial V$  the local absorbing boundary condition on  $\partial V_{i,k}$  is:

$$\boxed{\vec{\mathcal{F}}_{i,k|SM}^o = \bar{\mathbf{A}}_{i,k}^+ \cdot \vec{\mathbf{u}}_{i,k}^+} \quad (5.1)$$

where  $\vec{\mathbf{u}}_{i,k}^+$  is computed using the desired reconstruction scheme with a modified stencil. The accuracy of this boundary condition can be explained by assuming a single point-source located at the origin of a spherical computational domain  $V$  and by assuming that  $\partial V$  is sufficiently far from the origin that the observed fields can be approximated by spherical waves as shown in Figure 5.1 (a). As the computational boundary is spherical, the direction from the source to any point on the boundary is precisely the outer normal  $\vec{n}$  to  $\partial V$ . Consequently the assumption of field dependence along the direction  $\vec{n} = \vec{m}$  in (2.22) is validated. Therefore, by setting the incoming flux across  $\partial V$  to zero we are simply assuming that all sources of electromagnetic radiation are contained in  $V$  and that the solution to Maxwell's Equations satisfy a Sommerfeld radiation condition [31].

In the context of mesh truncation, the unstructured geometric modeling flexibility of the FVTD formulation extends beyond accurately capturing fine structural details by permitting arbitrary shapes of the domain  $V$ . For unbounded radiation problems we can choose computational domains with spherical boundaries, sufficiently removed from all radiation sources, such that the Silver-Müller boundary conditions provide ac-

curate modeling results as shown in Figure 5.1 (b). For approximate normal incidence these boundary conditions can be shown, using experiments, to provide reflection coefficients less than  $-50$  [dB] [13]. Approximate normal incidence can be achieved when the boundary is placed between one and two  $\lambda_{max}$  away, where  $\lambda_{max}$  is the maximum wavelength of interest in the FVTD simulation [13].

## 5.2 Global Mesh Truncation: Exact Radiating Boundary Conditions

A global mesh-truncation scheme reconstructs an accurate boundary flux on  $\partial V$  from the global solution of Maxwell's Equations. In this context such a boundary condition is sometimes referred to as an Exact Radiating Boundary Condition (ERBC) [32]. In the time-domain such a boundary condition can be developed using a carefully selected Huygens' surface inside the domain  $V$ .

### 5.2.1 Huygens' Surface Field Relationships

We begin by considering a source-free region  $V_\Sigma$  bounded by a surface  $\partial V_\Sigma$  as shown in Figure 5.2. As detailed in Appendix B, and as shown in [33], if in addition to being source-free  $V_\Sigma$  is stationary and homogeneous with material parameters  $\mu$  and  $\epsilon$  such that the velocity  $c = (\mu\epsilon)^{-1/2}$ , the fields at any location  $\vec{r}$  inside  $V_\Sigma$  are related to the boundary integral of the fields over the boundary  $\partial V_\Sigma$ :

$$\begin{aligned}
 \vec{\mathcal{E}}(t, \vec{r}) = & + \frac{1}{4\pi} \iint_{\partial V_\Sigma} \left( -\vec{n}' \times \left( [[\vec{\mathcal{E}} + c^{-1}R\partial_t\vec{\mathcal{E}}]] \right) \times \frac{\vec{R}}{R^3} \right. \\
 & \left. - \left( \vec{n}' \cdot \left( [[\vec{\mathcal{E}} + c^{-1}R\partial_t\vec{\mathcal{E}}]] \right) \right) \frac{\vec{R}}{R^3} + \frac{\mu}{R} \vec{n}' \times [[\partial_t\vec{\mathcal{H}}]] \right) dS'
 \end{aligned} \tag{5.2}$$

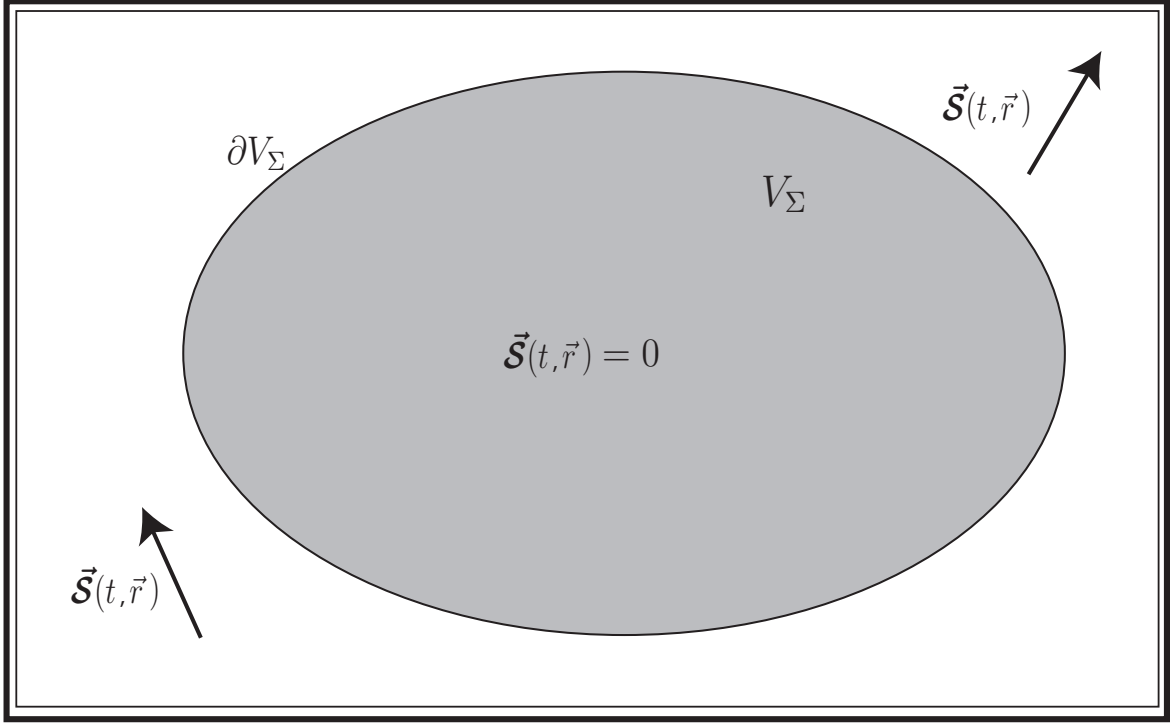


Fig. 5.2: The general geometry of a Huygens' Surface  $\partial V_\Sigma$  surrounding a source-free region  $V_\Sigma$ .

$$\begin{aligned}
 \vec{\mathcal{H}}(t, \vec{r}) = & + \frac{1}{4\pi} \iint_{\partial V_\Sigma} \left( -\vec{n}' \times \left( [[\vec{\mathcal{H}} + c^{-1}R\partial_t\vec{\mathcal{H}}]] \right) \times \frac{\vec{R}}{R^3} \right. \\
 & \left. - \left( \vec{n}' \cdot \left( [[\vec{\mathcal{H}} + c^{-1}R\partial_t\vec{\mathcal{E}}]] \right) \right) \frac{\vec{R}}{R^3} - \frac{\epsilon}{R} \vec{n}' \times [[\partial_t\vec{\mathcal{E}}]] \right) dS'
 \end{aligned} \tag{5.3}$$

In these two equations,  $\vec{n}' = \vec{n}(\vec{r}')$  is the unit outward normal to  $V_\Sigma$  on the boundary  $\partial V_\Sigma$  and the notation  $[[\cdot]]$  indicates the field quantities should be evaluated at the retarded time  $t_R = t - R/c$ . The quantity  $R = |\vec{R}| = |\vec{r} - \vec{r}'|$  is a function of both the observation point  $\vec{r}$  and the integration point  $\vec{r}'$ . The fields under the integration sign are assumed to be a function of  $\vec{r}'$ . From these equations we can obtain an exact solution at the boundary of the computational domain for unbounded field problems by considering the geometry shown in Figure 5.3. The computational domain  $V$  is divided into two regions. The first region  $V_S$ , which may be disconnected, contains all

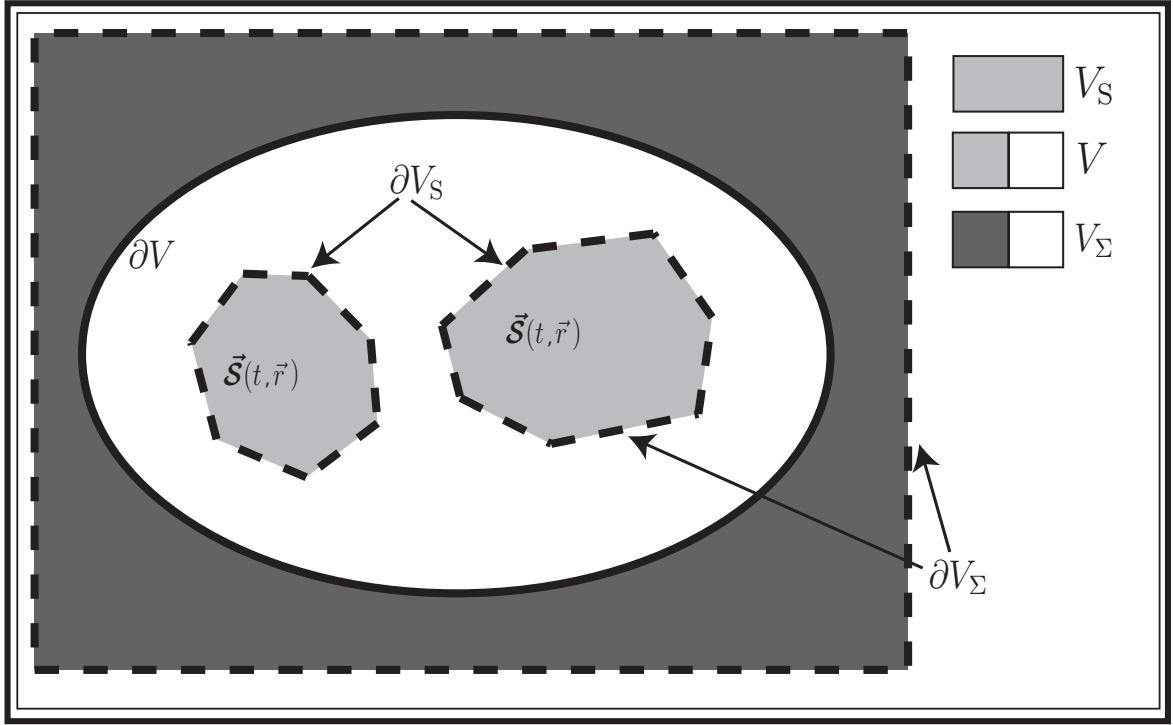


Fig. 5.3: Practical geometric configuration for incorporating a Huygens' Surface  $\partial V_\Sigma$  into the FVTD Computational Domain  $V$ . The computational boundary  $\partial V$  exists inside  $V_\Sigma$ .

field sources and inhomogeneities, and is bounded by the surface  $\partial V_S$ . In the figure, the computational domain  $V$  is both the white and light-grey regions (the latter being  $V_S$ ) and is bounded by the solid black curve. The second region  $V_\Sigma = V \setminus V_S$  exists external to  $V_S$  and is the union of the white and dark-grey regions in the figure. It extends beyond the computational domain and is bounded at infinity. Therefore, the computational boundary  $\partial V$  lies entirely in the source-free, and homogeneous region  $V_\Sigma$ . As there are no sources outside of  $V_S$  the fields at infinity are zero and the surface integrals in (5.2) and (5.3) reduce to the portion of the surface  $\partial V_\Sigma = -\partial V_S$ , where the negative sign is present to remind us that the orientation of the two surfaces oppose each other. Evaluation of these integrals can then be used to compute the fields on  $\partial V$  which lies entirely in  $V_\Sigma$ . For the remainder of this work we will simply consider

that the integral is evaluated on  $\partial V_S$  and that if the outward normal of  $\partial V_S$  is used, then the evaluation of (5.2) and (5.3) requires a multiplication by  $-1$ .

The evaluation of the surface integrals (5.2) and (5.3) require knowledge of the solution and its derivative on the surface  $\partial V_S$  at appropriate retarded times. In order to evaluate the integral the following points should be considered:

- As the FVTD solution inside  $V_S$  is given only in terms of cell-averages, we must once again reconstruct the solution on the boundary  $\partial V_S$ .
- A mechanism for determining (approximating) the time-derivatives of the solution is required.
- We will be required to interpolate the time-solutions. For a given pair of points  $\vec{r} \in \partial V$ , and  $\vec{r}' \in \partial V_S$ , the retarded time  $t_r = t - |\vec{r} - \vec{r}'|/c$  is unlikely to be equal to  $n\Delta t$  for some time-step index  $n$ .
- The evaluation of the surface integrals should be addressed in the context of the local time-stepping (LTS) scheme presented in Section 4.3.

All of these issues are dealt with in what follows.

### *5.2.2 Spatial Reconstruction and Integration*

We assume that the surface  $V_S$  is made up of facets of elements in  $\{V_i\}$ . Therefore,  $V_S$  is comprised of first-order surface segments as shown in Figure 5.4. For the moment we will assume that the fields are known continuously in time. Computing the field at location  $(t, \vec{r})$  on  $\partial V$  can then be performed to the same spatial accuracy as the volumetric solution by numerically integrating the reconstructed fields (and their



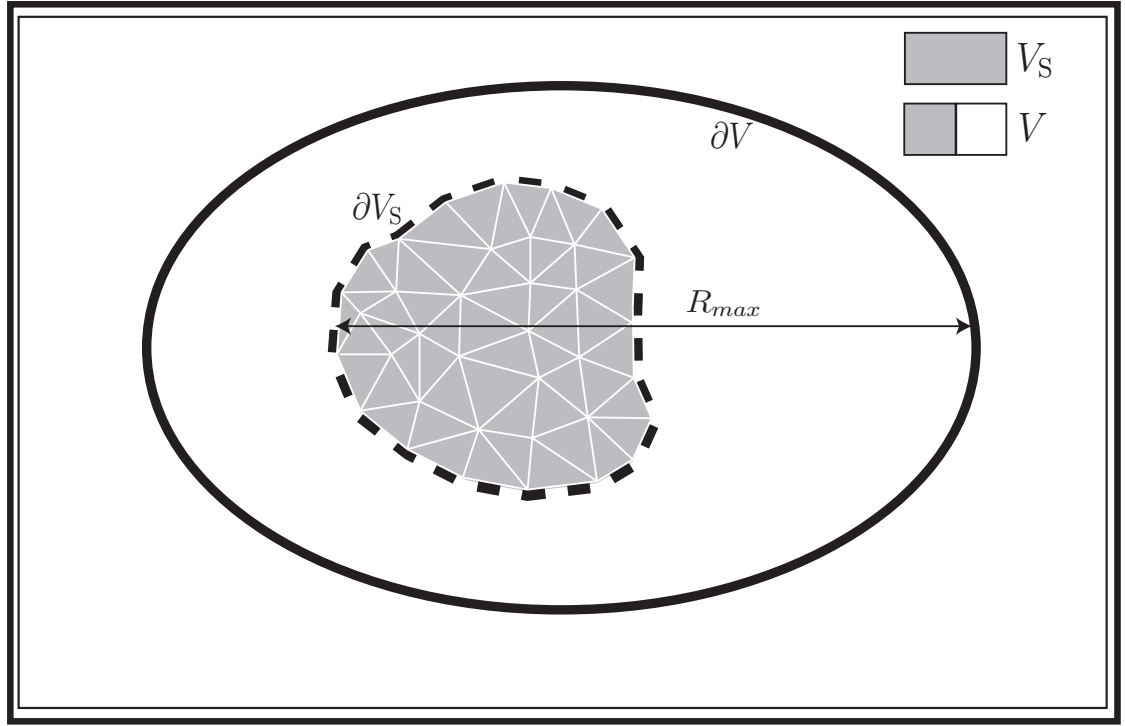


Fig. 5.4: A 2D representation of a global mesh truncation integration surface. The surface  $\partial V_S$  is assumed to be aligned with the facets of the volumetric mesh. Cells external to  $\partial V_S$  are not shown.

derivatives) on each facet  $\partial V_{i,k} \in V_S$  at time  $t - R/c$ . The reconstructed fields can be obtained using the same reconstruction procedure as the underlying FVTD solution. The integration can also be performed using the same scheme as the volumetric flux integration. In this way, the spatial accuracy of the FVTD scheme is preserved provided that we assume that nearest observation point  $\vec{r}$  is far enough from  $V_S$  that the negative powers of  $R = |\vec{r} - \vec{r}'|$  appearing in the integrals (5.2) and (5.3) are sufficiently smooth that the numerical integration is accurate. If this is not the case, singularity extraction techniques or customized numerical integration rules may be required to accurately compute the fields at  $(t, \vec{r})$  on  $V_S$ .

## 5.2.3 Temporal Interpolation, Time-Derivatives and Local Time-Stepping

When computing the flux on the boundary  $\partial V$  of the computational domain at time  $n\Delta t$ , we require knowledge of the solution and its time-derivatives on  $V_S$  at times  $n\Delta t - R/v$ . This retarded time will not, in general, correspond to times when the values of the solution are known. In the absence of local time-stepping, a time-interpolation scheme that is compatible with the temporal accuracy of FVTD running at the fundamental time-step  $\Delta t$  can be developed as follows.

First we compute  $R_{max}$  as the maximum distance between a flux evaluation point  $\vec{r}$  on  $\partial V$  and an integration point  $\vec{r}'$  on  $\partial V_S$  as shown in Figure 5.4. The maximum propagation delay between points on  $\partial V_S$  and  $\partial V$  is denoted by  $t_{r,max}$  and is defined as:

$$t_{r,max} \triangleq \frac{R_{max}}{c} \quad (5.4)$$

Assuming that we are interested in evaluating the surface integrals at time  $n\Delta t$  for position  $\vec{r}$  on the computational boundary we assume that the solution on  $\partial V_S$  is stored for  $M$  time-steps prior to  $n$ , where  $M$  is defined as

$$M \triangleq \left\lceil \frac{R_{max}}{c\Delta t} \right\rceil \quad (5.5)$$

Now, for two arbitrary positions  $\vec{r} \in \partial V$  and  $\vec{r}' \in \partial V_S$  as shown in Figure 5.4, we compute the required retarded time  $t_r = t - |\vec{r} - \vec{r}'|/c \leq t_{r,max}$ . This retarded time is used to produce an index  $m$ :

$$m = \left\lfloor \frac{|\vec{r} - \vec{r}'|}{c\Delta t} \right\rfloor \leq M \quad (5.6)$$

such that the solution at  $(t_r, \vec{r}')$  is between the known stored samples  $(n - m)\Delta t$

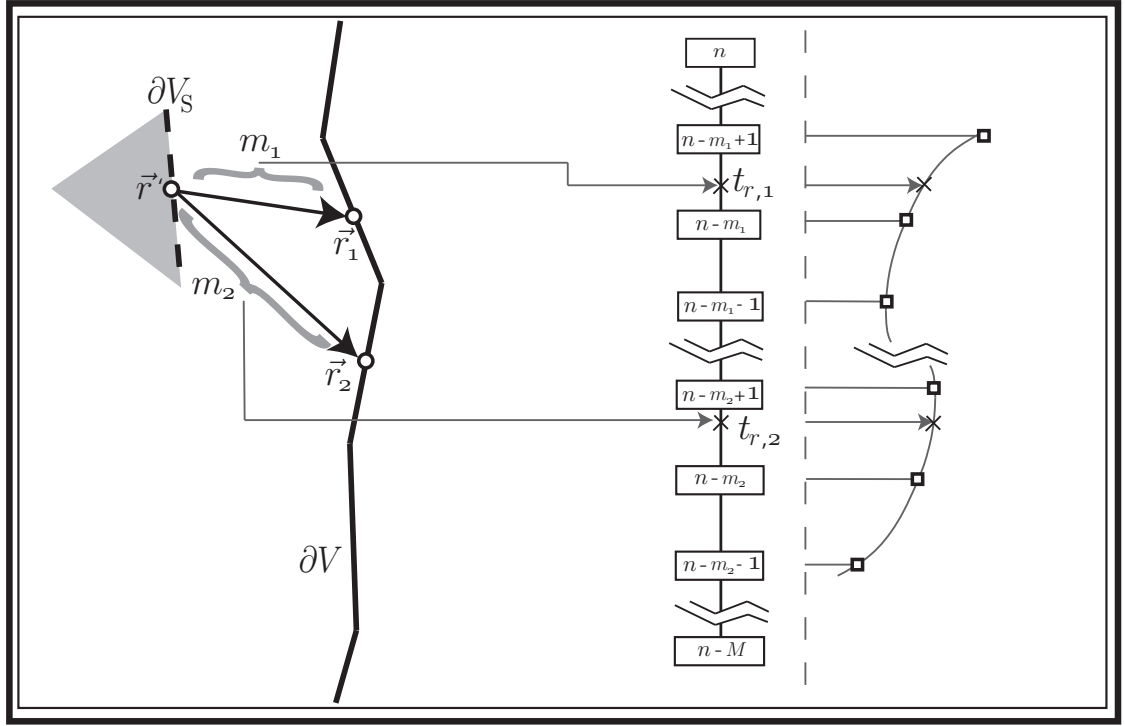


Fig. 5.5: Time interpolation in the global mesh truncation scheme. For the source point  $\vec{r}'$ , and two boundary points  $\vec{r}_1$  and  $\vec{r}_2$ , the retarded times  $t_{r,1}$  and  $t_{r,2}$  respectively decode to indices  $m_1$  and  $m_2$ . The solution at  $(m_1\Delta t, \vec{r}')$  and its nearest neighbours are fit using a polynomial function.

and  $(n - m + 1)\Delta t$  as shown for two different observation positions  $\vec{r}_1$  and  $\vec{r}_2$ , with corresponding indices  $m_1$  and  $m_2$  in Figure 5.5. Once the appropriate index,  $m$ , has been determined the time-samples in the vicinity of  $m$  can be used to fit a polynomial of degree corresponding to the time-marching accuracy. Once the polynomial coefficients are computed, the derivative of the solution at  $t_r$  is easily computed by differentiating the polynomial. Using this time-interpolation scheme poses no issues when the local time-stepping scheme presented in Section 4.3 is used. In fact, the integration surface  $\partial V_S$  can cross any number of time-step boundaries. All that is required is that the solution at the retarded time is interpolated properly.

## 5.2.4 Incorporation of a Huygens' Surface in FVTD

The evaluation of (5.2) and (5.3) over  $V_S$  can be used to evaluate the fields at the current time-step on the boundary  $\partial V$  of the computational domain which, by definition, exists inside  $V_\Sigma$ . Empirically we have found, however, that this cannot be done in an adhoc manner. For example, we cannot use the integrals to compute both  $\vec{U}_{i,k}^+$  and  $\vec{U}_{i,k}^-$  on a facet  $\partial V_{i,k} \in \partial V$ . Instead we have found that stability is maintained if we compute  $\vec{U}_{i,k}^+$  using the local upwinding methods described in Section 3.7, while we compute  $\vec{u}_{i,k}^- = \vec{u}_{i,k}^{\text{ERBC}}$  by evaluating the integrals (5.2) and (5.3). Although this mesh truncation scheme was developed independently, it ends up being similar to the way that stable ERBCs are implemented for FDTD [32]. The global boundary flux can be computed as:

$$\boxed{\vec{\mathcal{F}}_{i,k|ERBC}^o = \bar{\mathbf{A}}_{i,k}^+ \cdot \vec{u}_{i,k}^+ + \bar{\mathbf{A}}_{i,k}^- \cdot \vec{u}_{i,k}^{\text{ERBC}}} \quad (5.7)$$

While a rigorous proof of the stability of this scheme is beyond the scope of this work, we can offer some motivation for its functionality. The Huygens integrals use global field information to translate fields through time and space. This is in contrast to the local evaluation of the fields in which approximation errors appear, and travel, locally. Upwinding using the two solutions then acts to mitigate the numerical tension between them.

## 5.2.5 Global Mesh-Truncation Complexity and Volumetric Size Reduction

Using the global mesh-truncation scheme presented in this section requires the evaluation of the surface integrals (5.2) and (5.3) on  $\partial V_S$  for each flux evaluation point on  $\partial V$ . Given  $N_{\partial V_S}$  facets on  $\partial V_S$  and  $N_{\partial V}$  facets on  $\partial V$  the computational complexity

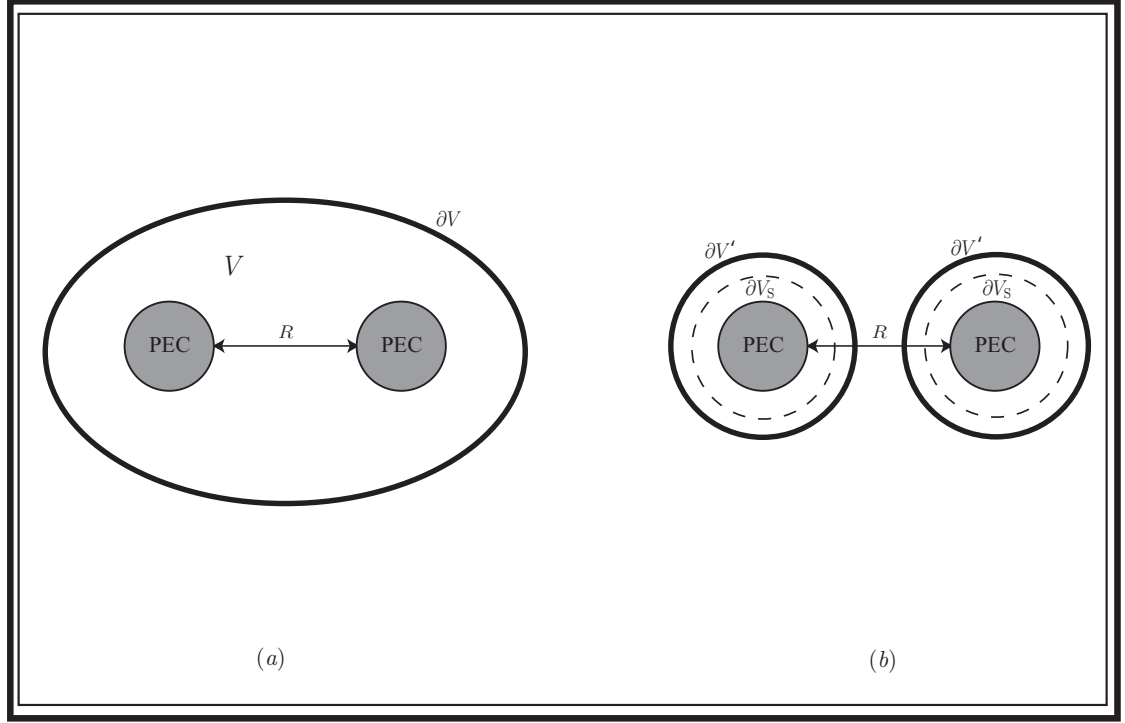


Fig. 5.6: Problem size reduction using global mesh truncation. In the case of a scattered-field formulation and two PEC spheres, the local mesh truncation scheme requires the problem to be formulated as shown in (a) with a single domain  $V$ . The two PEC spheres act as sources and can be enclosed by a disconnected surface  $\partial V_S$ . Computation of the incoming fluxes on  $\partial V$  can be computed directly from values on  $\partial V_S$  permitting a reduction in the volumetric problem size (b).

scales as  $O(N_{\partial V_S} N_{\partial V})$ . The additional memory required to back-store  $M$  solutions on  $\partial V_S$  scales as  $O(N_{\partial V_S} M)$ . While the additional costs associated with the global mesh truncation scheme are in addition to the  $O(N_V)$  time and memory required to run FVTD simulations with local mesh truncation schemes, the global truncation scheme can, for some problems, be used to reduce the overall volumetric discretization. Consider, for example, the scattering problem of two PEC spheres separated by a distance  $R$  as shown in Figure 5.6. When only the local mesh-truncation scheme is considered, the configuration (a) is required to determine the scattered field inside  $V$ . By enclosing the inhomogeneities and sources (the PEC spheres) by a closed surface  $\partial V_S$  we can reduce the computational domain to  $V'$ , bounded by  $\partial V'$  as shown in configuration

(b). If field computations at locations inside  $V \setminus V'$  are desired, the surface integrals over  $\partial V_S$  can be used. There is, however, a limit to the applicability of this approach. If, for example, the distance  $R$  separating the spheres in Figure 5.6 became increasingly large, the maximum retarded time index  $M$  would grow accordingly and the back-storage of the required retarded solutions would bottleneck the algorithm. Of course, this bottleneck would be far less stringent than attempting to discretize the entire computational domain.

### 5.3 Chapter Summary

In this chapter two techniques for truncating the computational domain  $V$  have been presented. The first, a local truncation scheme using the Silver-Müller boundary conditions, depends on the field values in the vicinity of the computational boundary  $\partial V$  at the current simulation time. They are easily implemented and do not incur any computational overhead. The second, a global truncation scheme using a Huygens' surface for computing the incoming boundary fluxes on  $\partial V$ , requires additional time and memory but has the benefit that the boundary of the computational domain can be arbitrarily shaped and disconnected.

Combining either of these mesh truncation schemes with the upwind, flux-split, time-marching FVTD schemes presented to this point, completes the discussion of a volumetric FVTD code. Implementation details and numerical validation are the subject of the next chapter.

One possible mesh-truncation scheme that has not been mentioned is the FVTD perfectly-matched layer developed in, for example, [34]. This method will be investigated in the future.

# Implementation and Validation of the FVTD Volumetric Field Solver

---

*Computers are useless. They can only give you answers.* – Pablo Picasso

In this chapter we are concerned with the validation of the volumetric FVTD field solver, including the local time-stepping (LTS) scheme and global mesh-truncation. The validity of these algorithms will be presented for the simple case of scattering from a PEC sphere as an analytic solution is available for comparison [35].

## 6.1 Implementation

The upwind and flux-split FVTD numerical engine has been implemented in the C++ programming language providing a versatile object-oriented approach to handling the

larger subdivisions of the code [26]. For core geometric and time-stepping computations, the paradigm presented in [26] is replaced with C-style structures and C-code, permitting greater computational efficiency at the cost of slightly less compact code. Beyond this general description, coding-specific implementation details will not be provided. For the serial implementation, no third-party libraries are used. All mathematical operations, mesh representation and FVTD algorithmic components have been implemented (in some cases as libraries themselves) as part of this work.

For finite-volume mesh generation we use *Gmsh* [36] which is a free software tool, available for download online [37]. All post-process visualization is performed using ParaView [38], again a free tool available at [39].

## 6.2 Volumetric Scattering from a PEC Sphere

The scattering problem under consideration consists of a PEC sphere of radius 1 [m], centered at the origin within an 8 [m] radius computational domain assumed to be free-space, truncated with local Silver-Müller boundary conditions. The volume internal to the PEC is not introduced in the mesh as the closed PEC volume is completely described by the appropriate flux-reconstruction conditions given in Section 3.6.3. Two mesh discretizations are used:

- **Mesh M1**, having roughly 39,000 tetrahedral volumes and 630 elements discretizing the PEC surface. For this mesh,  $\Delta t$  is computed as  $6.3975 \times 10^{-11}$  [s]. A rough geometric approximation suggests that the mesh supports frequencies up to 50 [MHz] at 10 cells per wavelength.
- **Mesh M2**, contains approximately 532,000 tetrahedral elements and 1200 PEC



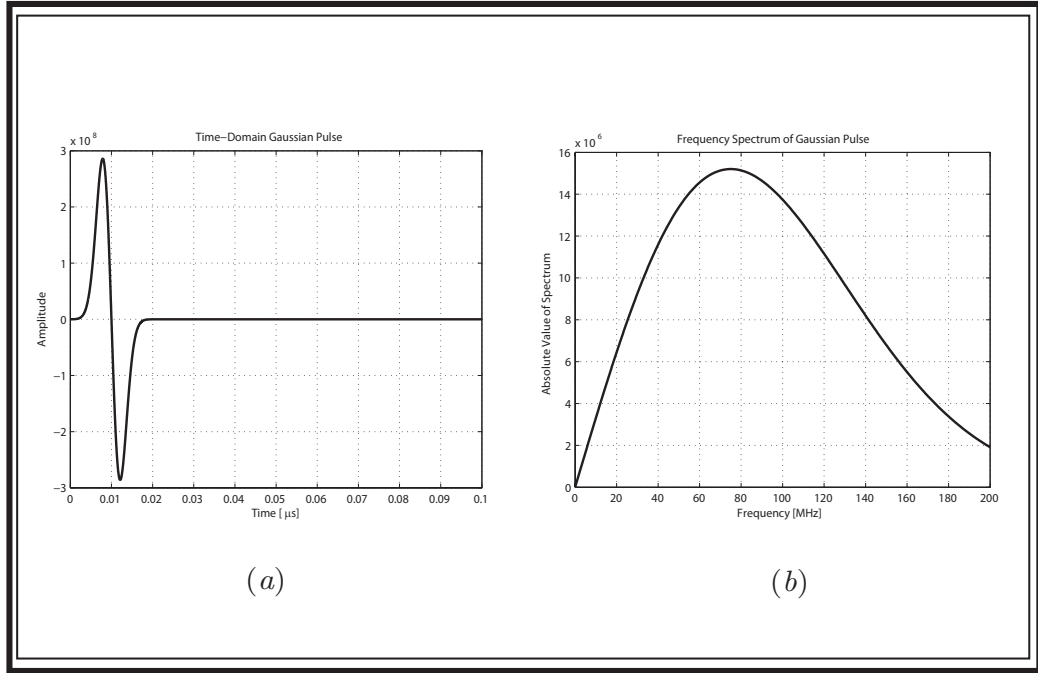


Fig. 6.1: Gaussian sources for volumetric PEC scattering. (a) Time-domain pulse with mean  $1 \times 10^{-8}$  [s] and deviation  $3 \times 10^{-9}$  [s]. (b) Frequency spectrum of the time-domain pulse showing a frequency range up to 180 [MHz]

surface elements. The time-step is  $\Delta t = 4.47 \times 10^{-11}$  [s] and the mesh supports frequencies up to 150 [MHz] at 10 cells per wavelength.

For both meshes, the FVTD solution was computed using MUSCL reconstruction and predictor-corrector time-stepping. The incident electromagnetic field was selected as a  $z$ -directed,  $x$ -polarized plane-wave, time-varying as the derivative of a Gaussian pulse with mean  $1 \times 10^{-8}$  [s] and standard deviation  $3 \times 10^{-9}$  [s]. The pulse and its spectrum are shown in Figure 6.1 indicating that the plane-wave carries frequencies beyond 180 [MHz]. The resulting scattered-field is stored over time at the back-scattered location  $(0, 0, -5)$  and its frequency response is computed by a Fast Fourier Transform (FFT) so that it can be compared to the analytic solution which is more readily available in the frequency domain. Although the analytic time-domain solution can be obtained by

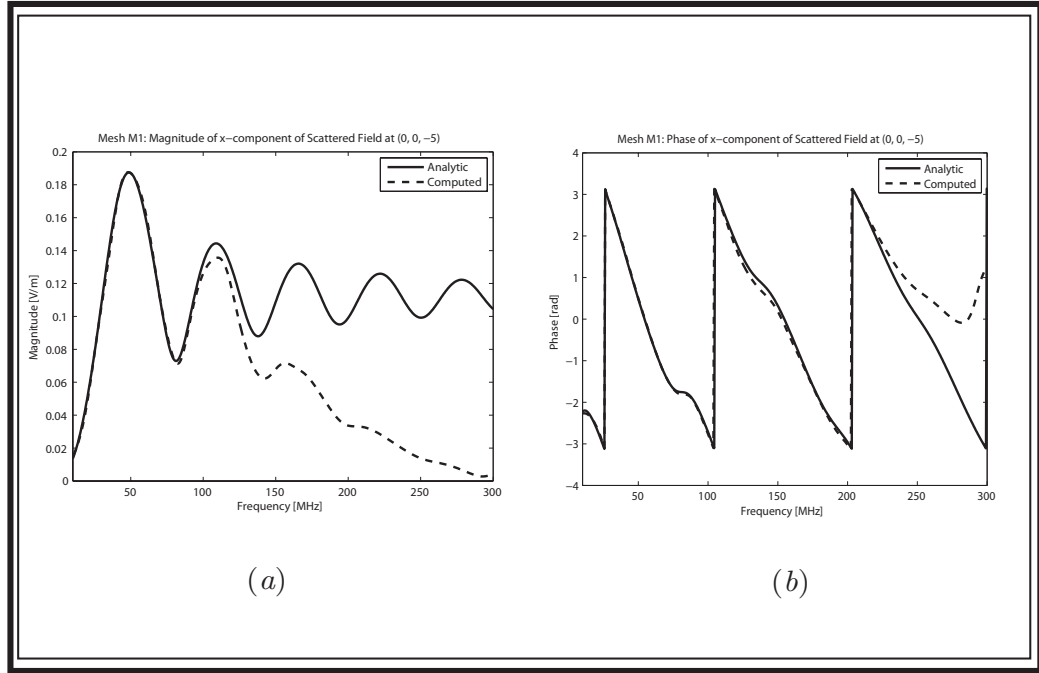


Fig. 6.2: FVTD Solution to Scattering from a PEC Sphere for the  $M1$  mesh. Back-scattered frequency spectrum of the dominant  $x$ -component at  $(0, 0, -5)$ . (a) Magnitude (b) Phase.

taking the inverse FFT (IFFT) of the analytic frequency-domain solution, it is more insightful to compare frequency-domain results directly. The results for the mesh  $M1$  are shown in Figure 6.2, while  $M2$  results are shown in Figure 6.3. Beginning with the  $M1$  mesh in Figure 6.2 we see excellent agreement in the magnitude of the scattered-field up to roughly 100 [MHz]. Further, the phase is reconstructed accurately up to about 200 [MHz]. This is an excellent depiction of the dissipative nature of upwind FVTD, where errors manifest themselves as a loss of amplitude, visible in the figure above 100 [MHz]. This is unlike FDTD where the errors are dispersive and can be expected to show themselves in the phase.

Comparing the results of the coarser  $M1$  mesh to the results of the finer  $M2$  mesh we see, for the finer discretization in Figure 6.3, that the amplitude is converging to the analytic solution up to 150 [MHz], approximately the frequency limit of the mesh.

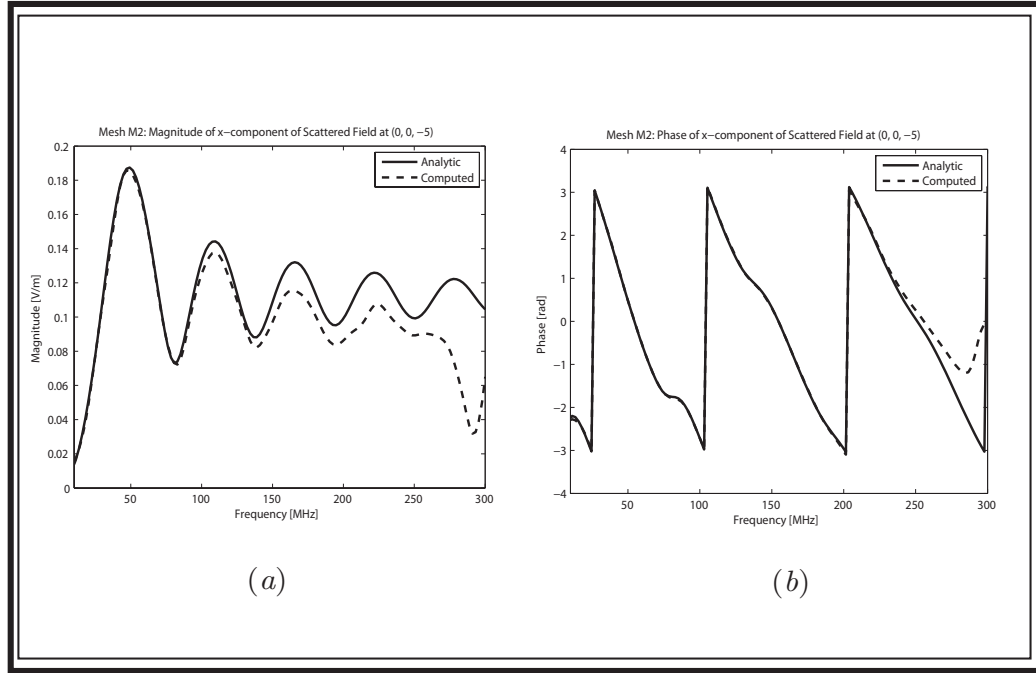


Fig. 6.3: FVTD Solution to Scattering from a PEC Sphere for the  $M2$  mesh. Back-scattered frequency spectrum of the dominant  $x$ -component at  $(0, 0, -5)$ . (a) Magnitude (b) Phase.

The phase reconstruction is, in general, more accurate. Finally, in Figure 6.4 we show an example of the magnitude of the electric (top) and magnetic (bottom) scattered fields after 400 time-steps. The figures show the  $x$ - $z$  cross-section of the computational domain at  $y = 0$ .

### 6.3 LTS Validation

In order to validate the local time-stepping (LTS) scheme introduced in Section 4.3, we consider a mesh similar to the  $M2$  mesh. The difference is that the  $M2$  mesh was optimized (by Gmsh) for “element quality” which, in turn, improves the time-step on each element. In fact, in the  $M2$  mesh, the minimum permissible time-step is  $4.4669 \times 10^{-11}$  [s] while the maximum permissible time-step is  $2.56 \times 10^{-10}$  [s], less than

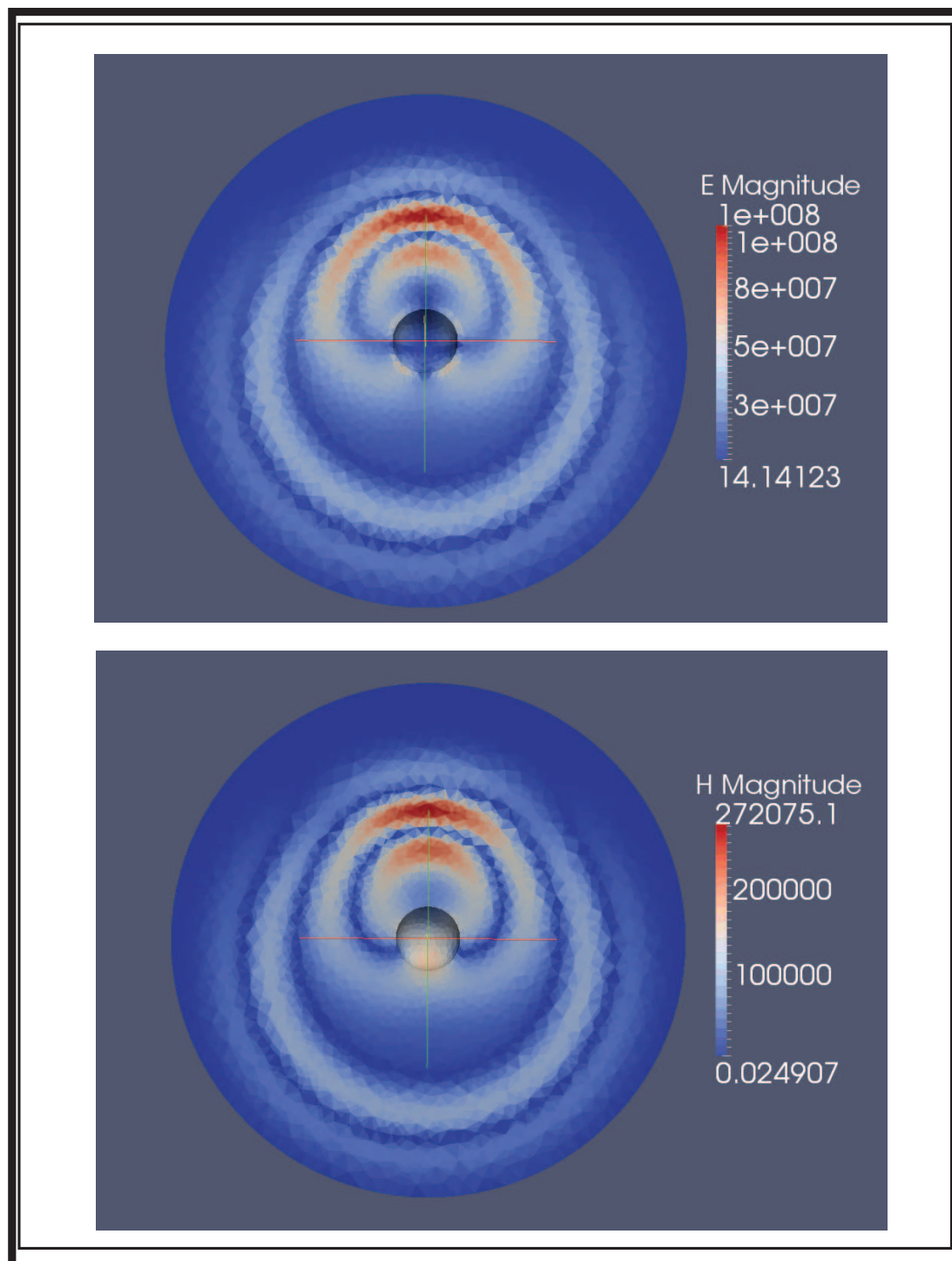


Fig. 6.4: (Colour) Volumetric time-domain scattered-field visualization. The magnitude of the electric (top) and magnetic (bottom) fields are shown at 400 time-steps.

four times larger. For the  $M2$  mesh, only two time-levels would be allowed. Therefore we consider a new mesh,  $M2_{no-opt}$  having the same discretization parameters as the  $M2$  mesh but not subjected to a Gmsh optimization. For the  $M2_{no-opt}$  mesh, the minimum local time-step is  $4.554 \times 10^{-13}$  [s] while the maximum local time-step is  $1.187 \times 10^{-10}$  [s] resulting in 8 time-stepping levels having, in order of increasing level, approximately 300, 3000, 20000, 80000, 280000, 222000, 40000 and 200 volumetric elements. A cross-section of the volume and the associated LTS levels and boundaries are shown in Figure 6.5. The back-scattered observation point  $(0, 0, -2)$  lies in the 6th time-step level and Figure 6.6 shows the  $x$ -component of the scattered electric field for the  $M2$  (no LTS) and  $M2_{no-opt}$  (LTS) meshes. Excellent agreement between the two algorithms is demonstrated. The closeup shows that the observation point, at level 6 in the LTS scheme, is constant while the lower levels update. While in practice we would likely never run a simulation on an un-optimized mesh, this example is analogous to a problem where very fine discretizations are required to capture geometric features. The speedup observed using the LTS scheme for the  $M2_{no-opt}$  mesh was a factor of 4.6. This is in the published range of speedups in [15], which range over 3.7-11.4 for a variety of geometries therein.

### 6.4 Global Mesh Truncation Validation

To conclude the validation results of this chapter, we demonstrate the functionality of the global mesh-truncation scheme presented in Section 5.2. We consider the interesting geometry depicted in Figure 6.7: two disconnected volumetric regions, coupled by the integrals over the Huygens-surface (shown in red - top). The integration surface encloses the PEC sphere (yellow) and can be used to provide the incoming fluxes on the boundary of the computational domain (blue - top). This provides a way of in-

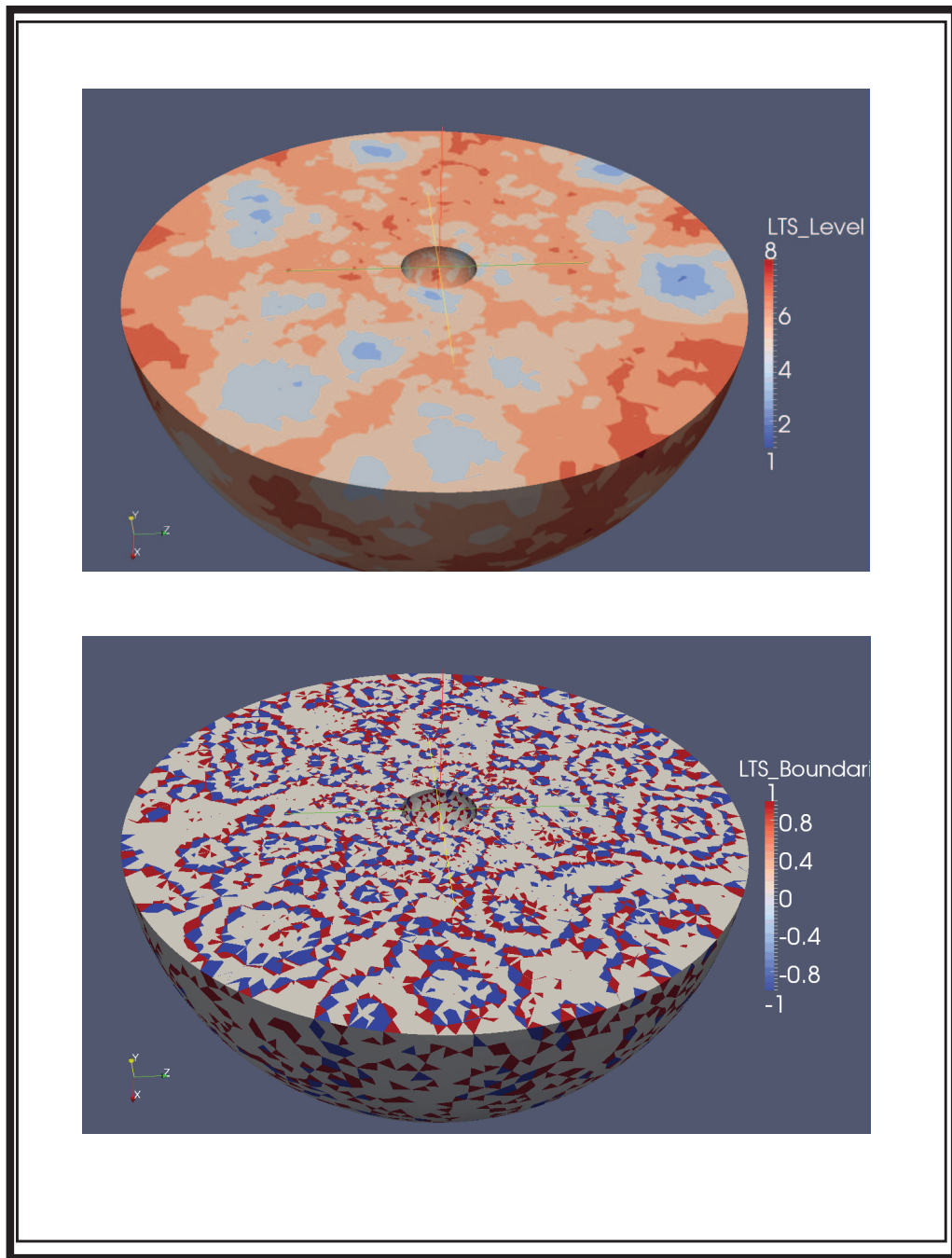


Fig. 6.5: (Colour) LTS levels of the unoptimized  $M2_{no-opt}$  mesh. The levels (top) and boundaries (bottom) are shown for a cross-section of the computational domain. For the boundaries, 1 denotes elements on the upper side, while  $-1$  denotes lower boundary elements.

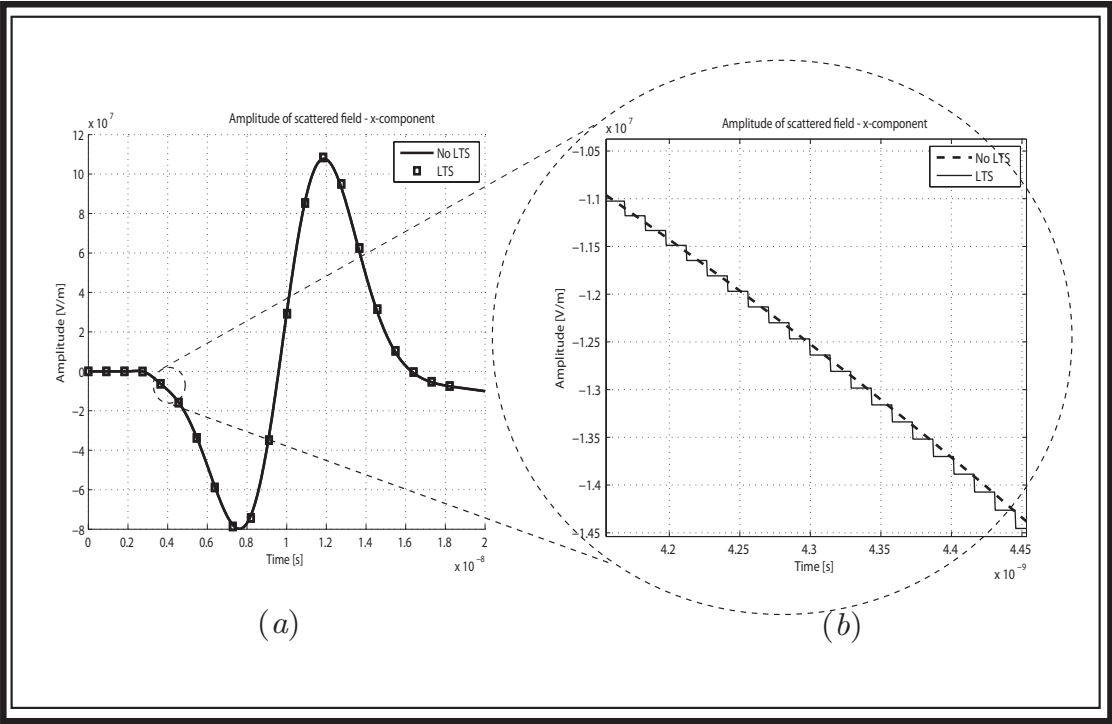


Fig. 6.6: Comparison of LTS and non-LTS solutions for the time-domain scattered field obtained using the  $M2$  and  $M2_{no-opt}$  mesh. (a) The  $x$ -component of the electric field is shown at the back-scatter point  $(0, 0, -2)$ . (b) When the LTS fields are stored at each time-step corresponding to level 1, we see that the the observation point, at LTS level 6, is constant while the lower levels update.

cluding the source-free volumetric region (blue - bottom) into the FVTD simulation. While we could have included sources (also surrounded by an integration surface) in the blue volumetric region, we have not. Without them we are still just solving for the scattering of a PEC sphere and an analytic solution can be used to verify the fields in the disconnected volume.

The geometry shown in Figure 6.7 is summarized as follows: the inhomogeneous volume consists of a 1 [m] radius PEC sphere, centered at the origin, inside a 5 [m] radius computational domain. The integration surface is 3 [m] is radius, centered on the PEC sphere. The secondary, source-free volume, is centered at  $(0, 0, -8)$  [m] and has a radius equal to 1 [m]. We again consider an  $x$ -polarized,  $z$ -propagating plane-wave. The scattered-field is measured at the centre of the secondary sphere at  $(0, 0, -8)$  [m], in the back-scattering direction of the incident field. The mesh discretization consists approximately of 500,000 volumetric elements (supporting a maximum frequency of around 100 [MHz]), 600 PEC surface elements and 6,300 integration surface elements. The maximum distance between the integration and boundary surfaces is 11 [m], and at the computed global time-step of  $3.591 \times 10^{-11}$  [s], requires a maximum back-storage of 1100 time-samples. Currently, the code does not support a different number of time-samples per integration surface patch. While this improvement will be made in the future, we currently require  $1,100 \times 6,300$  samples of the fields (6 components) to support the integration surface data. Further, the computational domain is bounded by approximately 2,700 facets. At each (partial) time-step, the global truncation scheme requires the integration over 6,300 patches for each of these 2,700 boundary surface elements, plus the appropriate interpolation of the time-solution (and its derivatives) on the surface  $\partial V_S$ .

The incident field time variation is again chosen to be the derivative of a Gaussian



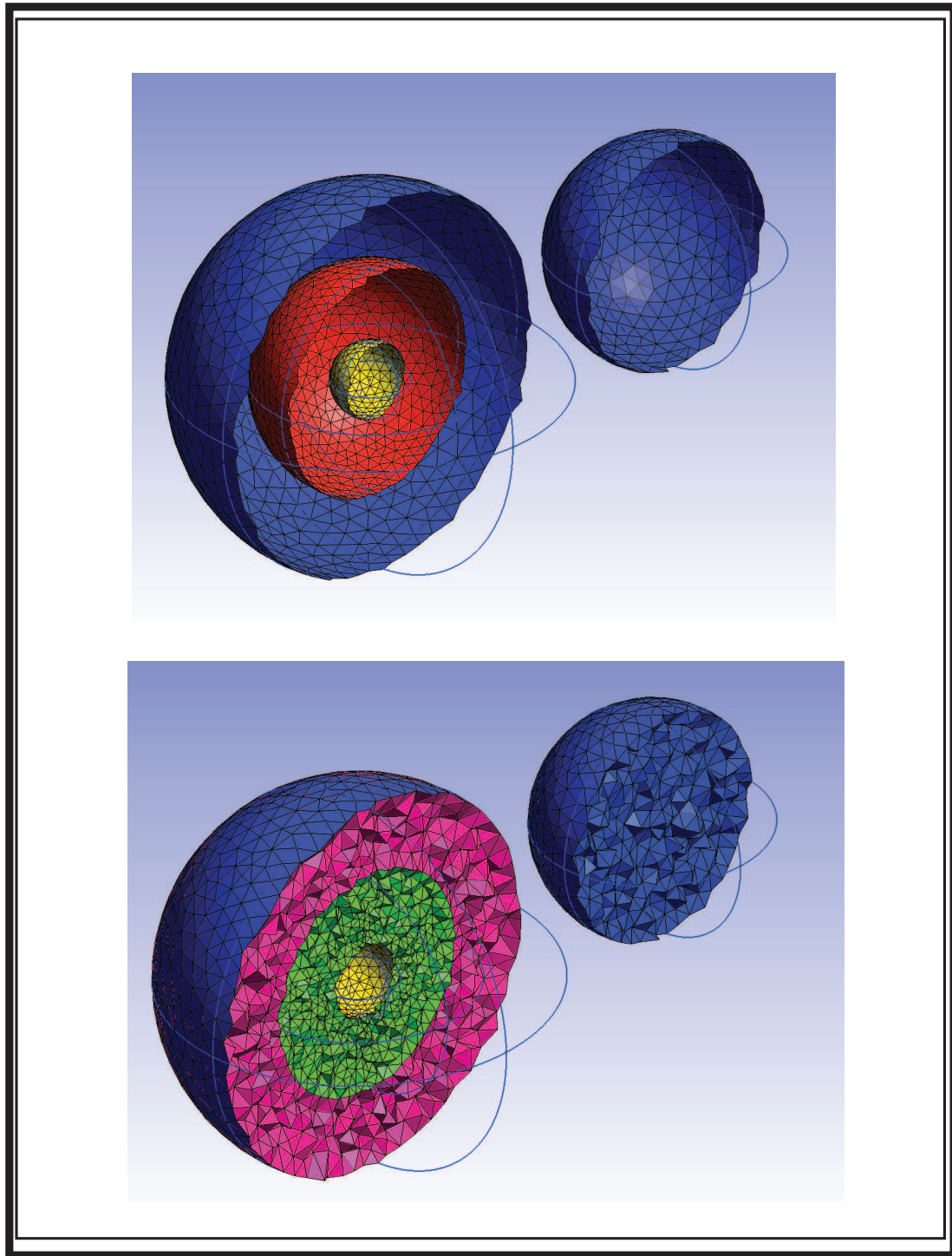


Fig. 6.7: (Colour) Sample geometry used for the validation of the global mesh-truncation scheme. A PEC sphere (yellow surface - top) is enclosed by an integration surface  $\partial V_S$  (red - top). The surface separates the free-space volumes internal to  $\partial V_S$  (green-bottom) and external external to  $\partial V_S$  (fuchsia and blue - bottom) . Using the fields computed by integration over this surface, the disconnected volume (blue - bottom) can be included in the FVTD simulation.

pulse, this time with mean  $2 \times 10^{-8}$  [s] and standard deviation  $3 \times 10^{-9}$  [s]. In Figure 6.8 we show the propagation of the scattered field through the disconnect domains.

In Figure 6.9 we compare the  $x$ -component of the electric field to the analytic solution in the disconnected volume at  $(0, 0, -8)$  [m] up to 100 [MHz]. The magnitude of the  $x$ -component of the scattered field is shown in (a), while the phase is shown in (b). The phase comparison is excellent. The magnitude diverges slightly from the analytic solution but is quite reasonable – sufficient, at least, to validate the global mesh truncation scheme.

## 6.5 Chapter Summary

In this chapter the volumetric features of an upwind and flux-split numerical FVTD engine have been validated for scattering from a PEC sphere, including local time-stepping and global mesh-truncation. Although we have not shown results for problems involving dielectrics, the FVTD engine is also accurate in those cases. For example, the code is currently being used to simulate dielectric and conductive scattering in multi-layered media for sea-ice modeling [40].

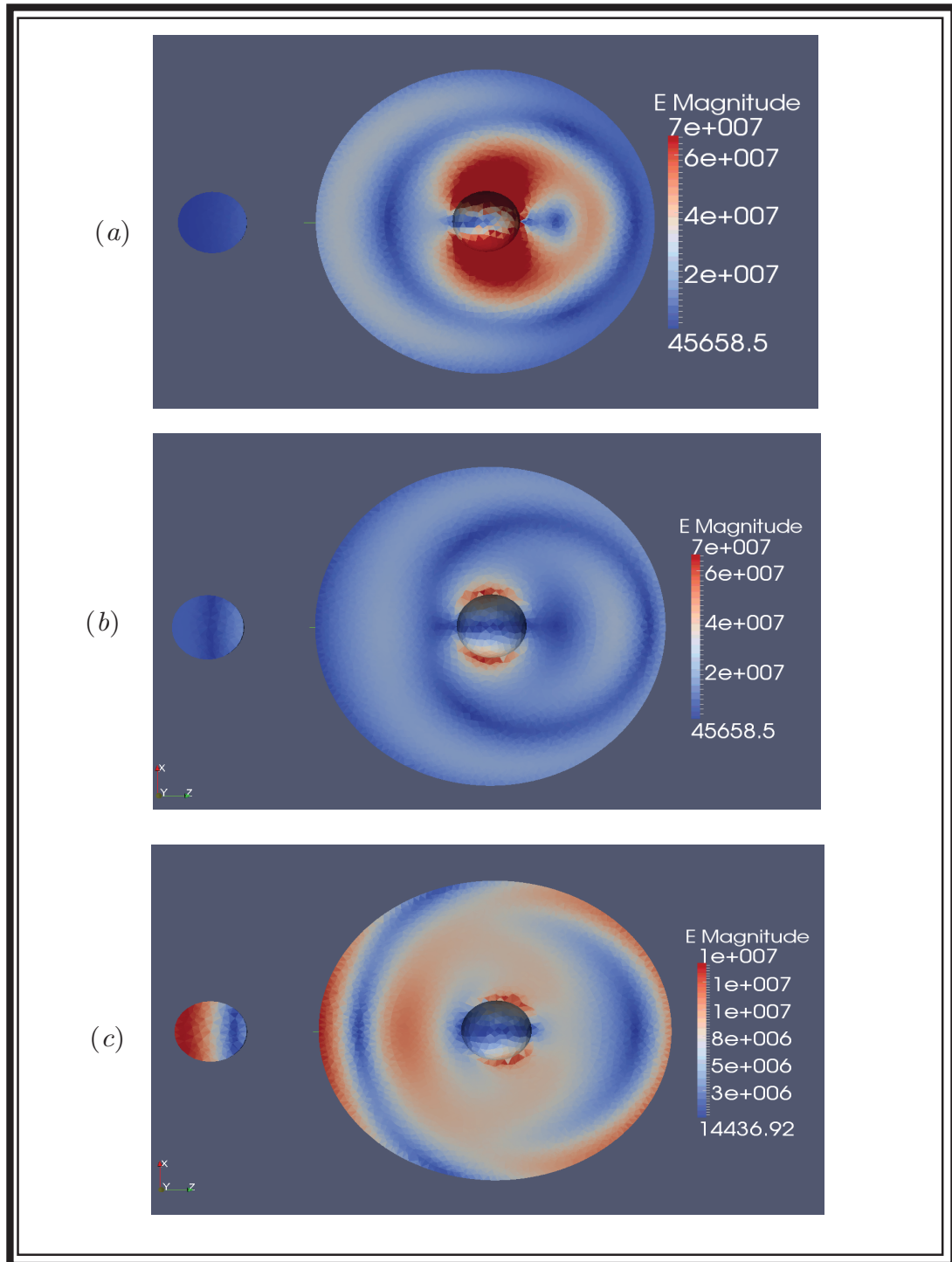


Fig. 6.8: (Colour) Propagation of scattered electric fields using global mesh truncation. The magnitude of the scattered field is shown at (a) 1000 time-steps (b) 1200 time-steps and (c) 1400 time-steps. Note that in (c) the scale has been intentionally reduced to emphasize the fields in the disconnected volume.

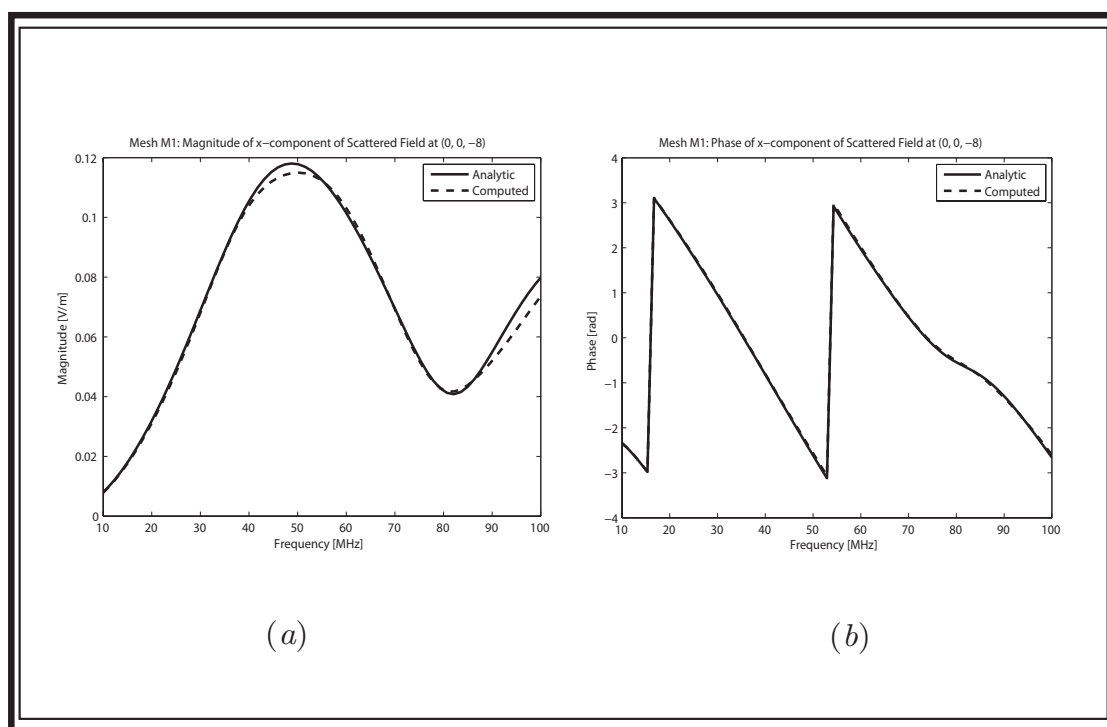


Fig. 6.9: Analytic comparison of the  $x$ -component of the scattered electric field at the backscatter location  $(0, 0, -8)$  in the disconnected geometry. (a) Magnitude and (b) Phase.

# An FVTD Thin-Wire Subcell Model

---

*All models are false but some models are useful.* – George Edward Pelham  
Box

The volumetric solution of Maxwell's Equations by the upwind, flux-split FVTD formulation presented to this point, does not require a structured partition of the computational domain  $V$ . One great benefit of using FVTD on unstructured meshes is that accurate modeling of problems involving both small and large geometric features (with respect to the wavelengths of interest) only requires a fine discretization in proximity to fine structural details. Unfortunately, there is only so far that this benefit can take us. If the problem of interest is dominated by fine structural detail, the number of cells required to model the geometry will be much larger than the number of cells required to spatially reproduce the simulation frequency bandwidth. Specifically, there

are many engineering applications that require modeling very thin wire-structures. By “thin” we imply both that the length of the wire is much longer than its radius, and that the radius of the wire is much smaller than the spatial sampling rate. In this case a subcell thin-wire model, where the wire is introduced into the volumetric mesh as a one-dimensional structure (along the length of the wire), can be used. As the full volumetric geometry of the wire is not required in such a model, the number of cells in the mesh can be drastically reduced. The difficulty then remains as to how the one-dimensional wire structure should affect, and be affected by, the volumetric fields – this is the “model” – and it ends up that for arbitrary, unstructured meshes, developing a flexible, accurate, model is difficult.

A thin-wire model for FVTD has been formulated in [9], and requires physical wires (or their discrete approximations) to be aligned with the edges of, and surrounded by, prismatic elements. First, this places constraints on the meshing procedure; the algorithm used to generate the mesh must support prismatic elements. Second, multi-wire junctions cannot be supported; it is not geometrically possible to interface three wires, surrounded by prismatic elements, without permitting the elements to overlap. A model that does not impose these constraints is the focus of this chapter.

Recently, a model that supports thin-wires that can be arbitrarily oriented within the computational domain has been developed for the Finite-Difference Time-Domain (FDTD) and Finite-Element Time-Domain (FETD) algorithms [18, 19, 41]. In this chapter the FVTD equivalent of these models is developed. The model equations result in a system of hyperbolic partial differential equations that are solved using a one-dimensional, upwind and flux-split FVTD formulation to ensure simple, yet effective, synchronization between the thin-wire model and the volumetric model.

### 7.1 *A Brief History of Time-Domain Thin-Wire Modeling*

To provide some context for the choice of thin-wire model developed in this work, a brief summary of available time-domain thin-wire modeling methods is warranted. A concise summary of differential thin-wire models used in the FDTD method is provided in [42]. Therein, the authors summarize available thin-wire models as falling into two distinct categories. First, there are methods in which the wire is assumed to carry charge and current, and a system of differential equations is constructed to relate these quantities to the surrounding fields. This method was introduced by Holland and Simpson [43], and is the approach adopted by [18, 19, 41] and this work. The benefit of introducing the additional variables of wire-charge and wire-current is that they permit flexibility when geometrically introducing the wire into the mesh. The second method is to assume that the presence of wire-current and charge introduce constraints on appropriately constructed contour integrals in the vicinity of the wire. In this method, the wire-current and charge are not introduced as additional unknown physical quantities to be determined by the solver. Instead, the fields near the wire simply act as if a wire is present. In FDTD this method can work quite well, as the contour integrals coincide with the regular geometry of the FDTD grid [44, 45]. For unstructured meshes, it is unclear how such a scheme can be constructed. (It should also be noted that in [42] a third method based on a weighted-residual approach is provided as a generalization of the first method.)

Yet another way of introducing thin-wires in to the numerical algorithm is to impose an integral equation (IE) relating the charge and current on the wire. The hybrid IE/FDTD/FETD schemes presented in [46] and [47] offer ways to incorporate IEs for thin-wire structures in the presence of dielectric objects but are beyond the scope of this work.

## 7.2 FVTD Thin-Wire Formulation

In this section the theory for a sub-cell thin-wire model compatible with the FVTD solution to Maxwell's Equations is developed. The method is based on the Holland-Simpson model [43] and its adaptation to arbitrarily oriented thin-wire structures as developed by Edelvik et. al. [18, 19]. At the core of the thin-wire formulation is the assumption that, in proximity to radiating wire structures, the fields are dominated by static contributions.

### 7.2.1 Static Fields of an Infinitely Long Thin Wire

According to the Holland-Simpson formulation presented in [43], we assume an infinitely long thin-wire of radius  $a$ , directed in the  $\hat{\zeta}$  direction, embedded in a homogeneous and simple medium  $\epsilon$ ,  $\mu$ , and  $\sigma$  as shown in Figure 7.1 (a). Introducing a right-handed system of cylindrical coordinates  $(\rho, \phi, \zeta)$  and the additional unitary vectors  $\hat{\rho}$  and  $\hat{\phi}$ , it is assumed that the wire carries a current  $\mathcal{I}(t, \zeta)$  [A] and a per-unit-length (PUL) charge-density  $Q(t, \zeta)$  [C/m<sup>2</sup>]. Assuming that the volumetric discretization is much finer than the smallest wavelength of interest, the wire-driven fields in finite-volumes near the wire are well-approximated by a static  $\hat{\rho}$ -directed electric field and a static  $\hat{\phi}$ -directed magnetic field:

$$\vec{\mathcal{E}}(t, \rho, \phi, \zeta) = \mathcal{E}_\rho(t, \rho, \phi, \zeta)\hat{\rho} = \frac{Q(t, \zeta)}{2\pi\epsilon\rho}\hat{\rho} \quad (7.1)$$

and

$$\vec{\mathcal{H}}(t, \rho, \phi, \zeta) = \mathcal{H}_\phi(t, \rho, \phi, \zeta)\hat{\phi} = \frac{\mathcal{I}(t, \zeta)}{2\pi\rho}\hat{\phi} \quad (7.2)$$



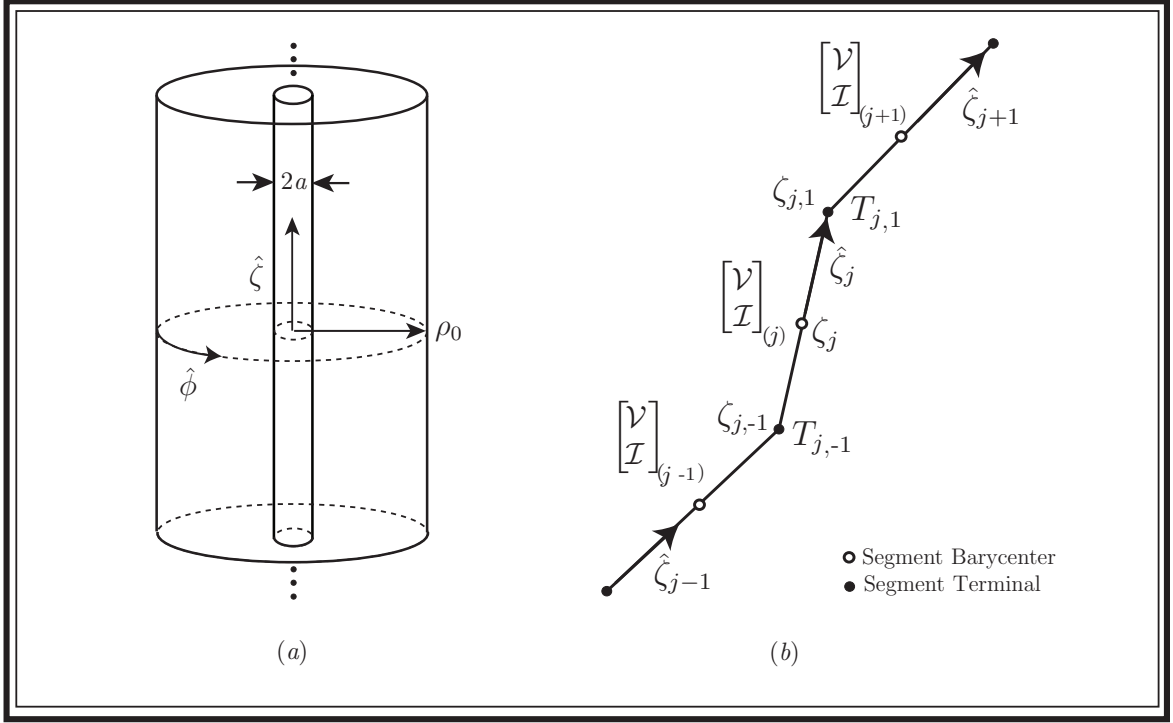


Fig. 7.1: (a) A portion of an infinite  $\hat{\zeta}$ -directed thin-wire of radius  $a$  used to derive the thin-wire subcell model. Field/wire coupling occurs to the distance  $\rho_0$ . The medium inside the cylinder of radius  $\rho_0$  is assumed constant. (b) Discretization of the continuous wire domain into first-order wire segments. The infinite model is assumed to approximately hold on each segment.

The assumption that the volumetric discretization is finer than the smallest wavelength adheres to practical volumetric sampling.

### 7.2.2 The Thin-Wire Equations

We now derive a system of equations describing the physical behaviour of the wire current and charge as a function of time and position along the wire. Assuming that the thin-wire has a conductivity  $\sigma_w[S/m]$ , the  $\hat{\phi}$  component of Faraday's law is:

$$\frac{\partial \mathcal{E}_\zeta(t, \vec{r})}{\partial \rho} = \mu \frac{\partial \mathcal{H}_\phi(t, \vec{r})}{\partial t} + \frac{\partial \mathcal{E}_\rho(t, \vec{r})}{\partial \zeta} + \frac{\mathcal{J}_\zeta(t, \vec{r})}{\sigma_w} \quad (7.3)$$

where  $\mathcal{J}_\zeta(t, \vec{r})$  is a wire-confined current density due to the wire current. Substituting the near-wire assumptions (7.1) and (7.2) into (7.3) gives:

$$\frac{\partial \mathcal{E}_\zeta(t, \vec{r})}{\partial \rho} = \frac{\mu}{2\pi\rho} \left( \frac{\partial \mathcal{I}(t, \zeta)}{\partial t} + c^2 \frac{\partial \mathcal{Q}(t, \zeta)}{\partial \zeta} \right) + \frac{\mathcal{J}_\zeta(t, \vec{r})}{\sigma_w} \quad (7.4)$$

where, as usual,  $c = \sqrt{\epsilon\mu}^{-1}$ . The  $\rho$  derivative is removed by integrating this last equation over disk of radius  $\rho$  transverse to the  $\zeta$  direction resulting in

$$\mathcal{E}_\zeta(t, \rho, \phi, \zeta) = \frac{\mu}{2\pi} \ln\left(\frac{\rho}{a}\right) \left( \frac{\partial \mathcal{I}(t, \zeta)}{\partial t} + c^2 \frac{\partial \mathcal{Q}(t, \zeta)}{\partial \zeta} \right) + \mathcal{I}(t, \zeta) R \quad (7.5)$$

where  $R = 1/(\pi a^2 \sigma_w)[\Omega/\text{m}]$  is a PUL resistance and where we have used the fact that the tangential field component  $\mathcal{E}_\zeta(t, \vec{r})$  decays very rapidly inside good conductors.

Equation (7.5) relates the current and PUL charge at location  $\zeta$  along the thin-wire to the longitudinal component  $\mathcal{E}_\zeta(t, \rho, \phi, \zeta)$  at some distance away from the wire. By obtaining this longitudinal field from the three-dimensional FVTD solution to Maxwell's Equations, we can interpret (7.5) as a single equation relating the current and PUL charge on the thin-wire. The obvious question then becomes: which values of  $\rho$  and  $\phi$  should we use? An answer to this question that allows the wire to be arbitrarily oriented within the volumetric mesh is to choose a weighted average over all  $\phi$  and over  $\rho$  to some distance  $\rho_0$ . This was the pioneering step apparently introduced by Ledfelt [48, 41] and used in [18, 19]. It is important to note that by truncating  $\rho$  to  $\rho_0$  we ensure that the driving longitudinal field can be computed at  $\zeta$  in constant time. Selecting an appropriate value of  $\rho_0$  is discussed in Section 7.5. We introduce a weighting function  $g(\rho)$  [ $1/\text{m}^2$ ] which serves to couple the fields to the wire in a weighted average

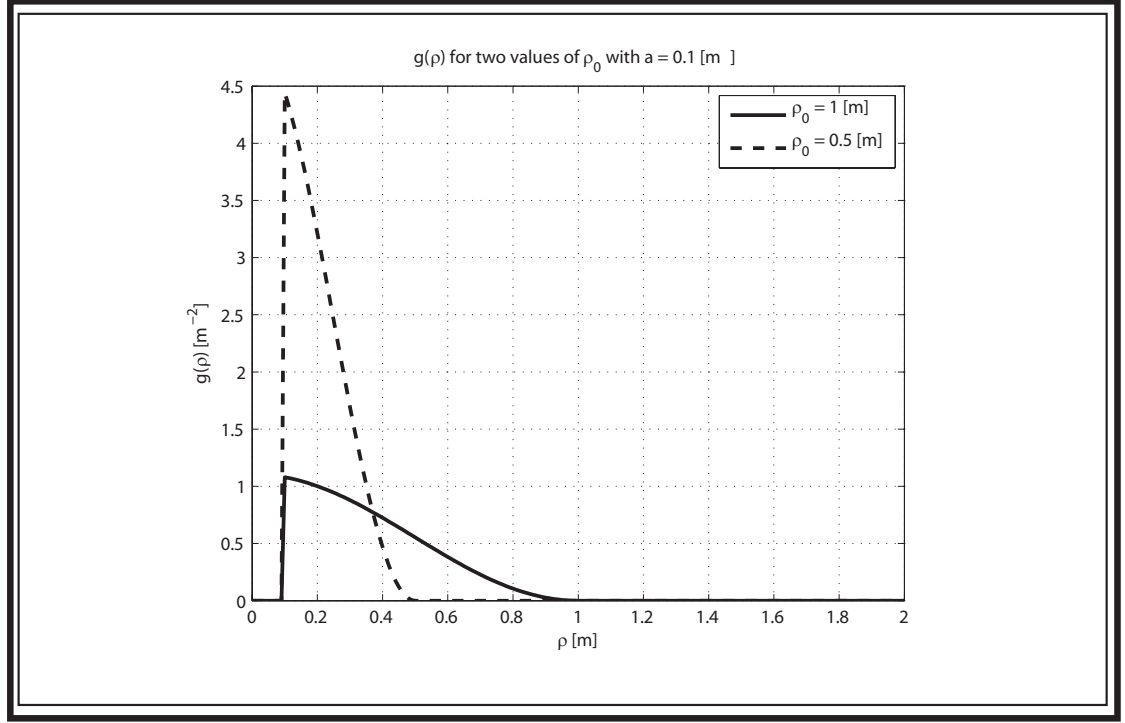


Fig. 7.2: The wire-field coupling function  $g(\rho)$  shown for two values of  $\rho_0$ . One limitation of this function is that as  $\rho_0$  decreases, it becomes more and more difficult to numerically integrate.

sense. The function  $g(\rho)$  is assumed to have the following properties:

$$\begin{aligned}
 g(\rho) &= 0 & \forall \rho < a \\
 g(\rho) &= 0 & \forall \rho > \rho_0 \\
 \int_0^{\rho_0} 2\pi g(\rho) d\rho &= 1
 \end{aligned} \tag{7.6}$$

In this work we adopt the same weighting function used in [18, 19] which is given explicitly as:

$$g(\rho) = \frac{1 + \cos \frac{\pi\rho}{\rho_0}}{\pi(\rho_0^2 - a^2) - \frac{2\rho_0^2}{\pi} \left( 1 + \cos \frac{\pi a}{\rho_0} + \frac{\pi a}{\rho_0} \sin \frac{\pi a}{\rho_0} \right)}, \quad a \leq \rho \leq \rho_0 \tag{7.7}$$

The function  $g(\rho)$  is shown depicted for two values of  $\rho_0$  and an exaggerated wire-radius of 0.1 [m] in Figure 7.2. Weighting both sides of (7.5) by  $g(\rho)$  and integrating over a disk of radius  $\rho_0$  transverse to the wire axis gives:

$$\langle \mathcal{E}_\zeta(t, \vec{r}) \rangle = L \left( \frac{\partial \mathcal{I}(t, \zeta)}{\partial t} + c^2 \frac{\partial \mathcal{Q}(t, \zeta)}{\partial \zeta} \right) + \mathcal{I}(t, \zeta) R \quad (7.8)$$

where the weighted average of the longitudinal electric field is

$$\langle \mathcal{E}_\zeta(t, \vec{r}) \rangle \triangleq \int_0^{2\pi} \int_0^{\rho_0} \mathcal{E}_\zeta(t, \vec{r}) g(\rho) \rho d\rho d\phi \quad (7.9)$$

and where the PUL inductance  $L$  [H/m] is defined as:

$$L \triangleq \frac{\mu}{2\pi} \int_0^{\rho_0} \ln\left(\frac{\rho}{a}\right) 2\pi g(\rho) \rho d\rho. \quad (7.10)$$

A second equation relating the current and charge along the wire can be obtained by conservation of charge:

$$\frac{\partial \mathcal{I}(t, \zeta)}{\partial \zeta} + \frac{\partial \mathcal{Q}(t, \zeta)}{\partial t} = -\frac{\sigma}{\epsilon} \mathcal{Q}(t, \zeta) \quad (7.11)$$

The two equations (7.8) and (7.11) form a coupled pair of partial differential equations relating the PUL charge and current on the thin-wire structure to the surrounding fields that can be written as a system of equations:

$$\boxed{\frac{\partial}{\partial t} \begin{bmatrix} \mathcal{Q} \\ L\mathcal{I} \end{bmatrix} + \frac{\partial}{\partial \zeta} \begin{bmatrix} \mathcal{I} \\ Lc^2\mathcal{Q} \end{bmatrix} + \begin{bmatrix} \sigma\epsilon^{-1}\mathcal{Q} \\ R\mathcal{I} \end{bmatrix} = \begin{bmatrix} 0 \\ \langle \mathcal{E}_\zeta \rangle \end{bmatrix}} \quad (7.12)$$

where the explicit dependence on time and position have been dropped for notational

brevity. It will be shown in Section 7.2.6 that (7.12) is a hyperbolic system of two partial differential equations that permits a finite-volume solution. Before introducing the finite-volume scheme for solving this system, we will introduce a thin-wire voltage.

### 7.2.3 The Thin-Wire Voltage

For the extraction of engineering parameters, such as input impedance, it is beneficial to define the thin-wire equations (7.12) in terms of a voltage. This is accomplished by defining a PUL capacitance  $C$  [F/m] satisfying  $1/(LC) = c^2$  [18]. The PUL capacitance permits the definition of a wire voltage  $\mathcal{V}(t, \zeta) = Q(t, \zeta)/C$  [V] and the system of equations (7.12) can be re-written as:

$$\boxed{\frac{\partial}{\partial t} \begin{bmatrix} C\mathcal{V} \\ LI \end{bmatrix} + \frac{\partial}{\partial \zeta} \begin{bmatrix} \mathcal{I} \\ \mathcal{V} \end{bmatrix} + \begin{bmatrix} G\mathcal{V} \\ RI \end{bmatrix} = \begin{bmatrix} 0 \\ \langle \mathcal{E}_\zeta \rangle \end{bmatrix}} \quad (7.13)$$

where  $G = \sigma C/\epsilon$  [S/m] is a PUL conductance. Insight is obtained by investigating the nature of the introduced voltage, which is defined in terms of the arbitrary definitions of  $L$  and  $C$ . Integrating the static electric field transverse to the wire, from  $a$  to  $\rho$  for fixed  $\zeta$ , defines a voltage  $\mathcal{V}^*(t, \rho, \zeta)$  referenced from  $a$  to  $\rho$ :

$$\mathcal{V}^*(t, \rho, \zeta) \triangleq \int_a^\rho \mathcal{E}_\rho(t, \rho', \zeta) d\rho' = \frac{Q(t, \zeta)}{2\pi\epsilon} \ln\left(\frac{\rho}{a}\right). \quad (7.14)$$

Averaging  $\mathcal{V}^*$  analogously to the computation of  $\langle \mathcal{E}_\zeta \rangle$  such that:

$$\langle \mathcal{V}^*(t, \rho, \zeta) \rangle = Q(t, \zeta) \frac{1}{2\pi\epsilon} \int_0^{\rho_0} \ln\left(\frac{\rho}{a}\right) 2\pi g(\rho) \rho d\rho \quad (7.15)$$

and defining a PUL capacitance  $C^*$  [F/m] as:

$$C^{*-1} \triangleq \frac{1}{2\pi\epsilon} \int_0^{\rho_0} \ln\left(\frac{\rho}{a}\right) 2\pi g(\rho) \rho d\rho \quad (7.16)$$

shows, from the definition of  $L$  given in (7.10), that  $1/(LC^*) = \mu\epsilon$  *i.e.*  $C^* = C$ . It is a direct consequence of this last equality that:

$$\mathcal{V}(t, \zeta) = \langle \mathcal{V}^*(t, \rho, \zeta) \rangle = \mathcal{Q}(t, \zeta) C^{-1} \quad (7.17)$$

From (7.17) we conclude that the voltage  $\mathcal{V}(t, \zeta)$  along the thin-wire is defined in terms of the coupling distance  $\rho_0$  and should be interpreted carefully.

#### 7.2.4 Discretization of the Wire Domain

The FVTD solution to the one-dimensional hyperbolic system (7.13) is formulated by partitioning the wire domain  $W$  into  $N_W$  first-order line-segments  $W_j$ ,  $j = 1, \dots, N_W$  as shown in Figure 7.1 (b). To simplify the notation we assume the coordinate  $\zeta$  continuously defines a unique position in  $W$ . For distinct physical wires, different coordinate systems are used on each wire. The barycentre of the  $j^{\text{th}}$  wire-segment is denoted locally by  $\zeta_j$  and corresponds to a global coordinate  $\vec{r}(\zeta_j)$  in three-dimensional space. The segment is bounded by two facets (points) denote by  $T_{j,k}$ ,  $k = \pm 1$ , located along the wire at  $\zeta_{j,k} = \zeta_j + k \cdot |W_j|/2$ , where  $|W_j|$  is the length of the segment. Corresponding to these local coordinates are two global coordinates  $\vec{r}(\zeta_{j,k})$  that define the segment orientation  $\hat{\zeta}_j = (\vec{r}(\zeta_{j,1}) - \vec{r}(\zeta_{j,-1}))/|W_j|$ . Accordingly, in one dimension, the “outward normal” in coordinate  $\zeta$  to the segment at the end  $T_{j,k}$  is  $k$ . The physical parameters of the medium are assumed to be constants  $\epsilon_j$ ,  $\mu_j$  and  $\sigma_j$ , to a transverse distance  $\rho_0$ .

## 7.2.5 The FVTD Formulation of the Thin-Wire Equations

For the time-being we will assume that we are interested in solving the system of equations (7.13) only in the one-dimensional wire domain  $W$ ; that is, we will not be concerned with the source terms coming from the volumetric field model. Coupling between the developed one-dimensional theory and the volumetric solver is addressed in Section 7.5.

The current and voltage on each wire segment are assumed to be related by a segment-dependent instance of the system (7.13). On each segment we can write (7.13) as:

$$\partial_t \mathbf{u}(t, \zeta) + \bar{a}_j \cdot \partial_\zeta \mathbf{u}(t, \zeta) = -\bar{b}_j \cdot \mathbf{u}(t, \zeta) + \bar{c}_j \cdot \mathbf{s}(t, \zeta) \quad \zeta \in W_j \quad (7.18)$$

where  $\mathbf{u}(t, \zeta) = [\mathcal{V}(t, \zeta) \mathcal{I}(t, \zeta)]^T$  and  $\mathbf{s}(t, \zeta) = [0 \langle \mathcal{E}_\zeta(t, \zeta) \rangle]^T$  in analogy to the three-dimensional solution and source vectors for the volumetric field solver. The constant  $2 \times 2$  matrices  $\bar{a}_j$ ,  $\bar{b}_j$  and  $\bar{c}_j$  are:

$$\bar{a}_j \triangleq \begin{bmatrix} 0 & C_j^{-1} \\ L_j^{-1} & 0 \end{bmatrix} \quad \bar{b}_j \triangleq \begin{bmatrix} G_j/C_j & 0 \\ 0 & R_j/L_j \end{bmatrix} \quad \bar{c}_j \triangleq \begin{bmatrix} C_j^{-1} & 0 \\ 0 & L_j^{-1} \end{bmatrix} \quad (7.19)$$

FVTD equations for segment  $j$  are obtained by averaging (7.18) over the segment and applying the one-dimensional divergence theorem converting (7.18) to:

$$\boxed{d_t \mathbf{u}_{(j)}(t) = -\frac{1}{|W_j|} \sum_{k=-1}^1 \mathcal{F}_{j,k}^o(t, \zeta_{j,k}) - \bar{b}_j \cdot \mathbf{u}_{(j)}(t) + \bar{c}_j \cdot \mathbf{s}_{(j)}(t)} \quad (7.20)$$

As in the volumetric case, the quantities  $\mathbf{u}_{(j)}(t)$  and  $\mathbf{s}_{(j)}(t)$  denote averages of the solution and source vectors over  $W_j$  and where the flux  $\mathcal{F}_{j,k}^o = k\bar{a}_j \cdot \mathbf{u}^o(t, \zeta_{j,k})$  is

dependent on the segment boundary-limit of the solution  $\mathbf{u}^o(t, \zeta_{j,k})$ . The summation over  $k$  is a convenient way to represent the summation over the two “facets” (ends) of the wire, with  $k$  accounting for the outward normal and where  $k = 0$  does not contribute to the equation. This equation is completely analogous to (3.7) and all of the general FVTD methodology previous discussed applies. Specifically, we will again adopt an upwind, flux-split REA algorithm where the cell averages of the solution are associated with each cell.

### 7.2.6 Hyperbolicity and Flux-Splitting of the Thin-Wire Equations

To simultaneously prove the hyperbolicity of the system of PDEs (7.18) and to provide a flux-split mechanism for upwinding the FVTD thin-wire solution, we diagonalize the flux matrix,  $\bar{a}(\epsilon, \mu, k) = k\bar{a}(\epsilon, \mu)$ , at facet  $k$  (or in the direction  $k$ ) by a similarity transformation:

$$\bar{a}(\epsilon, \mu, k) = \bar{\psi}(\epsilon, \mu) \cdot \bar{\lambda}(\epsilon, \mu, k) \cdot \bar{\psi}(\epsilon, \mu)^{-1} \quad (7.21)$$

where the matrix of eigenvalues is:

$$\bar{\lambda}(\epsilon, \mu, k) = \text{diag}(-kc, kc) \quad (7.22)$$

and where the eigenmatrices are:

$$\bar{\psi}(\epsilon, \mu) = \begin{bmatrix} 1 & 1 \\ -Y & Y \end{bmatrix} \quad \bar{\psi}(\epsilon, \mu)^{-1} = \frac{1}{2} \begin{bmatrix} 1 & -Z \\ 1 & Z \end{bmatrix}. \quad (7.23)$$

As in the volumetric case, the impedance  $Z = \sqrt{\mu/\epsilon} = \sqrt{L/C}$  and the admittance  $Y = Z^{-1}$ . For each direction  $k$  along the wire the flux matrix  $\bar{a}(\epsilon, \mu, k)$  is split into positive and negative parts  $\bar{a}^{\pm}(\epsilon, \mu, k) = \bar{\psi}(\epsilon, \mu) \cdot \bar{\lambda}^{\pm}(\epsilon, \mu, k) \cdot \bar{\psi}(\epsilon, \mu)^{-1}$  where  $\bar{\lambda}^{\pm}(\epsilon, \mu, k)$



is defined analogously to (3.8). As the eigenvalues of the thin-wire flux-matrix are real, the thin-wire system of PDEs is hyperbolic. The flux-split matrices for direction  $k$  in  $W_j$  are:

$$\bar{\mathbf{a}}_j^+(k) = \frac{1}{2} \begin{bmatrix} c_j & kC_j^{-1} \\ kL_j^{-1} & c_j \end{bmatrix} \quad \bar{\mathbf{a}}_j^-(k) = \begin{bmatrix} -c_j & kC_j^{-1} \\ kL_j^{-1} & -c_j \end{bmatrix} \quad (7.24)$$

To simplify things we introduce the following notation inside  $W_j$ :

$$\begin{aligned} \bar{a}_{j,k} &\triangleq \bar{a}_j(k) \\ \bar{\lambda}_{j,k} &\triangleq \bar{\lambda}(\epsilon_j, \mu_j, k) \\ \bar{\psi}_j &\triangleq \bar{\psi}(\epsilon_j, \mu_j) \end{aligned} \quad (7.25)$$

The flux-split, upwind, thin-wire system can then be written as

$$\begin{aligned} \partial_t \mathbf{u}_{(j)}(t) &= -\bar{b}_j \cdot \mathbf{u}_{(j)}(t) + \bar{c}_j \cdot \mathbf{s}_{(j)}(t) - \sum_{k=-1}^1 (\bar{a}_{j,k}^+ \cdot \mathbf{u}^*(t, \zeta_{j,k}) + \bar{a}_{j,k}^- \cdot \mathbf{u}^{**}(t, \zeta_{j,k})) \\ &= -\bar{b}_j \cdot \mathbf{u}_j(t) + \bar{c}_j \cdot \mathbf{s}_j(t) - \sum_{k=-1}^1 (\mathcal{F}_{j,k}^*(t) + \mathcal{F}_{j,k}^{**}(t)) \end{aligned} \quad (7.26)$$

where  $\mathbf{u}^*(t, \zeta_{j,k})$  is the limit of the solution at facet  $k$  from inside  $W_j$  and  $\mathbf{u}^{**}(t, \zeta_{j,k})$  is the limit of the solution at facet  $k$  reconstructed from outside  $W_j$ . The flux reconstruction must account for the physical boundary conditions on the current and voltage at the wire-segment terminals. These conditions can be constructed by analogy to the three-dimensional boundary conditions provided in [9] as shown in Section 7.3. An alternative way to impose the boundary conditions is to use a subcell circuit model to join/terminate wire segments as developed in Chapter 8. In fact, these subcell models provide a direct mechanism for reconstructing the flux at multi-wire junctions.

## 7.2.7 Thin-wire Characteristics

We conclude this section by determining the thin-wire characteristics which will be used to develop thin-wire boundary conditions in Section 7.3 and Chapter 8. We assume the source-free and loss-free instance of equation (7.18) in the  $k$  direction:

$$\partial_t \mathbf{u}(t, \zeta) + \bar{\bar{a}}_{j,k} \cdot \partial_\zeta \mathbf{u}(t, \zeta) = 0$$

multiplying this equation by  $\bar{\bar{\psi}}_j^{-1}$  uncouples the system:

$$\partial_t \mathbf{w}_{j,k}(t, \zeta) + \bar{\bar{\lambda}}_{j,k} \cdot \partial_\zeta \mathbf{w}_{j,k}(t, \zeta) = 0, \quad \zeta \in W_j$$

where the subscript  $j, k$  notation is again used to associate the characteristic with the  $j^{\text{th}}$  segment in the  $k$  direction, and where the thin-wire characteristic vector  $\mathbf{w}_{j,k}(t, \zeta)$  is:

$$\mathbf{w}_{j,k}(t, \zeta) = \bar{\bar{\psi}}_j^{-1} \cdot \mathbf{u}(t, \zeta) = \frac{1}{2} \begin{bmatrix} \mathcal{V}(t, \zeta) - Z_j \mathcal{I}(t, \zeta) \\ \mathcal{V}(t, \zeta) + Z_j \mathcal{I}(t, \zeta) \end{bmatrix} \quad (7.27)$$

For  $k = 1$ , the first entry in the characteristic vector propagates unchanged along the characteristic curve  $\zeta + t/c_j = \text{constant}$  at velocity  $-c_j$  *i.e.* it propagates in the negative  $\zeta$  direction. The second entry propagates at the same speed in the positive  $\zeta$  direction along the curve  $\zeta - t/c_j = \text{constant}$ .

$$\mathbf{w}_{j,k}^\pm(t, \zeta) \triangleq \frac{1}{\pm c_j} \bar{\bar{\lambda}}_j^\pm(k) \cdot \bar{\bar{\psi}}_j^{-1} \cdot \mathbf{u}(t, \vec{r}) \quad (7.28)$$

The positive characteristic vector in the  $k = 1$  ( $\zeta$ ) direction is:

$$\mathbf{w}_{j,1}^+(t, \zeta) = \frac{1}{c_j} \begin{bmatrix} 0 & 0 \\ 0 & c_j \end{bmatrix} \cdot \frac{1}{2} \begin{bmatrix} \mathcal{V}(t, \zeta) - Z_j \mathcal{I}(t, \zeta) \\ \mathcal{V}(t, \zeta) + Z_j \mathcal{I}(t, \zeta) \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 0 \\ \mathcal{V}(t, \zeta) + Z_j \mathcal{I}(t, \zeta) \end{bmatrix} \quad (7.29)$$

while the negative characteristic in the  $k = 1$  ( $\zeta$ ) direction is:

$$\mathbf{w}_{j,1}^-(t, \zeta) = \frac{1}{-c_j} \begin{bmatrix} -c_j & 0 \\ 0 & 0 \end{bmatrix} \cdot \frac{1}{2} \begin{bmatrix} \mathcal{V}(t, \zeta) - Z_j \mathcal{I}(t, \zeta) \\ \mathcal{V}(t, \zeta) + Z_j \mathcal{I}(t, \zeta) \end{bmatrix} = \frac{1}{2} \begin{bmatrix} \mathcal{V}(t, \zeta) - Z_j \mathcal{I}(t, \zeta) \\ 0 \end{bmatrix} \quad (7.30)$$

### 7.3 Thin-Wire Boundary Conditions

In the thin-wire model we have assumed that surrounding each thin-wire segment, the physical parameters of the three dimensional medium are constant to a radial distance of  $\rho_0$  and independent of angle  $\phi$ . This assumption leads to a somewhat natural definition of the characteristic impedance of each line segment  $Z_j$ . So long as this assumption is satisfied, it still permits discontinuities in the characteristic impedance as we go from wire segment to wire segment, such that  $Z_j$  may not equal  $Z_{j+1}$ . This is also the case if the wires have different radii, as suggested by the definition of  $L_j$  in (7.10). As in the volumetric case we are required to handle the boundary conditions explicitly.

#### 7.3.1 General Thin-Wire boundary conditions

As discussed in the volumetric formulation, unwinding the partial differential form of the thin-wire equations is equivalent to computing the flux at the boundary of the

wire segments using only that part of the solution that flows into the boundary. We assume that wire segment  $j$  shares facet  $\partial W_{j,k} = \partial W_{m,k'}$  with wire segment  $m$ . We suppose that a solution  $\mathbf{u}_{j,k}^+$  is reproduced by spatial reconstruction alone at time  $t$  at  $\partial W_{j,k}$  from inside  $W_j$ . Similarly,  $\mathbf{u}_{j,k}^-$  is reproduced from the cell-average solution inside  $W_m$ . The true boundary limit values  $\mathbf{u}_{j,k}^*$  and  $\mathbf{u}_{j,k}^{**}$  can be written in terms of the characteristics as:

$$\begin{aligned} \frac{1}{c_j} \bar{\lambda}_j^+(k) \cdot \bar{\psi}^{-1} \cdot \mathbf{u}_{j,k}^+ &= \frac{1}{c_j} \bar{\lambda}_j^+(k) \cdot \bar{\psi}^{-1} \cdot \mathbf{u}_{j,k}^* \\ \frac{1}{-c_m} \bar{\lambda}_m^-(k') \cdot \bar{\psi}^{-1} \cdot \mathbf{u}_{j,k}^+ &= \frac{1}{-c_m} \bar{\lambda}_m^-(k') \cdot \bar{\psi}^{-1} \cdot \mathbf{u}_{j,k}^{**} \end{aligned} \quad (7.31)$$

Multiplying the first equation by  $c_j \bar{\psi}_{j,k}$  and the second by  $-c_m \bar{\psi}_{m,k'}$  gives

$$\begin{aligned} \bar{a}_{j,k}^+ \cdot \mathbf{u}_{j,k}^+ &= \bar{a}_{j,k}^+ \cdot \mathbf{u}_{j,k}^* \\ \bar{a}_{m,k'}^+ \cdot \mathbf{u}_{m,k'}^+ &= \bar{a}_{m,k'}^+ \cdot \mathbf{u}_{j,k}^{**} \end{aligned} \quad (7.32)$$

where  $\bar{a}_{m,k'}^+ = -\bar{a}_{m,-k'}^-$ . Recalling the symmetry of Figure 3.1, we can use  $\mathbf{u}_{j,k}^{**} = \mathbf{u}_{m,k'}^*$  to expand (7.32) in order to obtain the following constraints on the line current and voltage:

$$\begin{aligned} \mathcal{V}_{j,k}^* + kZ_j \mathcal{I}_{j,k}^* &= \mathcal{V}_{j,k}^+ + kZ_j \mathcal{I}_{j,k}^+ \\ \mathcal{V}_{m,k'}^* - kZ_m \mathcal{I}_{m,k'}^* &= \mathcal{V}_{m,k'}^+ - kZ_m \mathcal{I}_{m,k'}^+ \end{aligned} \quad (7.33)$$

$$\begin{aligned} \mathcal{I}_{j,k}^* + kY_j \mathcal{V}_{j,k}^* &= \mathcal{I}_{j,k}^+ + kY_j \mathcal{V}_{j,k}^+ \\ \mathcal{I}_{m,k'}^* - kY_m \mathcal{V}_{m,k'}^* &= \mathcal{I}_{m,k'}^+ - kY_m \mathcal{V}_{m,k'}^+ \end{aligned} \quad (7.34)$$

where the fact that  $k' = -k$  has been used. From the two systems of equations (7.33) and (7.34) conditions relating the true boundary values and reconstructed boundary values can be enforced for a variety of boundary conditions.

## 7.3.2 Inhomogeneous Dielectric Boundary Conditions

In the case of a discontinuity in the characteristic impedance from one wire-segment to another (due to a jump in the surrounding medium or wire-radius) , the appropriate boundary condition is continuity of the line voltage and current:

$$\begin{aligned}\mathcal{V}_{j,k}^* &= \mathcal{V}_{m,k'}^* = \mathcal{V}_{j,k}^o \\ \mathcal{I}_{j,k}^* &= \mathcal{I}_{m,k'}^* = \mathcal{I}_{j,k}^o\end{aligned}\tag{7.35}$$

These boundary conditions follow directly from the definition of the voltage and current and the continuity requirements of the electric and magnetic fields surrounding the thin-wire. Substituting these equations into (7.34) and solving for the boundary values  $\mathcal{V}_{j,k}^o$  and  $\mathcal{I}_{j,k}^o$  we obtain the following system:

$$\begin{bmatrix} \mathcal{I}_{j,k}^o \\ \mathcal{V}_{j,k}^o \end{bmatrix} = \begin{bmatrix} \frac{k}{Z_j+Z_m} & \frac{Z_j}{Z_j+Z_m} \\ \frac{Y_j}{Y_j+Y_m} & \frac{k}{Y_j+Y_m} \end{bmatrix} \cdot \begin{bmatrix} \mathcal{V}_{j,k}^+ \\ \mathcal{I}_{j,k}^+ \end{bmatrix} + \begin{bmatrix} \frac{-k}{Z_j+Z_m} & \frac{Z_m}{Z_j+Z_m} \\ \frac{Y_m}{Y_j+Y_m} & \frac{-k}{Y_j+Y_m} \end{bmatrix} \cdot \begin{bmatrix} \mathcal{V}_{m,k'}^+ \\ \mathcal{I}_{m,k'}^+ \end{bmatrix}\tag{7.36}$$

The required boundary flux at the wire terminal  $T_{j,k}$  in equation (7.20) is:

$$\mathcal{F}_{j,k}^o = \bar{\mathbf{a}}_{j,k} \cdot \mathbf{U}_{i,k}^o = \begin{bmatrix} kC_j^{-1}\mathcal{I}_{j,k}^o \\ kL_j^{-1}\mathcal{V}_{j,k}^o \end{bmatrix}\tag{7.37}$$

which, when substituted into (7.36) gives:

$$\begin{aligned}\mathcal{F}_{j,k}^o &= \begin{bmatrix} kC_j^{-1}\mathcal{I}_{j,k}^o \\ kL_j^{-1}\mathcal{V}_{j,k}^o \end{bmatrix} = \begin{bmatrix} C_j^{-1} & 0 \\ 0 & L_j^{-1} \end{bmatrix} \cdot \begin{bmatrix} \frac{1}{Z_j+Z_m} & \frac{kZ_j}{Z_j+Z_m} \\ \frac{kY_j}{Y_j+Y_m} & \frac{1}{Y_j+Y_m} \end{bmatrix} \cdot \begin{bmatrix} \mathcal{V}_{j,k}^+ \\ \mathcal{I}_{j,k}^+ \end{bmatrix} \\ &+ \begin{bmatrix} C_j^{-1} & 0 \\ 0 & L_j^{-1} \end{bmatrix} \cdot \begin{bmatrix} \frac{-1}{Z_j+Z_m} & \frac{kZ_m}{Z_j+Z_m} \\ \frac{kY_m}{Y_j+Y_m} & \frac{-1}{Y_j+Y_m} \end{bmatrix} \cdot \begin{bmatrix} \mathcal{V}_{m,k'}^+ \\ \mathcal{I}_{m,k'}^+ \end{bmatrix}\end{aligned}\tag{7.38}$$

which can be written in the compact form

$$\mathcal{F}_{j,k}^o = \bar{t}_{j,k} \cdot \bar{a}_{j,k}^+ \cdot \mathbf{U}_{j,k}^+ - \bar{c}_j^{-1} \cdot \bar{c}_m \cdot \bar{t}_{m,j} \cdot \bar{a}_{m,k'}^+ \cdot \mathbf{U}_{m,k'}^+ \quad (7.39)$$

where  $\bar{c}$  is defined in equation (7.19) and where the transmission matrix  $\bar{t}_{j,m}$  is given by

$$\bar{t}_{j,m} = \begin{bmatrix} \frac{2Z_j}{Z_j + Z_m} & 0 \\ 0 & \frac{2Y_j}{Y_j + Y_m} \end{bmatrix} \quad (7.40)$$

Finally, for inhomogeneous dielectrics, from (7.39) we have

$$\boxed{\mathcal{F}_{j,k|ID}^o = \bar{t}_{j,k} \cdot \bar{a}_{j,k}^+ \cdot \mathbf{U}_{j,k}^+ - \bar{c}_j^{-1} \cdot \bar{c}_m \cdot \bar{t}_{m,j} \cdot \bar{a}_{m,k'}^+ \cdot \mathbf{U}_{m,k'}^+} \quad (7.41)$$

It is interesting to note that this equation is in a form analogous to the inhomogeneous boundary condition (3.26) developed for the volumetric FVTD algorithm.

### 7.3.3 Homogeneous Dielectric Boundary Conditions

When there is no change in the impedance across a thin-wire segment boundary, the transmission matrices become unity,  $\bar{c}_j = \bar{c}_m$  and (7.41) reduces to

$$\boxed{\mathcal{F}_{j,k|HD}^o = \bar{a}_{j,k}^+ \cdot \mathbf{U}_{j,k}^+ + \bar{a}_{j,k}^- \cdot \mathbf{U}_{m,k'}^+} \quad (7.42)$$

Again we note the similarity to the homogeneous field case (3.27).

## 7.3.4 PEC Boundary Condition (Short-Circuit)

When the wire terminates at a short circuit (say, at a PEC surface in the volumetric mesh) the appropriate boundary condition can be formulated from the dielectric condition (7.41) by assuming no incoming flux from the boundary (wire-segment  $m$ ) and by letting  $Z_m$  go to zero in the transmission matrix definitions. The resulting expression is:

$$\boxed{\mathcal{F}_{j,k|PEC}^o = \bar{t}_{j,k}^{PEC} \cdot \bar{a}_{j,k}^+ \cdot \mathbf{u}_{j,k}^+, \quad \text{with } \bar{t}_{j,k}^{PEC} = \lim_{Z_m \rightarrow 0} \bar{t}_{j,m} = \begin{bmatrix} 2 & 0 \\ 0 & 0 \end{bmatrix}} \quad (7.43)$$

Expanding and simplifying this expression yields

$$\begin{bmatrix} \mathcal{I}_{j,k}^o \\ \mathcal{V}_{j,k}^o \end{bmatrix} = \begin{bmatrix} Y_j \mathcal{V}_{j,k}^+ + k \mathcal{I}_{j,k}^+ \\ 0 \end{bmatrix} \quad (7.44)$$

demonstrating that the voltage at the boundary is zero as we expect across a short circuit.

## 7.3.5 PMC Boundary Condition (Open-Circuit)

A PMC boundary condition can be formulated from the dielectric boundary condition (7.41) by assuming no incoming flux from the boundary and that the admittance  $Y_m$  is zero. The resulting expression is

$$\boxed{\mathcal{F}_{j,k|PMC}^o = \bar{t}_{j,k}^{PMC} \cdot \bar{a}_{j,k}^+ \cdot \mathbf{u}_{j,k}^+, \quad \text{with } \bar{t}_{j,k}^{PMC} = \lim_{Y_m \rightarrow 0} \bar{t}_{j,m} = \begin{bmatrix} 0 & 0 \\ 0 & 2 \end{bmatrix}} \quad (7.45)$$

which when expanded:

$$\begin{bmatrix} \mathcal{I}_{j,k}^o \\ \mathcal{V}_{j,k}^o \end{bmatrix} = \begin{bmatrix} 0 \\ k\mathcal{V}_{j,k}^+ + Z_j\mathcal{I}_{j,k}^+ \end{bmatrix} \quad (7.46)$$

shows that the current at the boundary is zero as we expect through an open circuit.

### 7.3.6 Absorbing Boundary Conditions

For a physical wire embedded in the volumetric domain  $V$ , an absorbing boundary condition doesn't make much sense. If however, we are interested in completing the theory for the solution to the thin-wire equations on  $W$  in one-dimensional space only, an absorbing boundary condition is useful. As the direction of propagation is in the  $\pm\zeta$  direction, the one-dimensional analog of the Silver-Müller boundary conditions presented in Section 5.1 provides perfect absorption to numerical precision. Therefore, if an absorbing boundary condition is desired we can impose

$$\boxed{\mathcal{F}_{j,k|SM}^o = \bar{a}_{j,k}^+ \cdot \mathbf{u}_{j,k}^+} \quad (7.47)$$

### 7.3.7 Imposed Boundary Conditions

As a final, very useful boundary condition we consider imposing the incoming flux at a terminal  $T_{j,k}$  of a wire. This concept is similar to that used when imposing the global mesh-truncation scheme in the volumetric formulation. Therefore, we permit the following boundary condition:

$$\boxed{\mathcal{F}_{j,k|IM}^o = \bar{a}_{j,k}^+ \cdot \mathbf{u}_{j,k}^+ + \bar{a}_{j,k}^- \cdot \mathbf{u}_{j,k}^{imposed}} \quad (7.48)$$



This type of boundary condition will be used exhaustively in Chapter 8 when a subcell circuit model is introduced. The circuit models will be responsible for determining an imposed incoming solution  $\mathbf{u}_{j,k}^{imposed}$ .

### 7.3.8 Multiple-Wire Junctions

In this section we wish to briefly point out that we could use characteristic theory to develop an appropriate boundary condition when multiple wire-segments intersect at a common terminal. It is easier, however, to interface multiple-wires using the subcell models developed in Chapter 8 and so we will not consider the multiple-wire junction case here.

## 7.4 Spatial Reconstruction Schemes

We will now consider appropriate schemes for interpolating the cell-centered thin-wire values  $\mathbf{u}_{(j)}$  to provide the upwinded values  $\mathbf{u}_{j,k}^+$  at the wire boundaries. The discussion will be somewhat brief, as the majority of the theory presented in Section 3.7 can be modified to the one-dimensional reconstruction problem.

### 7.4.1 First-Order Reconstruction

As in the three-dimensional field case, a simple, first-order interpolation scheme is available by assuming that the upwind interpolated values  $\mathbf{u}_{j,k}^+$  are equal to the barycentric wire values. That is

$$\boxed{\mathbf{u}_{j,k}^+ = \mathbf{u}_{(j)} + O(\Delta\zeta).} \quad (7.49)$$

While this interpolation scheme does offer a numerically stable method for computing the flux through the wire interfaces, as in the volumetric case, its first-order accuracy introduces unacceptable levels of dissipation and an alternative should be considered [11].

#### 7.4.2 Second-Order Minimum Modulus Reconstruction

The second-order accurate reconstruction scheme that is considered herein is Van-Leer's minimum modulus interpolation scheme [49]. This method is one of many so-called flux-limiting schemes and works in a manner similar to the MUSCL scheme for three dimensional interpolation [11]. We present the method for interpolating the voltage  $\mathcal{V}_{j,k}^+$ . The current is interpolated in the same way.

From Taylor's expansion around the barycentre  $\zeta_j$  of  $W_j$ , we may write the boundary value of the voltage  $\mathcal{V}_{j,k}^+$  as:

$$\begin{aligned}\mathcal{V}_{j,k}^+ &= \mathcal{V}(t, \zeta_i) + k \frac{|W_j|}{2} (\partial_\zeta \mathcal{V}(t, \zeta))|_{\zeta=\zeta_j} + O(|\Delta\zeta|)^2 \\ &= \mathcal{V}_{(j)}(t) + k \frac{|W_j|}{2} (\partial_\zeta \mathcal{V}(t, \zeta))|_{\zeta=\zeta_j} + O(|\Delta\zeta|)^2\end{aligned}\tag{7.50}$$

where, going from the first equality to the second, exploits the fact that value at the barycentre  $\zeta_j$  is approximated to second order by the average value over  $W_j$  (see (3.37)). This leaves us with the job of determining the spatial derivative  $\partial_\zeta \mathcal{V}$ . The minimum-modulus scheme constructs three estimates for the derivative: a first-order forward difference  $\Delta_F$ , a first-order backward difference  $\Delta_B$  and a second-order central

difference  $\Delta_C$ :

$$\begin{aligned}\Delta_{F_j} &= \frac{\mathcal{V}_{j+1} - \mathcal{V}_j}{0.5|W_{j+1}| + 0.5|W_j|} \\ \Delta_{B_j} &= \frac{\mathcal{V}_j - \mathcal{V}_{j-1}}{0.5|W_j| + 0.5|W_{j-1}|} \\ \Delta_{C_j} &= \frac{\mathcal{V}_{j+1} - \mathcal{V}_{j-1}}{0.5|W_{j+1}| + |W_j| + 0.5|W_{j-1}|}\end{aligned}\tag{7.51}$$

where the  $\mathcal{V}_j = \mathcal{V}(t, \zeta_j)$  is the voltage at the centroid of  $W_j$ , approximated to second order by the average  $\mathcal{V}_{(j)}$ . A first-order approximation to the derivative is then computed as:

$$(\partial_\zeta \mathcal{V})|_{\zeta=\zeta_j} = \text{minmod}(\Delta_{C_j}, \text{minmod}(\theta \Delta_{B_j}, \theta \Delta_{C_j})) + O(\Delta \zeta)\tag{7.52}$$

where the definition of the minmod function is given by:

$$\text{minmod}(a, b) = \begin{cases} 0 & \text{if } ab < 0 \\ a & \text{if } |a| < |b| \\ b & \text{otherwise} \end{cases}\tag{7.53}$$

and where the free parameter  $\theta \in [1, 2]$  is used to limit the reconstruction in the presence of high-gradients [49]. The product of the derivative approximation (7.52) with  $k|W_j|/2$  in (7.50) maintains a second-order accurate numerical scheme.

### 7.4.3 Higher-Order Reconstruction

As in the volumetric case, it is possible to develop higher-order schemes for spatially reconstructing the solution from each side of a thin-wire segment. In fact, for uniformly discretized wires, it is relatively straight-forward [50, 10], especially when we

limit consideration to smoothly varying currents and charge along the wire and we can neglect the need for shock-capturing, non-oscillatory schemes [50]. Without going into mathematical detail, when only smooth functions are concerned, the reconstructing problem reduces to reproducing a unique polynomial inside  $W_j$  from the average solutions in  $W_j$  and surrounding cells [10]. Its implementation is straightforward.

### 7.5 Thin-Wire to Field-Solver Coupling

To this point, we have developed the solution to the one dimensional wire equations (7.20) by assuming that the source term  $\mathcal{S}_{(j)}(t)$  is zero. As our objective is to provide a subcell thin-wire model for volumetric simulations, we must now address the coupling between the thin-wire model and the volumetric FVTD scheme. As previously discussed, this coupling occurs to a radial distance  $\rho_0$  surrounding each wire. This distance is empirically selected as  $1.7\Delta_V$  where, for uniform volumetric meshes,  $\Delta_V$  is the average edge length in the partition  $\{V_i\}$  [19]. For non-uniform meshes  $\Delta_V$  should be determined from the average edge lengths in proximity to the wires [19]. An analysis of the effects of the definition of  $\rho_0$  on the accuracy of the thin-wire model are beyond the scope of this work.

The coupling mechanism is subdivided into two parts: field-to-wire coupling and wire-to-field coupling and has been shown to be unconditionally stable [18].

#### 7.5.1 Field-to-Wire Coupling

Coupling from the fields to the wires occurs in the thin-wire equations (7.20) through the source term  $\mathcal{S}_{(j)}(t) = [0 \langle \mathcal{E}_\zeta(t, \zeta) \rangle_{(j)}]^T$ . For wire segment  $W_j$  the source is computed

as:

$$\langle \mathcal{E}_\zeta(t, \zeta) \rangle_{(j)} = \frac{1}{|W_j|} \iiint_V \mathcal{E}_\zeta(t, \zeta) g(\rho_j) \Phi_j(\zeta) dV \quad (7.54)$$

where  $\rho_j$  is dependent on the orientation of segment  $W_j$  and where  $\Phi_j(\zeta) = 1$  when  $\zeta \in W_j$  and is zero otherwise [18]. The finite support of both  $g(\rho_j)$  and  $\Phi_j(\zeta)$  limit the integration over the entire computational domain  $V$  to the vicinity of  $W_j$  and (7.54) can be evaluated in constant time. We denote the set of volumetric cells contributing to the integral (7.54) as  $\{V_m^j\} \subset \{V_i\}$ ,  $m = 1, \dots, M_j$ . The integral over each cell in  $\{V_m^j\}$  can be evaluated numerically to arbitrary order, or can be approximated to second order by evaluating (7.54) at the barycentre of each contributing volume. In either case, the electric field component  $\mathcal{E}_\zeta(t, \zeta) = \vec{\mathcal{E}}(t, \vec{r}) \cdot \hat{\zeta}_j$  is determined by reconstructing the electric field  $\vec{\mathcal{E}}(t, \vec{r})$  from cell-averaged field values in the vicinity of  $\vec{r}$ .

### 7.5.2 Wire-to-Field Coupling

Coupling from the wire to the fields is achieved by introducing a volumetric current produced by the wire-segments [18]:

$$\vec{\mathcal{J}}(t, \vec{r}) = \sum_{j=1}^{N_W} \mathcal{I}(t, \zeta) g(\rho_j) \Phi_j(\zeta) \hat{\zeta}_j \quad (7.55)$$

By definition, the contribution from  $W_j$  is restricted to  $\{V_m^j\}$ . Assuming that all impressed currents arise from the thin-wire model, the source term  $\vec{\mathcal{S}}_{(i)}(t)$  required for the volumetric update (3.5) on finite-volume  $V_i$  is computed by evaluating:

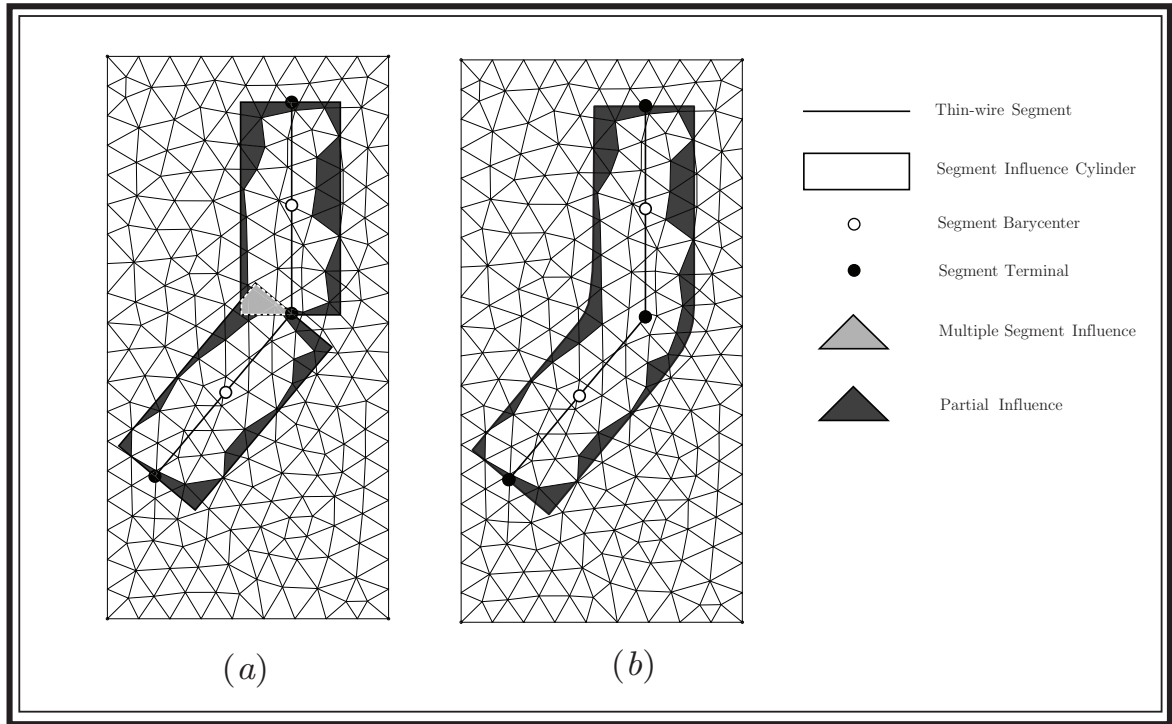
$$\vec{\mathcal{J}}_{(i)}(t, \vec{r}) = \frac{1}{|V_i|} \iiint_{V_i} \sum_{j=1}^{N_W} \mathcal{I}(t, \zeta) g(\rho_j) \Phi_j(\zeta) \hat{\zeta}_j dV \quad (7.56)$$

Once again the integral can be computed using numerical quadrature to arbitrary order, or can be approximated to second order by evaluating the integrand at the barycentre of  $V_i$ . The current  $\mathcal{I}(t, \zeta)$  is reconstructed for arbitrary  $\zeta$  using the stored segment-average values.

### 7.6 Bent Wires, Multi-wire Junctions and Charge Buildup

When curved wires are approximated using straight wire segments, or when multiple segments intersect at a common terminal, the coupling cylinders associated with each wire segment overlap as shown in Figure 7.3 (a). Under these circumstances, the distributed current density (7.56) is no longer spatially continuous and may result in an unphysical build-up of charge within the volumetric domain as shown by the light grey region in the figure. In [18, 19] the charge build-up manifests itself as an *undamped low-frequency oscillation* in the numerical simulation of thin-wire loops.

For continuously curved wires, the interpolation cylinders can be modified to ensure that the resulting current-density is continuous in the interpolation cylinder as shown in Figure 7.3 (b). In our implementation, we adopt the geometry bending approach, provided with complete details in [19]. For junctions involving three or more wire segments, enforcing a continuous current density is difficult. While Chapter 8 will provide a mechanism for interfacing an arbitrary number of thin-wire segments connected at a single point *in the absence of field-coupling*, we have not yet been able to solve the problem of discontinuous currents at multi-wire junctions and further research is required. This does not mean that multi-wire junctions cannot be simulated – we may allow the wire-segments to overlap and try to reduce the charge-build up by selecting time-functions with minimal low-frequency components. This is not the ideal solution,



*Fig. 7.3:* A 2D representation of the thin-wire to volume coupling influence. (a) Without modification, the coupling cylinders of connected, but bent, wire segments overlap resulting in discontinuous spatial current and charge build-up. (b) For two wires, the cylinder can be modified to ensure spatial continuity of the volumetric current density.

and a resolution to this problem will be the focus of future work. In addition to these problems, even for straight wires the coupling between segments and cells can result in unphysical discontinuous current densities when a finite-volume is only partially influenced by a segment as shown by the dark shaded areas in Figure 7.3 (a) and (b). In these cells, the forcing term (7.56) is non-zero only if the numerical quadrature point intersects the interpolation cylinder and is therefore dependent on the distribution of quadrature points within the influenced cell.

As will be shown in Chapter 9, in many cases the numerical artifacts associated with discontinuous currents in the interpolation cylinders do not prohibit the computationally economic extraction of broad-band system parameters. A rigorous study of the

effects of charge-buildup on problems sensitive to a lack of charge-conservation (*e.g.* near-field measurements in the proximity of multi-wire junctions) is reserved for future work.

### 7.7 Time Integration

The local system of equations (7.20) on each wire-segment  $W_j$  can be written in system form analogous to the volumetric system (3.46) as:

$$\boxed{d_t \underline{\mathbf{u}}(t) = -\underline{\mathcal{F}} \cdot \underline{\mathcal{R}} \cdot \underline{\mathbf{u}}(t) + \underline{\Sigma} \cdot \underline{\mathbf{u}}(t) + \underline{\mathcal{S}}(t)} \quad (7.57)$$

where the sparse global matrix  $\underline{\mathcal{F}}$  is understood to account for the flux integration and operates on the boundary states obtained by operating with the sparse reconstruction matrix  $\underline{\mathcal{R}}$  on the cell-averages. The diagonal matrix  $\underline{\Sigma}$  accounts for the loss terms. As this system is in precisely the same form as (3.46), all of the explicit time-integration schemes developed in Section 4.1 apply.

Global stability on the wire-segments is ensured provided the local time-step  $\Delta t_j$  given by:

$$\boxed{\Delta t_j \leq \frac{|W_j|}{c_j}} \quad (7.58)$$

is satisfied on each wire segment  $W_j$  [11]. Therefore, for a global time-stepping scheme, we must select the minimum of either the volumetric time-step or the wire time-step.

For time-integration in the context of the volumetric local-time-stepping (LTS) scheme presented in Section 4.3, we assume, for the sake of simplicity, that all connected wire-segments and the volumetric elements they couple to, exist at the same LTS level. This



can easily be ensured by imposing this constraint on the LTS pre-processing algorithm detailed in Section 4.3.1.

## 7.8 Chapter Summary

In this chapter, a thin-wire subcell model that can be arbitrarily oriented within the volumetric computational domain, has been developed. The upwind and flux-split FVTD solution to the resulting thin-wire equations has been provided in detail. In the next chapter we present a novel method for incorporating subcell circuits that can be used to interconnect, terminate and/or drive the thin-wire equations providing an elegant mechanism for the extraction of broad-band engineering system parameters. We reserve numerical validation of the thin-wire and circuit models to Chapter 9.

# FVTD Thin-Wire Boundary Conditions by Subcell Lumped-Circuit Models

---

*There are no surprising facts, only models that are surprised by facts; and if a model is surprised by the facts, it is no credit to that model. – Eliezer Yudkowsky*

In many applications it is desirable to interface thin-wires with lumped element circuits. Examples include sourcing a voltage onto an antenna [51]; modeling tunable radio-frequency (RF) coils used in Magnetic Resonance Imaging (MRI) applications, where thin-wires are connected in series or parallel to small tuning capacitors [52, 53]; and driving transmission lines with sub-cell circuits to determine system electromagnetic susceptibility to ambient electromagnetic field coupling to the transmission lines [54].

In this chapter a lumped-element subcell circuit model used to terminate, drive and/or interconnected thin-wire segments, is developed. The theory permits a unification of the three levels of electromagnetic analysis: circuit modeling, transmission-line modeling and electromagnetic field modeling. Motivated by the work in [55], we solve the coupled circuit-wire system using a modified nodal analysis (MNA) approach [56] and the thin-wire characteristic theory developed in Section 7.2.7. The resulting numerical scheme supports multiport networks, modeled as subcell circuits, to be connected to multiple thin-wire elements.

This chapter begins by describing the nodal analysis (NA) and modified nodal analysis (MNA) circuit solving techniques that are used to automate the construction of the circuit equations. Once the basic circuit-solving concepts have been covered, the appropriate coupling mechanism between the thin-wire equations and the MNA circuit description is developed. Finally, an explicit solution method for solving the coupled thin-wire/circuit equations is provided.

## 8.1 Circuit Solutions

Circuit analysis methods are based on the concepts of branch currents, nodal voltages, and the so-called *constitutive equation* relating a branch current and branch voltage (the later being the difference between the nodal voltages terminating a branch). Assuming we have a circuit branch denoted by  $Q_{pp'}$  between nodes  $P_p$  and  $P_{p'}$  having respective nodal voltages  $v_p$  and  $v_{p'}$  we can write a general form of a branch constitutive equation as

$$Y_{pp'}v_{pp'}(t) - Z_{pp'}i_{pp'}(t) = f_{pp'}(t) \quad (8.1)$$

Branch Type	Constitutive Equation	$Y_{pp'}$	$Z_{pp'}$	$f_{pp'}$
Resistor	$v_{pp'} - Ri_{pp'} = 0$	1	$-R$	0
Short Circuit	$v_{pp'} = 0$	1	0	0
Conductor	$Gv_{pp'} - i_{pp'} = 0$	$G$	$-1$	0
Open Circuit	$-i_{pp'} = 0$	0	$-1$	0
Capacitor	$sCv_{pp'} - i_{pp'} = Cv_{pp'}(0)$	$sC$	$-1$	$Cv_{pp'}(0)$
Inductor	$v_{pp'} - sLi_{pp'} = -Li_{pp'}(0)$	1	$-sL$	$-Li_{pp'}(0)$
IVS	$v_{pp'} = v_s(t)$	1	0	$v_s(t)$
ICS	$i_{pp'} = i_s(t)$	0	1	$i_s(t)$

Tab. 8.1: Constitutive relationships for some common circuit branches. The acronyms IVS and ICS respectively refer to independent voltage sources and independent current sources.

where  $v_{pp'} = v_p - v_{p'}$  is the branch voltage,  $i_{pp'}$  is the branch current flowing from  $P_p$  to  $P_{p'}$ ,  $Z_{pp'}$  is an impedance,  $Y_{pp'}$  is an admittance and  $f_{pp'}$  is some known forcing quantity. Introducing the Laplace variable  $s$  for the time-derivative, we summarize the constitutive equations for some common circuit elements in Table 8.1 [56]. For the capacitor and inductor, the forcing term  $f_{pp'}$  given by  $Cv_{pp'}(t = 0)$  and  $-Li_{pp'}(t = 0)$  respectively, denote the initial conditions across the branch. For simplicity in what follows we will always assume that at time  $t = 0$  no energy is stored in either capacitors or inductors.

### 8.1.1 Nodal Analysis

Nodal analysis (NA) is a very simple, but powerful, technique used to solve circuit equations. Topologically, the circuit is described by a set of  $P + 1$  nodes  $\{P_p\}, p = 0, \dots, P$ , where  $P_0$  is a reserved reference node associated with zero potential. Each node assumes a corresponding nodal voltage value  $v_p$  with  $v_0 = 0$  [V]. Nodal analysis seeks to uniquely determine the nodal voltages by enforcing Kirchoff's Current Law

(KCL) at each node. KCL states that the sum of all branch currents leaving node  $P_p$  is equal to zero. To enforce KCL at node  $P_p$  we denote the set of  $q_{pp'}$  branches between nodes  $P_p$  and  $P_{p'}$  as  $\{Q_{pp'}^q\}$ ,  $q = 1, \dots, q_{pp'}$  where  $q$  uniquely identifies each branch. NA assumes that the branch current  $i_{pp'}^q$  flowing from  $P_p$  to  $P_{p'}$  along the branch  $Q_{pp'}^q$  can be explicitly obtained from the branch constitutive equation. Then, KCL at node  $P_p$  can be written as:

$$P_p : \sum_{\substack{p'=0 \\ p' \neq p}}^{P-1} \sum_{q=1}^{q_{pp'}} i_{pp'}^q = 0 \quad (8.2)$$

If a branch  $Q_{pp'}^q$  does not exist, it can be interpreted as an open circuit and according to the constitutive equation in Table 8.1 the current  $i_{pp'}^q$  is zero and does not contribute to the summation.

Nodal analysis assumes an explicit equation for  $i_{pp'}^q$ . Inspection of Table 8.1 indicates that describing  $i_{pp'}^q$  explicitly is possible for conductors, open-circuits, capacitors and independent current sources. When the constitutive equation does not involve the branch current (short circuits and independent voltage sources) or involves the time-derivative of the current (inductors), nodal analysis cannot be applied directly and a modification is required. For finite, non-zero, resistive branches the current is obtained explicitly by converting the resistance to a conductance. As an example, nodal-analysis can be used to solve the circuit presented in Figure 8.1. The circuit consists of three nodes  $P_0$ ,  $P_1$  and  $P_2$ . The ground node  $P_0$  is assumed to be at 0 [V] while the values of the nodal voltages  $v_1$  and  $v_2$  constitute the nodal-analysis solution to the circuit. Satisfying KCL at each node gives the following equations:

$$\begin{aligned} P_1 : -i_s(t) + i_{12} &= -i_s(t) + G(v_1 - v_2) = 0 \\ P_2 : -i_{12} + i_{20} &= (v_2 - v_1)G + sC(v_2 - 0) = 0 \end{aligned} \quad (8.3)$$

As a general circuit can be much more complicated than this simple example, we write

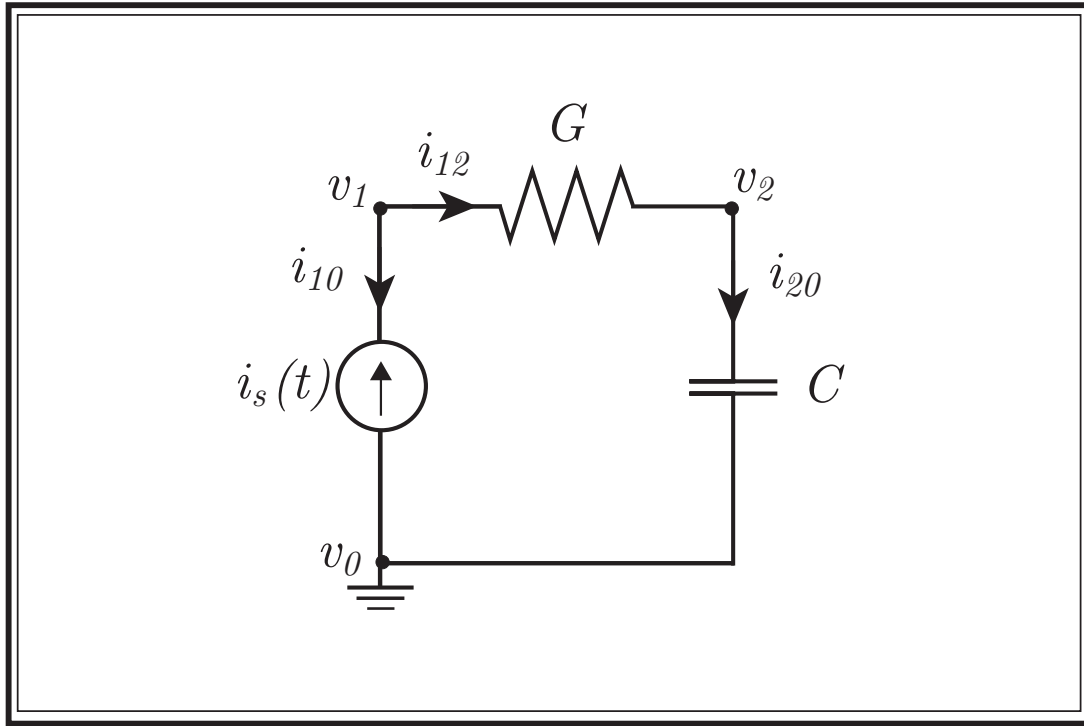


Fig. 8.1: A current-driven RC circuit.

the nodal equations as a system of equations to illustrate the general procedure:

$$\begin{bmatrix} G & -G \\ -G & G \end{bmatrix} \cdot \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} + \frac{d}{dt} \begin{bmatrix} 0 & 0 \\ 0 & C \end{bmatrix} \cdot \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} i_s(t) \\ 0 \end{bmatrix} \quad (8.4)$$

where the time-derivative has been reinstated for the Laplace variable  $s$ . The time-marching solution to this system of equations constitutes the nodal-analysis solution to the time-domain circuit.

### 8.1.2 Modified Nodal Analysis

Modified nodal analysis (MNA) is an extension of nodal-analysis that permits for branch constitutive equations that either *a*) do not involve the branch current (short

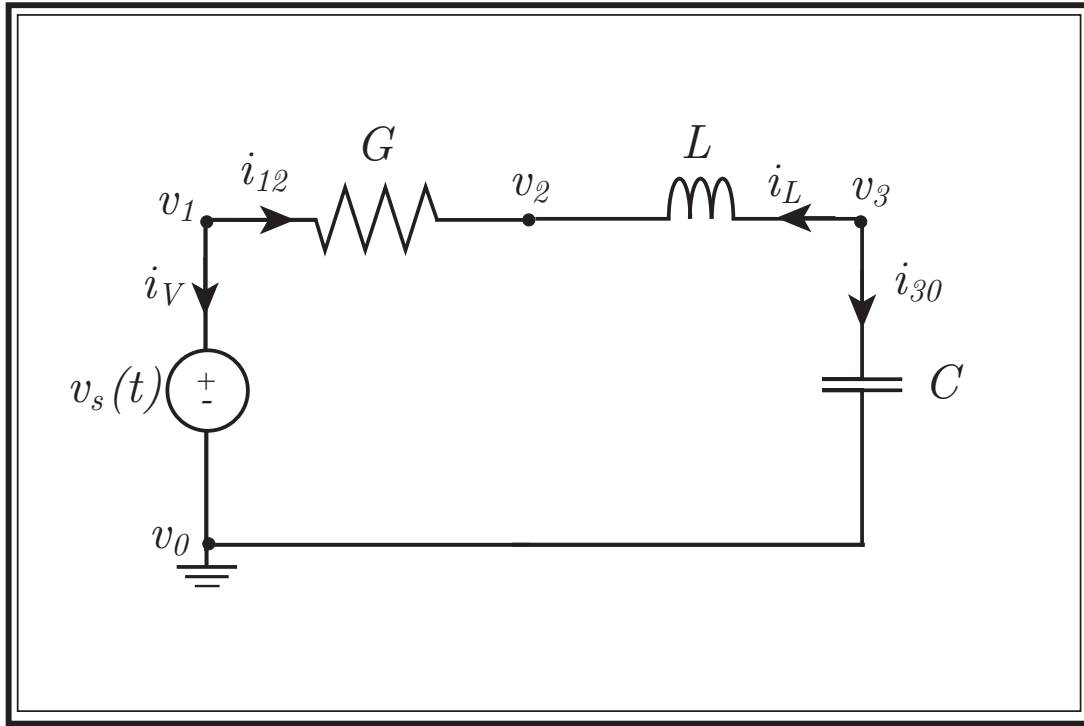


Fig. 8.2: A voltage-driven RLC circuit.

circuits and independent voltage sources) or *b*) involves the time-derivative of the current (inductors). For each of these circuit elements, an *auxiliary current* is assumed to exist and KCL is satisfied using the assumed current. For each auxiliary current introduced into the system we must determine an additional *auxiliary equation* such that new system still permits a unique solution. The constitutive equation for the branch serves as the auxiliary equation, but as we have introduced an auxiliary current there is no need to solve the constitutive equation for the current. In order to enable writing a system of equations for solving the circuit under the modified nodal analysis formulation, a unique identifier  $m = P + 1, \dots, P + M$  is associated with each of the  $M$  branches that require an auxiliary equation. In this way we can uniquely identify the equations to be satisfied by nodal analysis:  $P$  equations (ordered  $1, \dots, P$ ) for each KCL equation at the nodes  $P_p$  and  $M$  auxiliary equations. As an example, we consider the serial *RLC* circuit depicted in Figure 8.2 driven by an independent voltage

source. The circuit consists of four nodes  $P_0$  (assumed to be at zero potential),  $P_1$ ,  $P_2$  and  $P_3$ . The independent voltage source and the inductor require introducing two auxiliary currents  $i_V$  and  $i_L$  which are assumed to have reference directions as shown in Figure 8.2. The branch corresponding to the voltage source is assigned auxiliary index  $m = P + 1 = 4$  while the branch corresponding to the inductor is assigned auxiliary index  $m = P + 2 = 5$ . Satisfying KCL at each node then gives:

$$\begin{aligned} \text{Equation 1 at } P_1 : i_V + i_{12} &= i_V + (v_1 - v_2)G = 0 \\ \text{Equation 2 at } P_2 : -i_{12} - i_L &= (v_2 - v_1)G - i_L = 0 \\ \text{Equation 3 at } P_3 : i_L + i_{30} &= i_L + sCv_3 = 0 \end{aligned} \quad (8.5)$$

These three equations can be written as a system involving the *five* unknown quantities  $v_1$ ,  $v_2$ ,  $v_3$ ,  $i_V$  and  $i_L$ . The system is divided into two matrices, one for frequency independent values and one for frequency dependent values. The auxiliary currents are stored in the vector of unknowns at the assigned auxiliary index  $m$ :

$$\begin{bmatrix} G & -G & 0 & 1 & 0 \\ -G & G & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ i_V \\ i_L \end{bmatrix} + \frac{d}{dt} \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & C & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ i_V \\ i_L \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad (8.6)$$

This system of equations is under-determined. In order to solve it we impose the auxiliary equations. In this case the two auxiliary equations are the general constitutive equations for the voltage source and inductor in Table 8.1:

$$\begin{aligned} \text{Equation 4 (Auxiliary 1) } i_V : v_1 &= v_s(t) \\ \text{Equation 5 (Auxiliary 2) } i_L : v_{32} &= (v_3 - v_2) = sLi_L \end{aligned} \quad (8.7)$$



According to our auxiliary indexing convention we add these auxiliary equations at the end of the MNA system

$$\begin{bmatrix} G & -G & 0 & 1 & 0 \\ -G & G & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ i_v \\ i_L \end{bmatrix} + \frac{d}{dt} \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & C & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -L \end{bmatrix} \cdot \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ i_v \\ i_L \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ v_s \\ 0 \end{bmatrix} \quad (8.8)$$

where once again the time-derivative has been re-instated for the variable  $s$ .

## 8.2 MNA System Construction by Branch Stamps

For solving circuits involving an arbitrary number of branches and nodes, it is not desirable to have to manually enforce KCL at each node. Fortunately, there are a number of ways of automating the construction of an MNA system [56]. One straightforward way (and the method adopted herein) is to use the concept of *branch stamps* [56]. By generalizing the example of the previous section, the modified nodal analysis formulation of a general circuit consisting of  $P + 1$  nodes (including ground) and  $M$  auxiliary branches can be written as a system

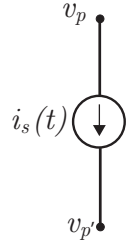
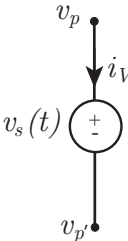
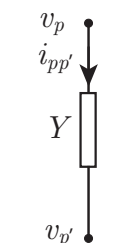
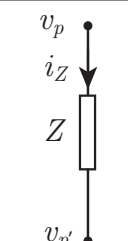
$$\bar{\bar{G}} \cdot \vec{U} + \bar{\bar{C}} \cdot \frac{d}{dt} \vec{U} = \vec{S} \quad (8.9)$$

where the  $(P + M) \times (P + M)$  matrices  $\bar{\bar{G}}$  and  $\bar{\bar{C}}$  respectively contain the frequency independent and frequency dependent parts of the MNA equations. The  $(P + M)$ -length solution vector  $\vec{U}$  consists of the  $P$  nodal voltages followed by the  $M$  auxiliary currents and the  $(P + M)$ -length source vector  $\vec{S}$  contains the source terms. For

a general circuit the construction of the matrices  $\bar{\bar{G}}$  and  $\bar{\bar{C}}$ , as well as the vector  $\vec{S}$ , can be automated. Given a branch  $Q_{pp'}$  connecting nodes  $P_p$  and  $P_{p'}$  (and, if necessary, an auxiliary index  $m$ ), a branch stamp provides a quick way of determining the contributions of the branch to the MNA matrices and source vector.

Branch stamps for some circuit elements are provided in Table 8.2 and are used as follows. In the *Stamp* column, a stamp matrix and stamp vector are provided. The matrices have dimension  $2 \times 2$  if no auxiliary equation is required or  $3 \times 3$  otherwise. The source vectors have the same number of rows as the stamp matrix. The entries in the stamp matrix and stamp vector are to be added to the MNA system matrices and source vector using the global row indexing  $p$ ,  $p'$  and  $m$  provided beside the rows of the stamp matrix. These indexes correspond to the nodal indexes  $p$ ,  $p'$  and the auxiliary equation  $m$  in the global circuit system. The same indexing is assumed across the columns, which are labeled by the circuit quantity they will multiply in the global system. Inherent in the stamps for current and voltage sources is an orientation defined by the source. For the admittance and impedance stamps, only the frequency dependent impedances (inductors) and admittances (capacitors) are placed in  $\bar{\bar{C}}$  while the rest of the stamp is placed in  $\bar{\bar{G}}$ .

To illustrate the use of the branch stamps for building the MNA circuit equations we will reconsider the circuit of Figure 8.2. We begin by constructing the matrices  $\bar{\bar{G}}$ ,  $\bar{\bar{C}}$  and the vector  $\vec{S}$  and initializing them to zero. We then consider each branch in the circuit and add the contribution of the branch stamp to the appropriate matrix and source vector entries. Beginning with the voltage-source branch, according to the voltage-source stamp we get:

Branch Element	Symbol	Stamp Matrix (Left) Source Vector (Right)	Equations
Current Source		$  \begin{array}{c} v_p \quad v_{p'} \\ p \\ p' \\ m \end{array}  \begin{bmatrix} & & \\ & & \\ & & \\ 1 & -1 & \end{bmatrix}  \begin{bmatrix} -i_s(t) \\ i_s(t) \\ \end{bmatrix}  $	$  \begin{aligned}  i_{pp'} &= i_s(t) \\  i_{p'p} &= -i_s(t)  \end{aligned}  $
Voltage Source		$  \begin{array}{c} v_p \quad v_{p'} \quad i_V \\ p \\ p' \\ m \end{array}  \begin{bmatrix} & & 1 \\ & & -1 \\ 1 & -1 & \end{bmatrix}  \begin{bmatrix} \\ \\ v_s(t) \end{bmatrix}  $	$  \begin{aligned}  v_p - v_{p'} &= v_s(t) \\  i_{pp'} &= i_V \\  i_{p'p} &= -i_V  \end{aligned}  $
Admittance		$  \begin{array}{c} v_p \quad v_{p'} \\ p \\ p' \\ m \end{array}  \begin{bmatrix} Y & -Y \\ -Y & Y \end{bmatrix}  \begin{bmatrix} \\ \\ \end{bmatrix}  $	$  \begin{aligned}  i_{pp'} &= Y(v_p - v_{p'}) \\  i_{p'p} &= -Y(v_p - v_{p'})  \end{aligned}  $
Impedance		$  \begin{array}{c} v_p \quad v_{p'} \quad i_Z \\ p \\ p' \\ m \end{array}  \begin{bmatrix} & & 1 \\ & & -1 \\ 1 & -1 & -Z \end{bmatrix}  \begin{bmatrix} \\ \\ \end{bmatrix}  $	$  \begin{aligned}  v_p - v_{p'} - Zi_Z &= 0 \\  i_{pp'} &= i_Z \\  i_{p'p} &= -i_Z  \end{aligned}  $

Tab. 8.2: Branch stamps for some common linear circuit elements.

$$\bar{\bar{G}} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad \bar{\bar{C}} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad \bar{\bar{S}} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ v_s \\ 0 \end{bmatrix} \quad (8.10)$$

Due to the fact that the voltage source is connected to node 0 (which is not part of the MNA construction) we see that the stamp contribution is simplified. Using the admittance stamp for the conductive branch contributes to  $\bar{\bar{G}}$  giving:

$$\bar{\bar{G}} = \begin{bmatrix} G & -G & 0 & 1 & 0 \\ -G & G & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad \bar{\bar{C}} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad \bar{\bar{S}} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ v_s \\ 0 \end{bmatrix} \quad (8.11)$$

Using the impedance stamp for the inductive branch and adding the frequency dependent part of the stamp to  $\bar{\bar{C}}$  gives:

$$\bar{\bar{G}} = \begin{bmatrix} G & -G & 0 & 1 & 0 \\ -G & G & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 \end{bmatrix}, \quad \bar{\bar{C}} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -L \end{bmatrix}, \quad \bar{\bar{S}} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ v_s \\ 0 \end{bmatrix} \quad (8.12)$$

Finally, applying the admittance stamp to the capacitive branch contributes to  $\bar{\bar{C}}$

giving:

$$\vec{\bar{G}} = \begin{bmatrix} G & -G & 0 & 1 & 0 \\ -G & G & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 \end{bmatrix}, \quad \vec{\bar{C}} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & C & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -L \end{bmatrix}, \quad \vec{\bar{S}} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ v_s \\ 0 \end{bmatrix} \quad (8.13)$$

The final system matrices and source vector in (8.13) produced by using the branch stamp approach is identical to the system (8.8) produced by enforcing KCL at each node and the auxiliary equations at each auxiliary branch. MNA systems for general circuit analysis can quickly and efficiently be constructed using branch stamps.

### 8.3 Thin-Wire Subcell Models Terminated at Subcell Circuits

Having developed the required circuit-theory, we can now present the primary focus of this chapter: terminating, driving and/or interfacing the thin-wire subcell model presented in Chapter 7 with lumped-element, subcell circuits. Explicit in the definition of a subcell circuit, is that the circuit dimensions are much smaller than the minimum wavelength of interest, and therefore subcell circuits can be introduced into the volumetric mesh at a single point. For the purpose of volumetric field simulations, we permit these circuit points to correspond to wire-segment terminals as shown, for example, by the double-box in Figure 8.3. This example circuit exists at terminal  $T_{j,1} = T_{\ell,-1}$  connecting segments  $W_j$  and  $W_\ell$ . In the absence of thin-wires, and independent of the actual circuit used, the circuit can be solved by the general procedure detailed in the previous sections. We must therefore present a method for systematically interfacing the wire equations and the circuit equations. We recall, that for

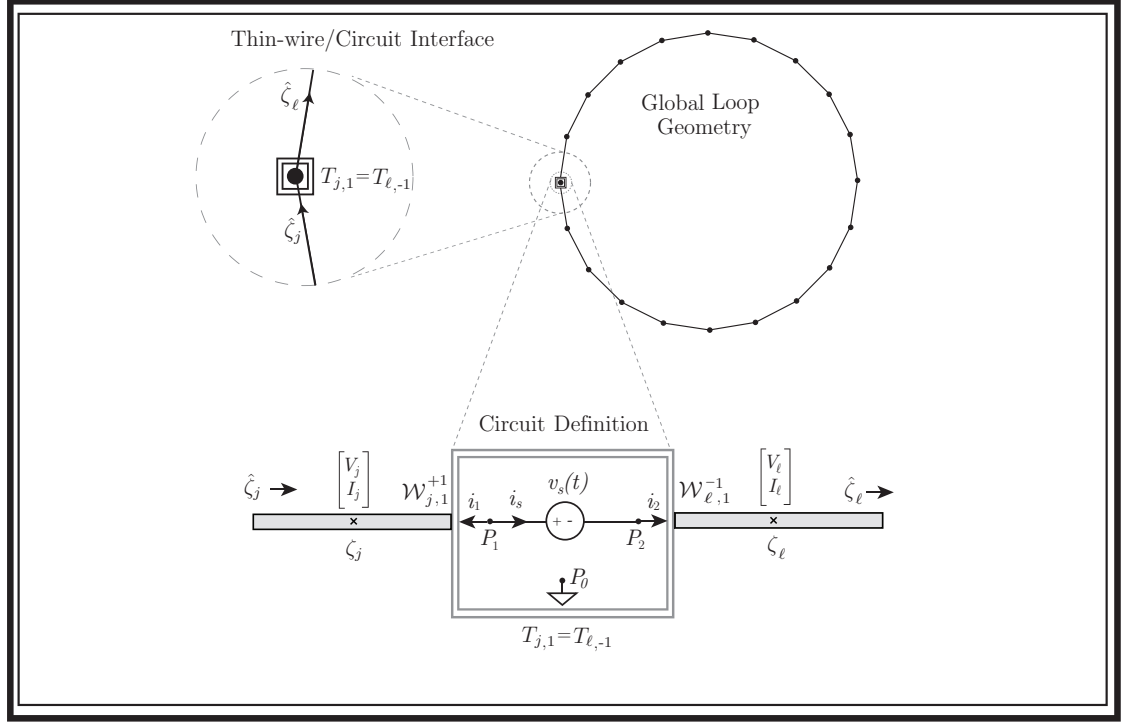


Fig. 8.3: A circuit-driven thin-wire loop antenna.

wire-segment  $W_j$ , the one-dimensional FVTD solution is given by 7.20, repeated here for convenience:

$$d_t \mathbf{u}_{(j)}(t) = - \frac{1}{|W_j|} \sum_{k=-1}^1 \mathcal{F}_{j,k}^o(t, \zeta_{j,k}) - \bar{b}_j \cdot \mathbf{u}_{(j)}(t) + \bar{c}_j \cdot \mathbf{s}_{(j)}(t) \quad (8.14)$$

and that we can allow for the incoming contribution of the upwind and flux-split boundary flux  $\mathcal{F}_{j,k}^o$  to be imposed according to (7.48), also repeated here:

$$\mathcal{F}_{j,k|IM}^o = \bar{a}_{j,k}^+ \cdot \mathbf{u}_{j,k}^+ + \bar{a}_{j,k}^- \cdot \mathbf{u}_{j,k}^{imposed} \quad (8.15)$$

where the imposed solution can be written as  $\mathbf{u}_{j,k}^{imposed} = [\mathcal{V}_{i,k}^{imposed} \mathcal{I}_{i,k}^{imposed}]^T$ . By terminating a thin-wire at a subcell circuit, we will be looking to determine an appropriate expression for  $\mathbf{u}_{j,k}^{imposed}$  as the solution to the subcell circuit at time  $t$ . The proce-

cedure for determining  $\mathcal{U}_{j,k}^{imposed}$  is most easily demonstrated by example. Consider the thin-wire-coupled lumped-element circuit depicted in Figure 8.4 (a). In the absence of thin-wires, this circuit is the same as that shown in Figure 8.1 and solved by (8.4). In the figure, thin-wire finite-volumes are denoted by gray rectangles and the thin-wire orientation is in the  $\zeta$  direction. To connect the thin-wire to the circuit we will convert Figure 8.4 (a), to the circuit in Figure 8.4 (b). That is, *from the perspective of the circuit*, we will replace the thin-wire with an auxiliary current  $i_{TW}$ . By introducing an auxiliary current, we will require an additional auxiliary equation. This will be discussed in more detail momentarily. First, however, we must ask ourselves under what conditions the circuit in Figure 8.4 (b) models Figure 8.4 (a).

### 8.3.1 Physical Considerations

As the motion of charge, current is a physical quantity. Therefore there is no difficulty in assuming that a current (motion of charge) in the circuit has the same definition as the current (motion of charge) in the thin-wire model. However, the definitions of voltage used in the two models, are not consistent. In the circuit, the voltage is referenced to an arbitrary node. In the thin-wire model, the voltage is referenced to the empirically selected cylinder radius  $\rho_0$  as discussed in Section 7.2.3. For the two models to be compatible we must make the following fundamental assumption:

- **Assumption** When a thin-wire is connected to a circuit, at the wire/circuit boundary the thin-wire voltage can be approximated as the difference between two nodal voltages in the circuit.

This assumption is required to ensure that the circuit and thin-wire models are referenced in a unified fashion and we should consider what it implies. On one hand, it

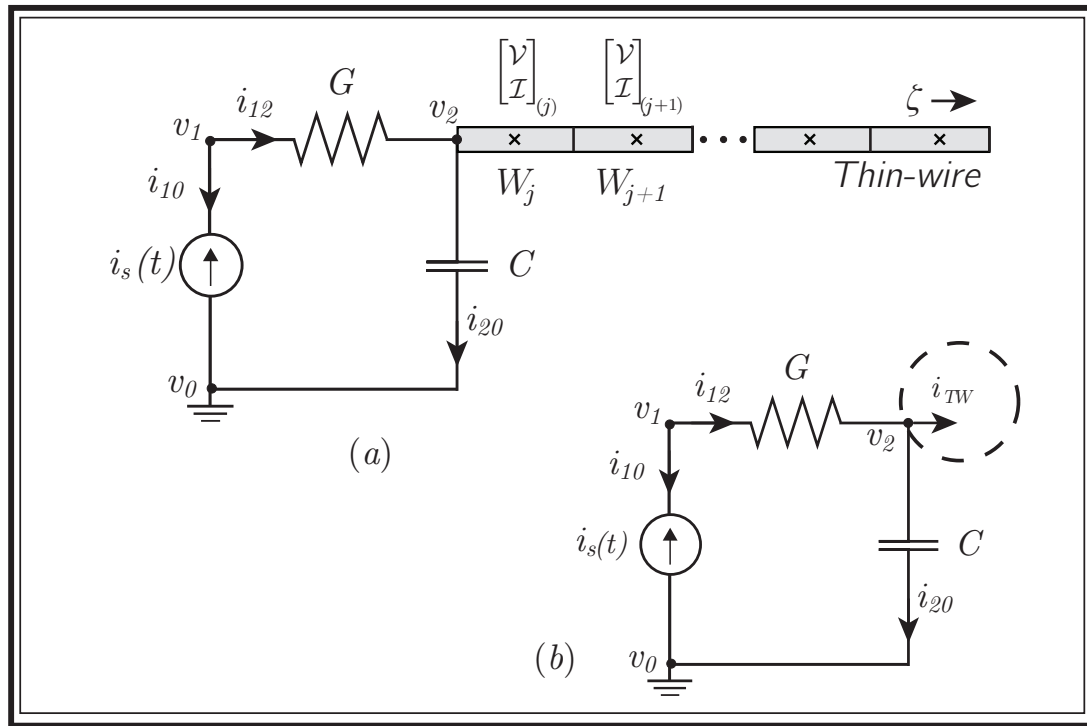


Fig. 8.4: Interfacing a single thin-wire to a circuit. The circuit/wire geometry shown in (a) is included in the circuit by the modification shown in (b).

appears that this assumption is quite limiting as the distance  $\rho_0$ , taken as 1.7 times the average edge length of the volumetric partition  $\{V_i\}$ , is mesh-dependent. For the circuit in Figure 8.4 this implies that the reference node  $v_0$  would be tied to  $\rho_0$  and could adversely affect simulations. Fortunately, there are a large class of situations where this assumption does not pose any real constraint. For example, the thin-wire loop in Figure 8.3 is driven differentially, and both the circuit and the wires connected to it, can be referenced to  $v_0$  at  $\rho_0$ . Brief consideration will confirm that this is true for any differential circuit. Furthermore, this assumption also holds whenever multiple thin-wires are connected, without circuit branches, at a single node as shown in Figure 8.5.



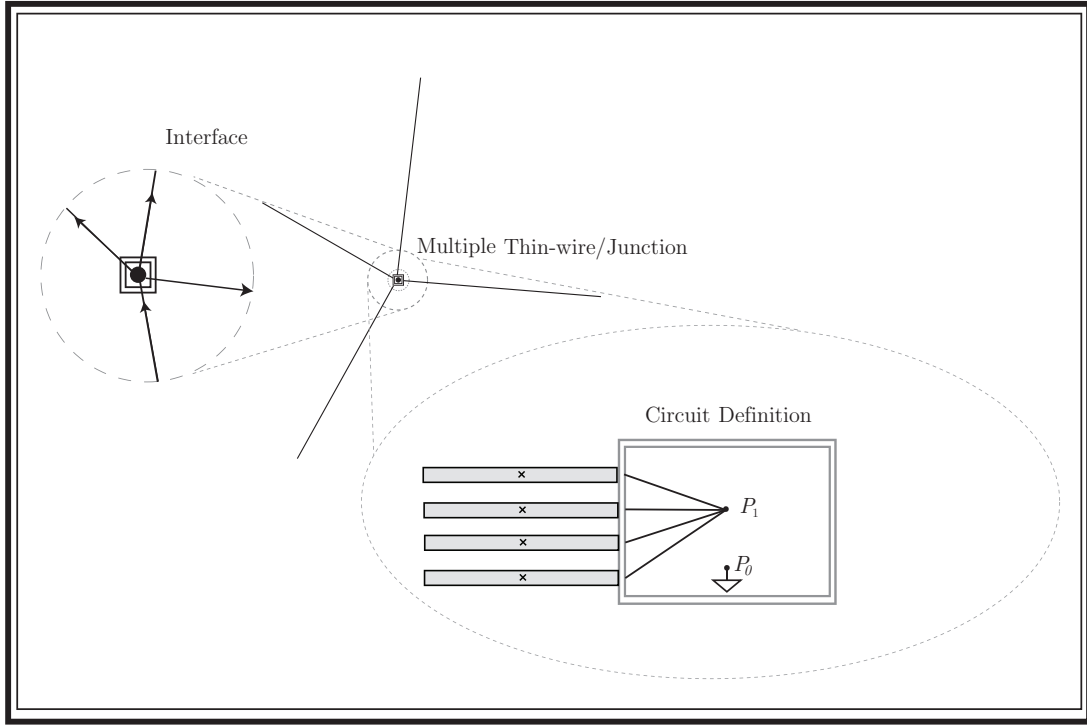


Fig. 8.5: Multiple thin-wire segments terminating at a circuit.

### 8.3.2 The Thin-Wire Auxiliary Equation

We now return to the problem of determining an appropriate auxiliary equation for the assumed auxiliary current  $i_{TW}$  for the example of Figure 8.4. In general, we can permit an arbitrary number of thin-wires to terminate at the circuit. Therefore:

1. For each thin-wire segment connected to a node  $P_p$  in the circuit we introduce an auxiliary current leaving node  $P_p$ .
2. Corresponding to the thin-wire auxiliary currents, we introduce auxiliary equations based on the characteristics of the thin-wire equations. In Section 7.2.7 it was shown that the characteristics

$$\mathcal{W}_{j,\pm 1}(t) = (\mathcal{V}_j(t) \pm Z_j \mathcal{I}_j(t))/2 \quad (8.16)$$

propagate unchanged along segment  $j$  in the  $\pm\zeta$  direction. The fundamental assumption that the thin-wire voltage is consistent with the circuit voltage, allows us to write the characteristics flowing into the circuit in terms of the nodal voltages and auxiliary currents at the wire/circuit interface and provides a straightforward way to obtain the appropriate auxiliary equation.

For the circuit given in Figure 8.4, the characteristic flowing into the circuit at the  $k = -1$  end of  $W_j$  is  $\mathcal{W}_{j,-1}^- = (\mathcal{V}_{j,-1} - Z_j \mathcal{I}_{j,-1})/2$ . We can therefore introduce the thin wire auxiliary equation:

$$v_2 - Z_j i_{TW} = \mathcal{W}_{j,-1}^- \quad (8.17)$$

relating the nodal voltages and auxiliary thin-wire current  $i_{TW}$  to the incoming characteristic. Here we have assumed that the thin wire voltage is equal to the difference between  $v_2$  and  $v_0$ . Implicit in all of this discussion is that we are working at time  $t$ , or in the discrete case  $n\Delta t$ . If the incoming characteristic is known at  $n\Delta t$ , equation (8.17) provides the required additional auxiliary equation relating  $i_{TW}$  to the nodal voltages. Augmenting the system (8.4) to the wire-free version of the circuit in Figure 8.4 with the thin-wire auxiliary current and equation we obtain:

$$\begin{bmatrix} G & -G & 0 \\ -G & G & 0 \\ 0 & 1/2 & -Z_j/2 \end{bmatrix} \cdot \begin{bmatrix} v_1 \\ v_2 \\ i_{TW} \end{bmatrix} + \frac{d}{dt} \begin{bmatrix} 0 & 0 & 0 \\ 0 & C & 0 \\ 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} v_1 \\ v_2 \\ i_{TW} \end{bmatrix} = \begin{bmatrix} i_s(t) \\ 0 \\ \mathcal{W}_{j,-1}^-(t) \end{bmatrix} \quad (8.18)$$

The system of equations (8.18) completely determines the behaviour of the coupled thin-wire, lumped-circuit system. *For any circuit, the auxiliary equations for the thin-wires are determined this way, by using the incoming characteristic as a source, and relating the thin-wire current and voltage to the auxiliary current and nodal voltages of the circuit.*

As a final illustrating example, we consider the example circuit in Figure 8.3 where the MNA system can be written as:

$$\begin{bmatrix} 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & -1 \\ 1 & 0 & -Z_j & 0 & 0 \\ 0 & 1 & 0 & -Z_\ell & 0 \\ 1 & -1 & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} v_1 \\ v_2 \\ i_1 \\ i_2 \\ i_s \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 2\mathcal{W}_{j,1}^{+1}(t) \\ 2\mathcal{W}_{\ell,-1}^{-1}(t) \\ v_s(t) \end{bmatrix} \quad (8.19)$$

where  $\bar{C}$  is zero. The first two equations enforce KCL at nodes  $P_1$  and  $P_2$ . The third and fourth (auxiliary) equations enforce the constitutive relationship between the circuit quantities and the thin-wire characteristics. The last equation is an auxiliary relationship between the voltage source and the nodal voltages. The solution to matrix equation uniquely determines the nodal voltages and auxiliary currents at time  $t$ .

Whenever a thin-wire segment  $j$  is connected to a circuit on facet  $k$ , the imposed incoming solution  $\mathbf{u}_{j,k}^{imposed}$  required in (8.15) can be obtained directly from the circuit solution by simple book-keeping. In the case of the differentially-driven thin-wire loop shown in Figure 8.3 the appropriate imposed solutions are:

$$\mathbf{u}^{imposed}(t, \zeta_{j,1}) = \begin{bmatrix} v_1 \\ -i_1 \end{bmatrix} \quad \mathbf{u}^{imposed}(t, \zeta_{\ell,-1}) = \begin{bmatrix} v_2 \\ i_2 \end{bmatrix}. \quad (8.20)$$

The above method, though illustrated by example, is completely general. An arbitrary number of thin-wire segments can be connected to an arbitrary circuit. The construction of the matrix equation, incoming characteristics and segment fluxes, is easily automated with minimal book-keeping. In addition, the time-history of the circuit solution can be stored to provide a simple mechanism for computing the input impedance

to the entire electromagnetic system. It remains to discuss the time-synchronization between the circuit solver and the thin-wire model.

### 8.4 Time-Synchronization

As discussed in Section 7.7, we assume that the thin-wires and volumetric update occur synchronously and we only need to consider time-synchronization between the thin-wire model and the circuit-solver. The key step here is to observe that, for any of the explicit time-integration schemes discussed in Section 4.1, the thin-wire flux integration required in (8.14) must be evaluated at the current time  $n\Delta t$  where the thin-wire solution is known. Therefore we can consider the general circuit/thin-wire system (8.9) evaluated at time  $n\Delta t$ :

$$\bar{\vec{G}} \cdot \vec{U}(n\Delta t) + \bar{\vec{C}} \cdot \frac{d}{dt} \vec{U}(n\Delta t) = \vec{S}(n\Delta t) \quad (8.21)$$

The right-hand side of this system  $\vec{S}(n\Delta t)$  is known: it is a combination of analytically defined forcing voltages and/or currents and thin-wire characteristics at time  $n\Delta t$ . Consequently the system can be solved using any implicit backward-difference scheme. Herein we consider only the simple backward-Euler time-integration:

$$\vec{U}^n = \left( \bar{\vec{G}} + \frac{\bar{\vec{C}}}{\Delta t} \right)^{-1} \left( \frac{\bar{\vec{C}}}{\Delta t} \vec{U}^{n-1} + \vec{S}^n \right) + O(\Delta t) \quad (8.22)$$

By storing the circuit variables for one additional time-step ( $n-1$ ) and by precomputing and inverting the leading matrix, the circuit solution (8.22) and the corresponding incoming wire-fluxes *e.g.* (8.20) are efficiently computed.

---

### 8.5 *Chapter Summary*

In this chapter a subcell circuit model for terminating/driving and or interfacing multiple thin-wire segments has been introduced. The model uses modified nodal analysis augmented by a characteristic thin-wire equation to provide a consistent system of equations coupling the circuits and the wires. These circuits can be used to implement any of the flux-reconstruction boundary conditions presented in Section 7.3.

# Subcell Modeling Validation

---

*Two scientific activities are equally valid if they achieve results that are true.* – Alvin M. Weinberg

In this chapter the thin-wire and circuit subcell model implementations presented in Chapters 7 and 8 are validated for some select physical problems.

## 9.1 Example 1: A Thin-Wire Dipole

We consider the example of a 41 [m] dipole antenna radiating in free space. This example was simulated in [18] and serves as a good basis for comparison. The wire is a 10 [mm] radius perfect electric conductor and is aligned in the  $z$  direction. The wire is embedded at the centre of a spherical volumetric domain of radius 100 [m]. The

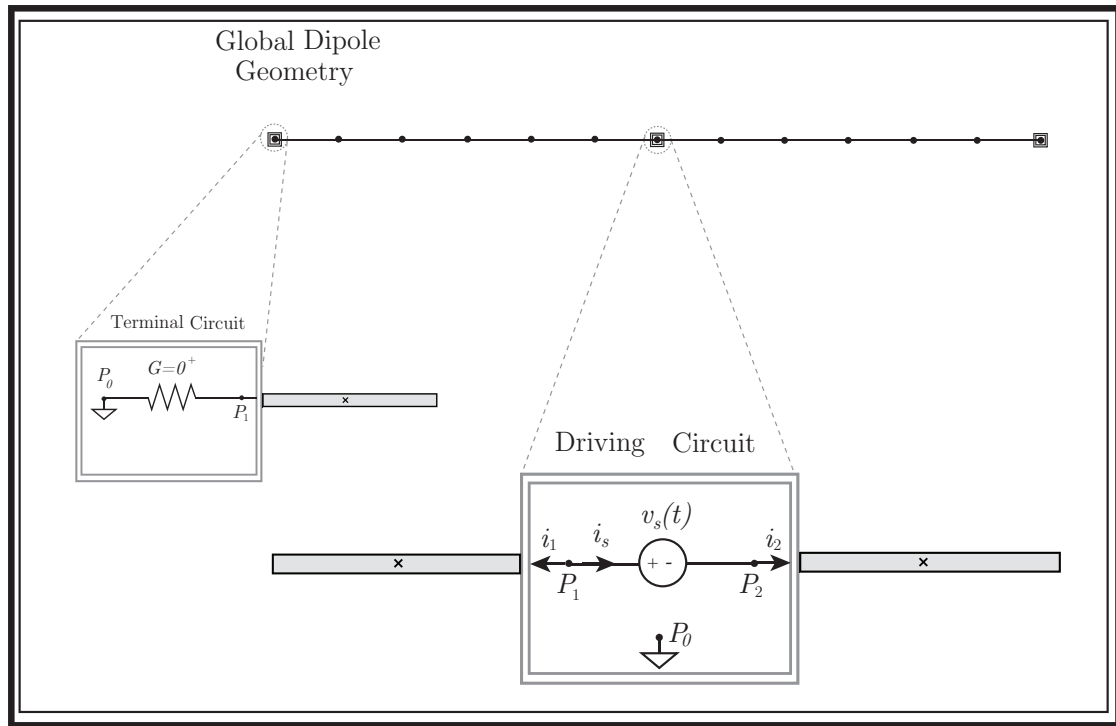


Fig. 9.1: Thin-wire dipole geometry. The dipole contains 42 segments (not all shown) and is driven by an independent voltage source modeled as a lumped-circuit. The wire terminals are also modeled by circuits having a near-zero conductance.

wire is discretized to have 42 segments of equal length. The volumetric discretization contains 300,000 tetrahedral volumes and supports a maximum frequency of around 30 [MHz]. The coupling distance  $\rho_0$  is automatically computed to be 2.78 [m] and the wire/field coupling integrals (7.54) and (7.56) are evaluated using third-order numerical quadrature. As shown in Figure 9.1, the dipole is driven by a circuit containing a single independent voltage source. The voltage time-variation was selected as the derivative of a Gaussian pulse with mean  $12 \times 10^{-8}$  [s] and standard deviation  $4.0 \times 10^{-8}$  [s] in order to correspond with the source used in [18]. This source contains appreciable energy up to 30 [MHz]. In addition to the voltage source, the dipole is terminated at each end by a circuit that models an open circuit. This is accomplished by using an infinite series resistance that is numerically modeled as a very large value (on the order

of  $10^9$ ). While we could have used the open-circuit boundary reconstruction provided in Section 7.3.5, we use the circuit to prove its versatility.

To extract the input impedance of the dipole, the source voltage  $v_s(t)$  and current  $i_1(t) = -i_s(t)$  (also shown in Figure 9.1) are stored over the duration of the simulation for post-processing. The results of a MUSCL, predictor-corrector simulation are shown in Figures 9.2 and 9.3. In Figure 9.2 (a) we show the time-domain voltage, while Figure 9.2 (b) shows the input current into the dipole. The simulated input impedance is extracted by taking the ratio of the FFT of the source voltage and input current. In part (c) of the figure, the input resistance is shown, while part (d) depicts the input conductance. These results compare very well with those of [18]. Specifically, the first half-wavelength resonance computed by the software package NEC-3 is  $72.2 \text{ } [\Omega]$  at  $3.53 \text{ } [\text{MHz}]$  [18]. From our results we obtain  $72.5 \text{ } [\Omega]$  at the same frequency.

## 9.2 Example 2: Straight and Bent Monopole Coupling

We now present the results of the thin-wire model applied to the coupling between two monopoles as published in our paper [57] in which the model is validated by measurements. We consider two similar geometries as shown in Figure 9.4. In the first geometry, (a), we are interested in simulating two monopole antennas above a PEC ground plane. The antennas are  $5 \text{ } [\text{cm}]$  long and separated by  $30 \text{ } [\text{cm}]$ . The ground plane is  $60 \text{ } [\text{cm}]$  by  $30 \text{ } [\text{cm}]$ . The expected quarter wavelength operating frequency is  $1.5 \text{ } [\text{GHz}]$ . For the second geometry (b), the same ground-plane is used and two L-shaped antennas made up of two  $5 \text{ } [\text{cm}]$  sections of wire were placed at the monopole locations. In both examples, the antennas have a radius of  $0.25527 \text{ } [\text{mm}]$ . Additional details are available in [57].



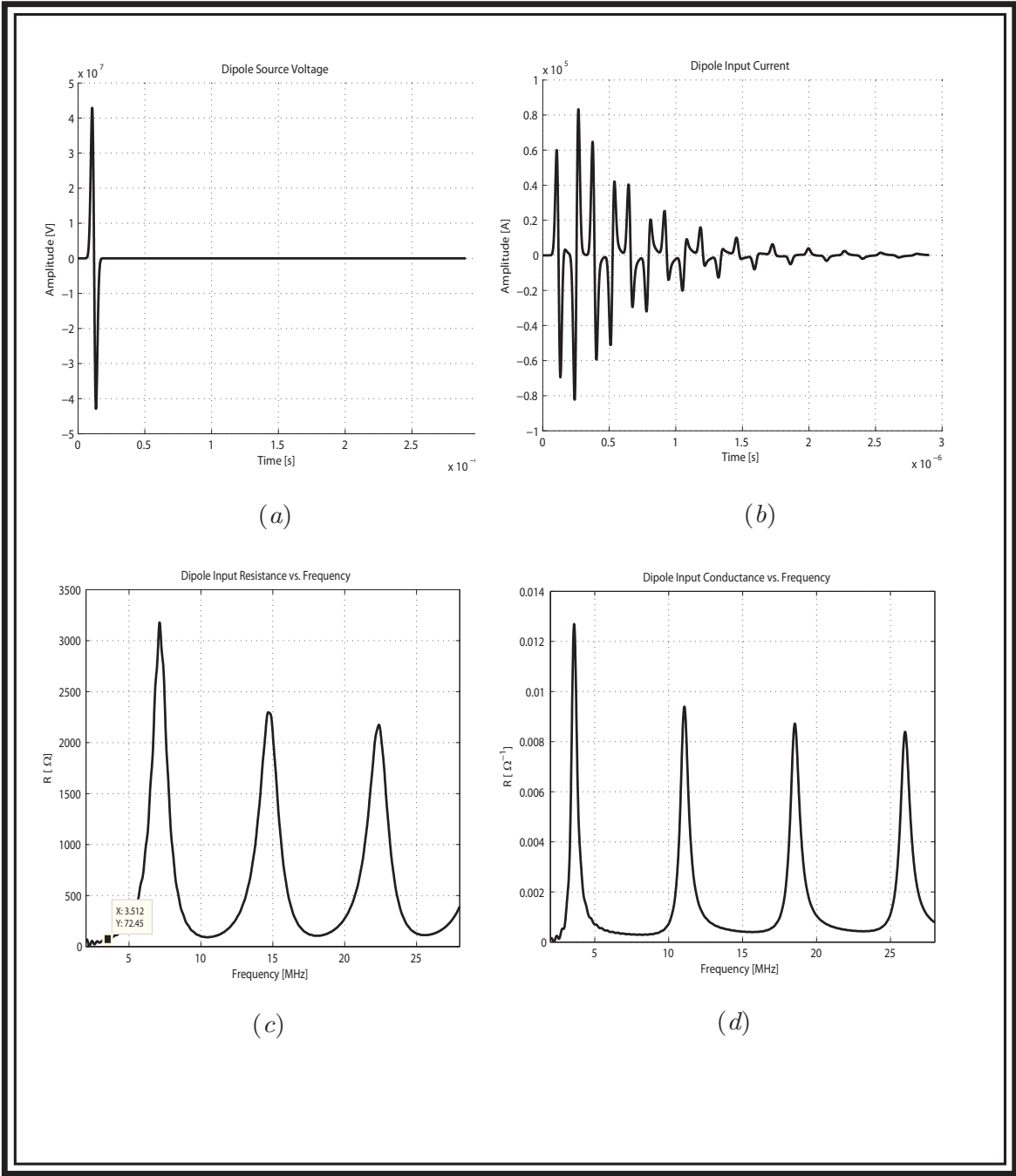


Fig. 9.2: Thin-wire dipole results. (a) Source Voltage, (b) Input Current, (c) Input Resistance and (d) Input Conductance.

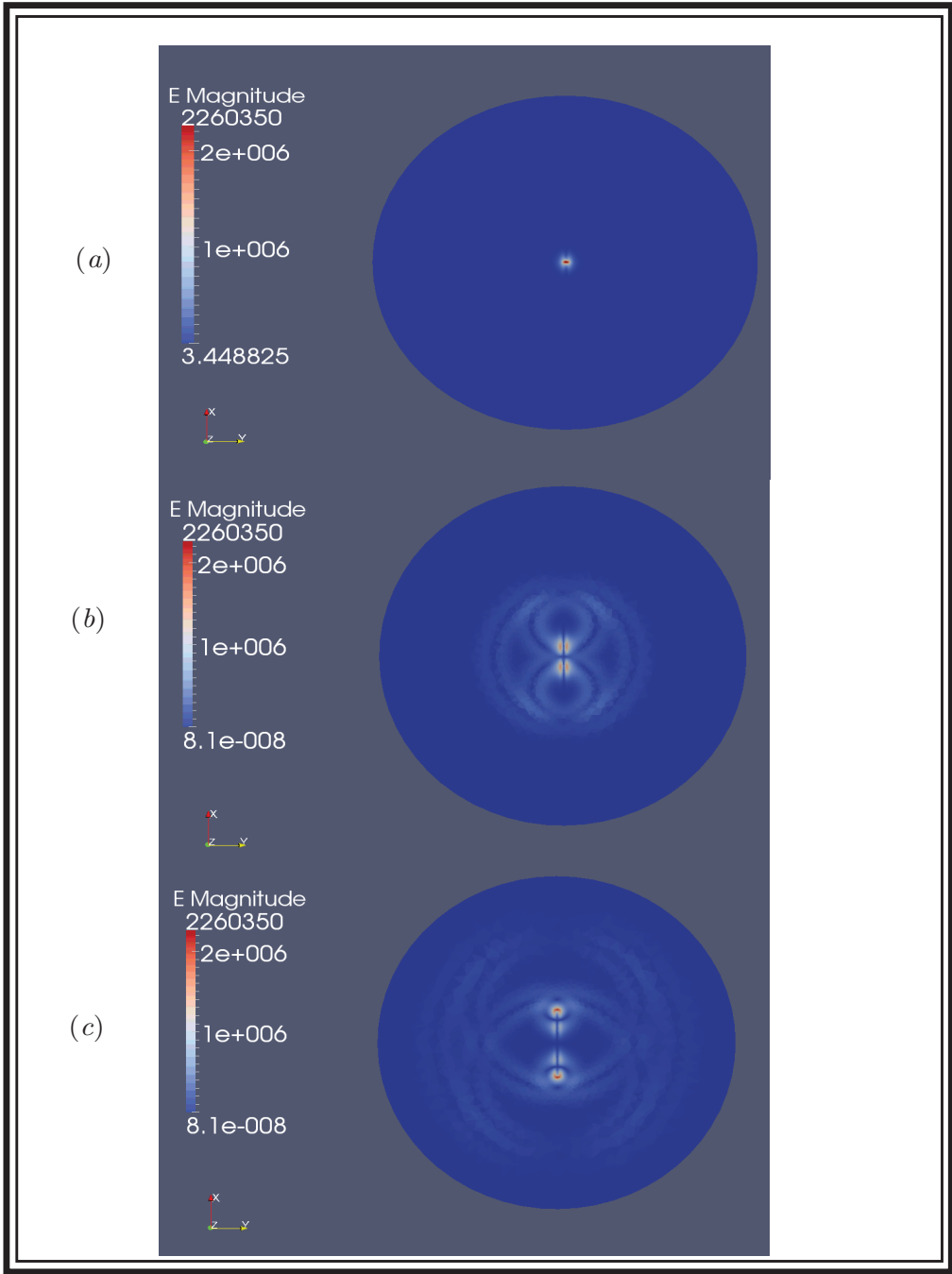


Fig. 9.3: (Colour) Visualization of simulated dipole fields. The magnitude of the electric field is shown at (a) 1200, (b) 2000 and (c) 2800 time-steps for a cross-section of the computational domain.

The volumetric discretization of the meshes resulted in roughly 159,000 tetrahedral volumes and 100 thin-wire segments for the two monopoles (200 segments for the L-shaped antennas). The maximum frequency supported by both meshes is approximately 2 [GHz]. In both simulations, one of the antennas was driven at the ground-plane junction by a voltage source. It is important to point out that this type of source is not differentially driven, and so the voltage referenced to  $\rho_0$  could have an effect. For both geometries, the antennas were excited using the derivative of a Gaussian pulse with mean  $6.5 \times 10^{-9}$  [s] and standard deviation  $10^{-10}$  [s] providing appreciable energy between 200 [MHz] and 5 [GHz]. From the simulation results, the network scattering parameters were computed using measured currents and voltages at the antenna feed-points. In addition, the geometries were fabricated by the lead authors of [57] and measurements were taken using a network analyzer. The simulation and measurement results for the two geometries are compared in Figure 9.5 (monopoles) and Figure 9.6 (L-shaped). A NEC simulation is also shown for comparison [58].

Despite the fact the wires were not differentially driven, the results are quite good. One possible explanation is provided by image theory: the ground plane is sufficiently large that an equivalent problem would be to remove the ground-plane and to differentially drive a system where the antennas are imaged across the ground-plane [35].

### 9.3 Example 3: A Thin-Wire Loop Antenna

We now demonstrate the FVTD thin-wire capabilities for modeling a thin-wire loop. This problem is particularly interesting as the thin-wire model permits the loop to be modeled using piece-wise continuous line-segments as opposed to other FDTD models that require stair-stepping. In order to compare with the results in [18], we choose a

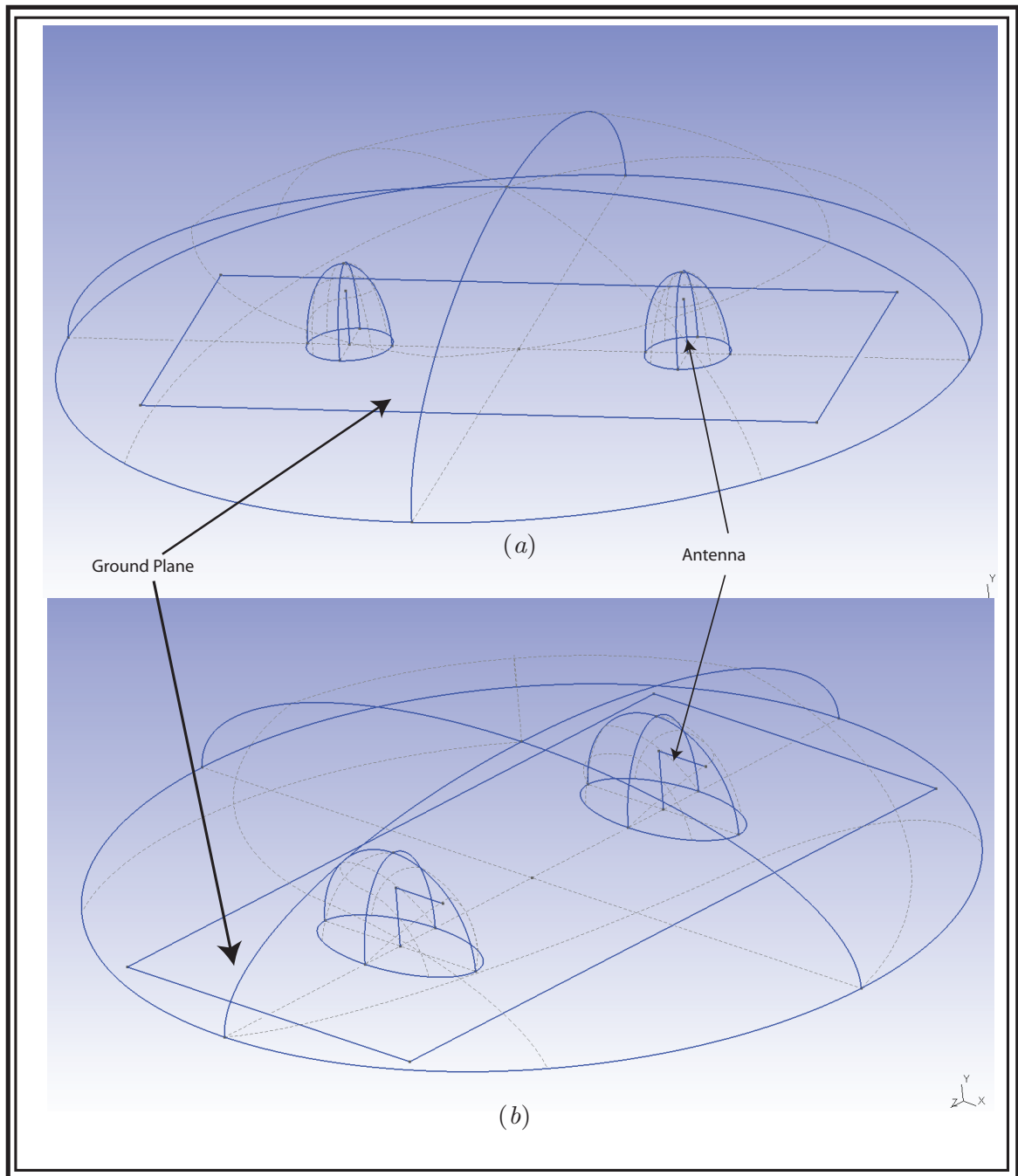


Fig. 9.4: (Colour) Geometry of simulated straight and L-shaped antennas. (a) Straight monopoles above a rectangular ground-plane. (b) L-shaped antennas above a rectangular ground plane. In both geometries, the regions surrounding the antennas are free-space and are introduced to ensure that the discretization in proximity to the antennas is the same size as the wire discretization.

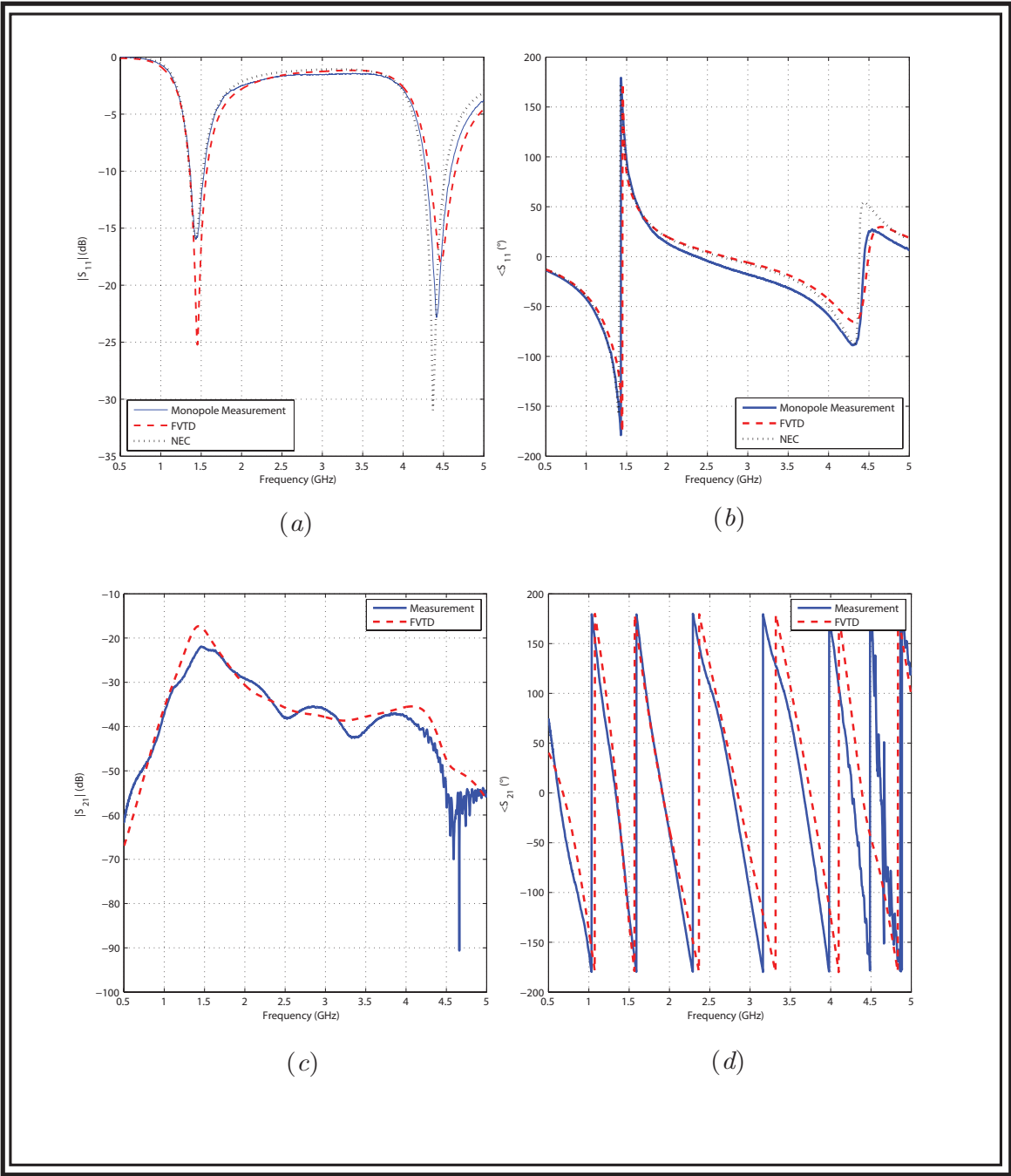


Fig. 9.5: (Colour) Simulation and measurement results for the coupling between two straight monopoles over a ground plane. (a) Magnitude of  $S_{11}$ , (b) Phase of  $S_{11}$ , (c) Magnitude of  $S_{21}$ , (d) Phase of  $S_{21}$

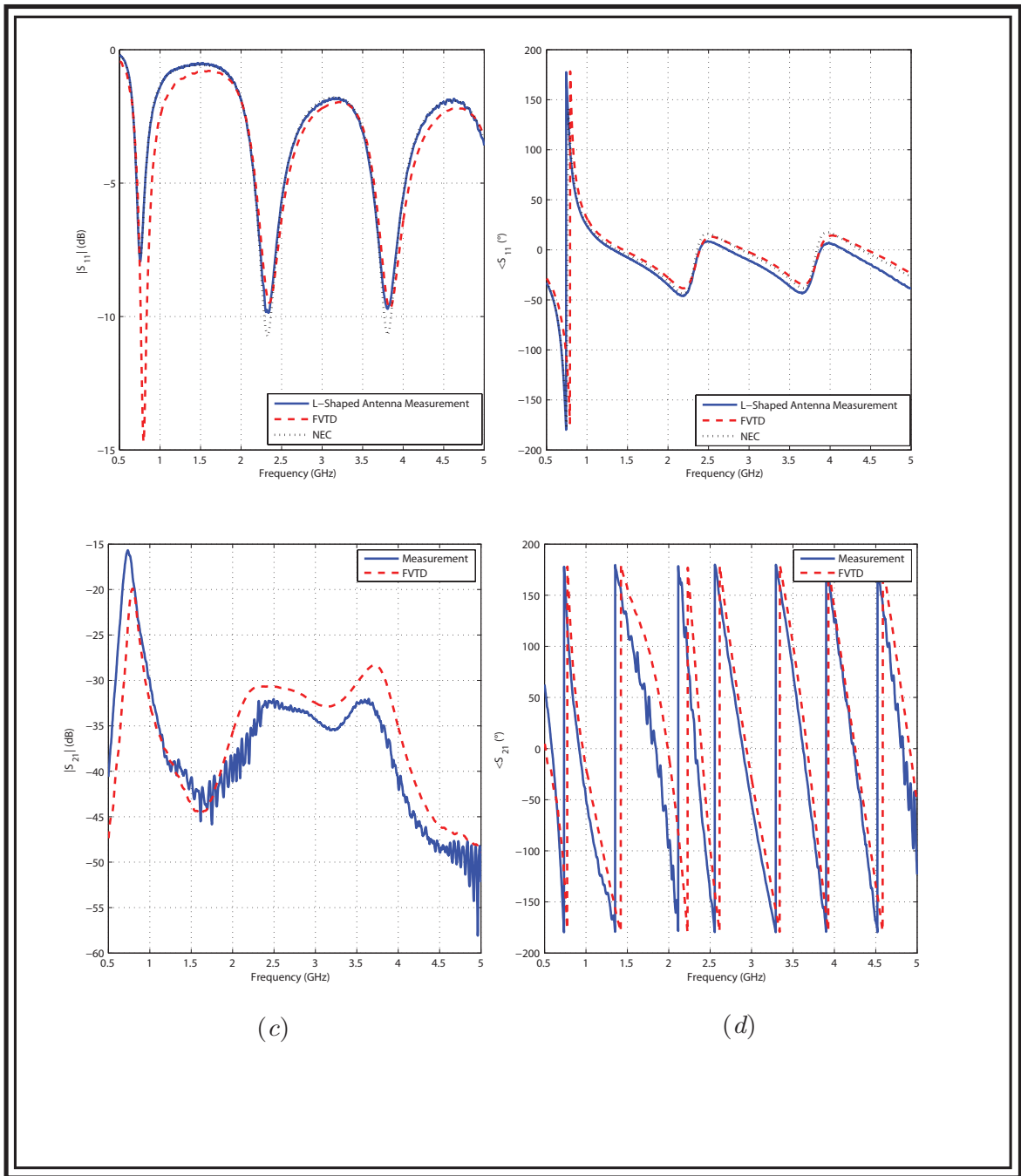


Fig. 9.6: (Colour) Simulation and measurement results for the coupling between two L-shaped antennas over a ground plane. (a) Magnitude of  $S_{11}$ , (b) Phase of  $S_{11}$ , (c) Magnitude of  $S_{21}$ , (d) Phase of  $S_{21}$

loop having a diameter equal to 1 [m], while the wire itself has a 1.05 [mm] radius. The loop is embedded in a spherical computational volume having a radius of 4 [m] and truncated using the local Silver-Müller boundary conditions. The discretized geometry contains 670,000 tetrahedral volumes and 48 equal-length wire-segments. The maximum frequency supported by the mesh is around 400 [MHz]. The loop is driven using the voltage source shown in Figure 8.3, as a single independent source placed between the terminals of two wire-segments. The derivative of a Gaussian pulse with mean  $10^{-8}$  [s] and standard deviation  $1.25 \times 10^{-9}$  [s] is used to excite the loop over a frequency range from 10 to 400 [MHz].

This geometry is particularly difficult to simulate and some remarks should be made to this effect.

- First, as shown in Figure 8.3, each pair of wire-segments is bent. It is absolutely necessary to bend the wire/field coupling cylinders as shown in Figure 7.3. Without this modification, undamped low-frequency oscillations appear in the solution, also observed in the FDTD and FETD implementations [18, 19]. These oscillations can be partially suppressed by selecting excitation voltages without low-frequency components (*i.e.* as the derivative of a Gaussian pulse), but is not always the best approach if we are concerned with low-frequency behaviour.
- Second, there is a large discrepancy between the minimum three-dimensional time-step of  $2.752 \times 10^{-11}$  [s] and the minimum one-dimensional time-step of  $2.18 \times 10^{-10}$  [s]. Therefore, we run the simulation at the volumetric time-step. Unfortunately, this causes an additional problem: the first-order Van Leer spatial reconstruction scheme (7.50) incurs too much dissipation at the volumetric time-step. Therefore, a high-order reconstruction scheme must be used. The results presented in this section are the combination of a fourth-order Runge-Kutta

time-integration and a third-order spatial reconstruction on the wire. In the volume, the MUSCL reconstruction was used. The high-order reconstruction on the wire is based on a simple polynomial reconstruction [10, 50]. As the source and fields are both smooth functions of time, a non-oscillatory reconstruction on the wire was not required. We should also mention that high-order reconstruction may not be required in all cases. For example, if radiative losses are sufficiently high, they may dominate the dissipative loss on the thin-wires due to lower-order reconstruction schemes.

- Third, and lastly, a sufficiently high quadrature rule must be used to ensure that the elements only partially influenced by the wire have a volumetric current density associated with them (recall Figure 7.3). For the loop we have used a 4th order integration rule.

The results of the FVTD simulation of the thin-wire loop are presented in Figures 9.7 and 9.8. In the first figure, the input voltage (a) and current (b) are shown. The input impedance is again computed as the ratio of the FFTs of these quantities and the input resistance (c) and conductance (d) are plotted up to 400 [MHz]. Again these results are in excellent agreement with those published in [18]. Specifically, the first resonance occurs at 99.58 [MHz] and corresponds to a resistance of 136 [ $\Omega$ ], within 1% of the 99.4 [MHz] resonance and 137 [ $\Omega$ ] resistance obtain in [18].

#### 9.4 Example 4: A Capacitively Tuned RF Loop for MRI

As a final example, we will further demonstrate the flexibility of the subcell circuit-model by tuning an RF loop antenna to a 63 [MHz] operating frequency. This problem is typical in the design of RF loops in Magnetic Resonance Imaging (MRI) signals for



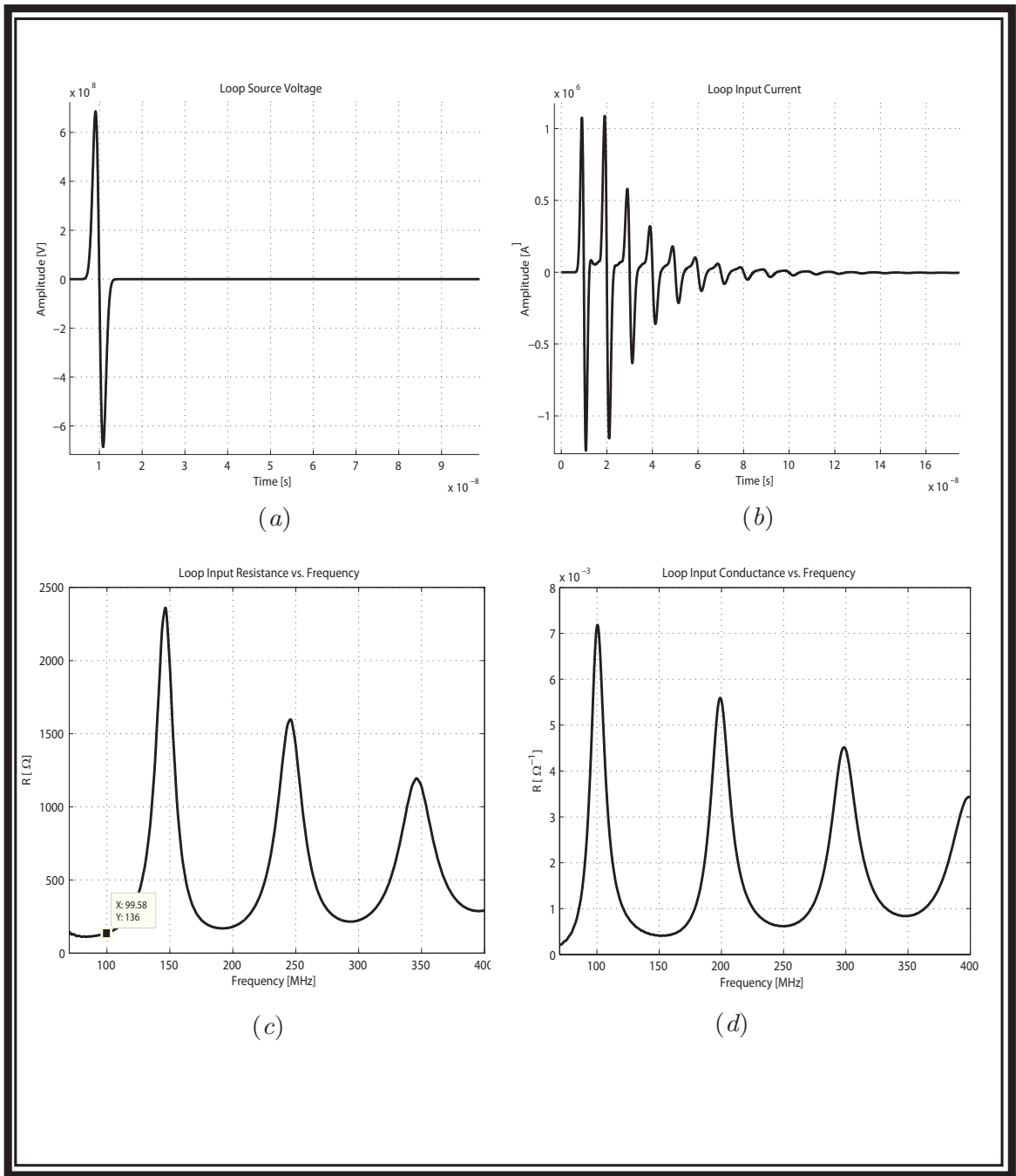


Fig. 9.7: Thin-wire loop simulation results (a) Source voltage (b) Input current, (c) Input resistance and (d) Input conductance.

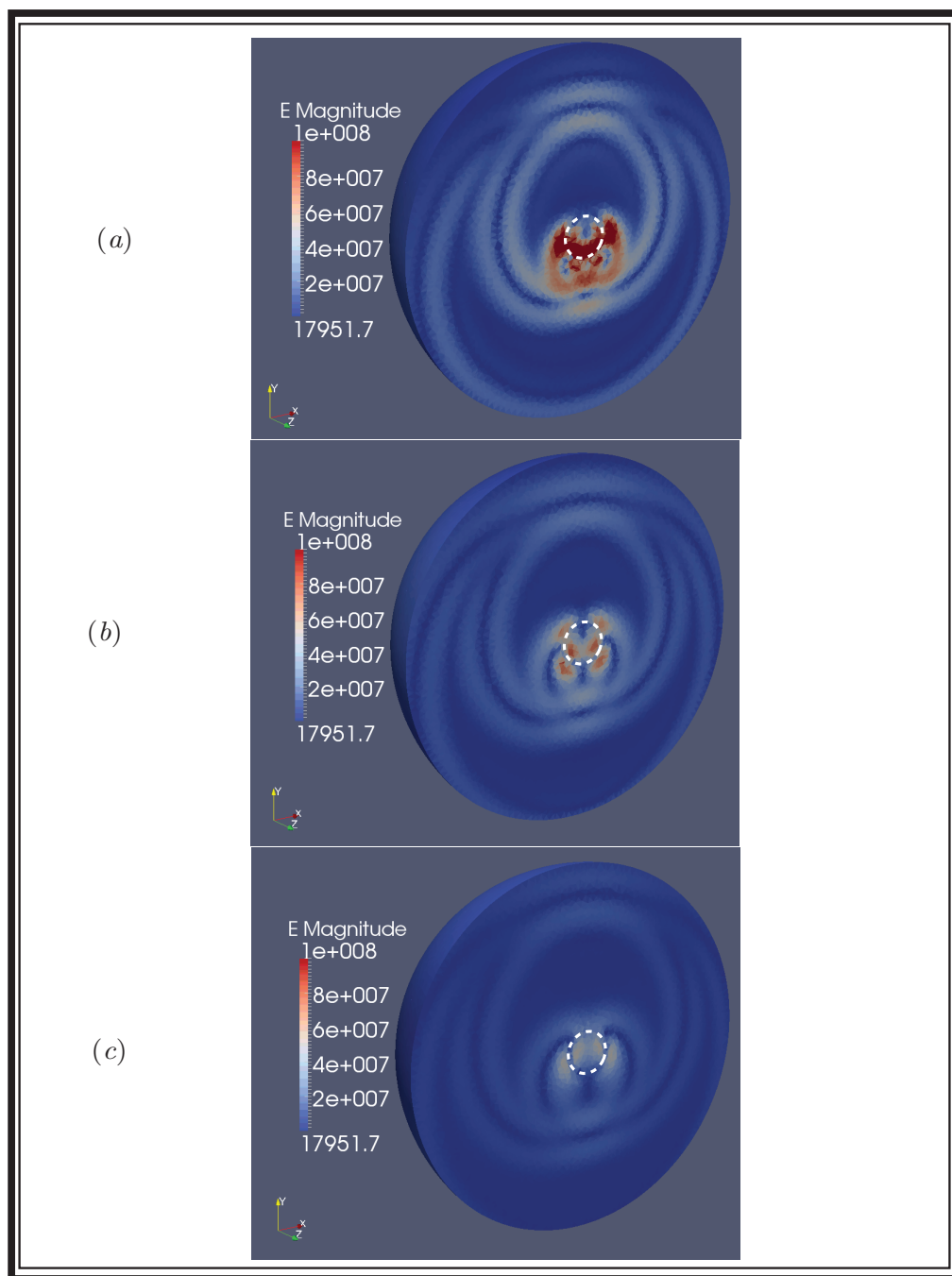


Fig. 9.8: (Colour) Thin-wire loop field visualization. The loop is represented by the dashed white curve. (a) 800 time-steps (b) 1200 time-steps and (c) 1600 time-steps.

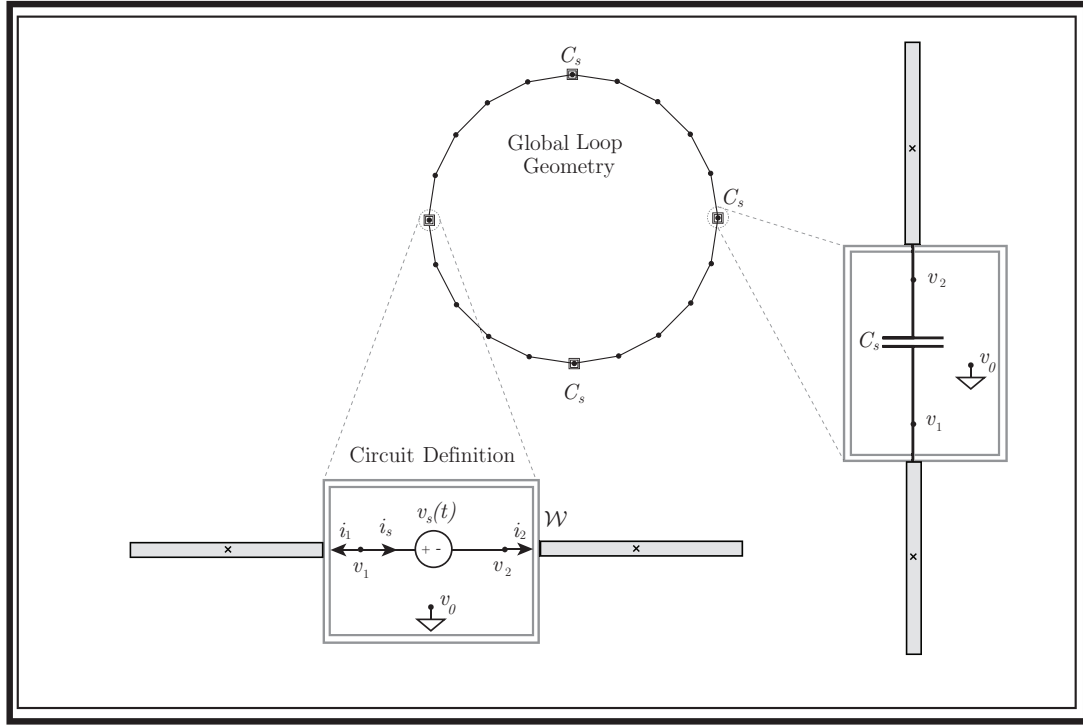


Fig. 9.9: Geometry of a capacitively tuned MRI thin-wire loop.

signal acquisition. The geometry is that of an 18.2 [cm] metallic loop (modeled as a PEC) having a diameter of 2 [mm]. The circumference of the loop is approximately 0.5719 [m] and corresponds to a half-wavelength at 262.3 [MHz]. The loop is embedded in an 80 [cm] spherical volume truncated with Silver-Müller boundary conditions and is excited by the circuit shown in Figure 9.9. The time-domain voltage source varied as the derivative of a Gaussian pulse with mean  $2 \times 10^{-9}$  [s] and standard deviation  $6 \times 10^{-10}$  [s], resulting in appreciable energy over a bandwidth from 200 to 700 [MHz]. The half-wavelength resonance of this loop at 262.3 [MHz] is not suited for, say, 1.5 Tesla MRI systems that operate at approximately 63 [MHz] [53]. It is, however, much more compact than the 37.9 [cm] radius loop having a natural half-wavelength resonance at 63 [MHz]. Therefore, we must bring the first resonance of the 18.2 [cm] radius loop down to 63 [MHz]. The low-frequency inductance of this loop can be computed analytically as  $L_{loop} = 0.525 \times 10^{-6}$  [H] [59]. Therefore, we can tune the loop by

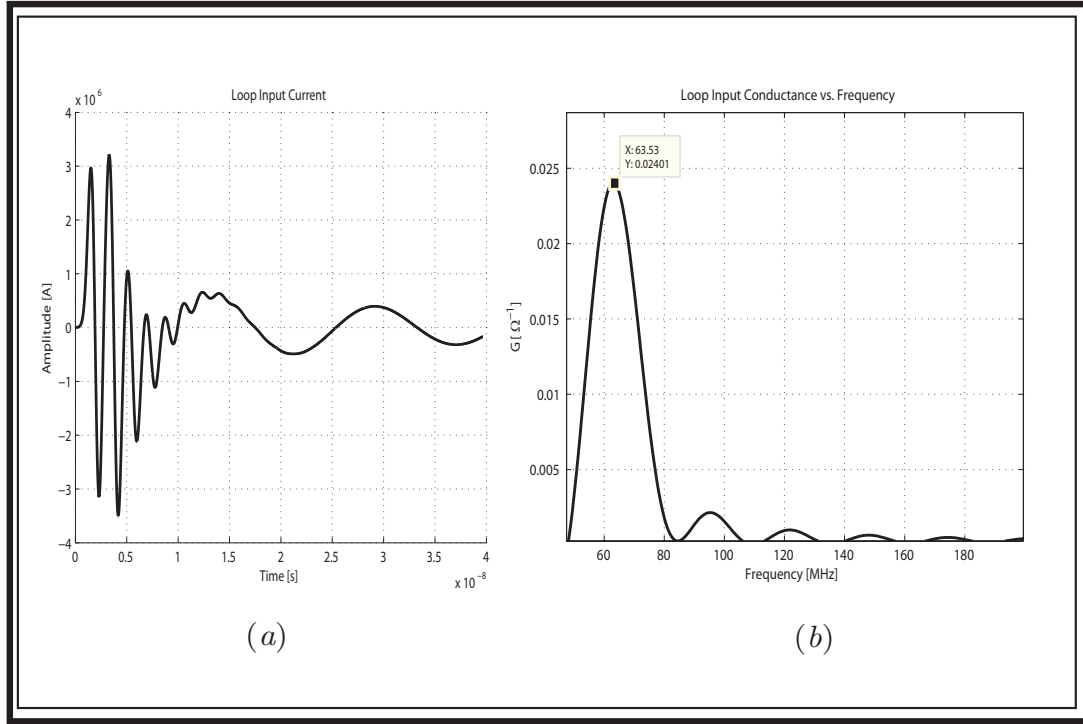


Fig. 9.10: Simulation results for an MRI-tuned RF loop. (a) Input current showing the introduction of the 63 [MHz] resonance and (b) Input conductance displaying a peak at 63.53 [MHz].

selecting a capacitance  $C_{loop}$  such that  $(L_{loop}C_{loop})^{-1/2} = 2\pi \cdot 63$  [MHz], or  $C_{loop} = 12.156$  [pF]. This capacitor could be added in parallel across the voltage source  $v_s(t)$  shown in Figure 9.9, but this problem is not that interesting because its input impedance could be computed by adding the input impedance of the original loop in parallel with the analytic input impedance for the capacitor. Instead, we add the capacitor in series by introducing three capacitors  $C_s$  at the three locations shown in the figure. The value of  $C_s$  is set equal to  $3C_{loop}$  so that the series sum of the capacitances equals  $C_{loop}$ . While such a geometry may not be precisely that used in an MRI system, it serves the purpose of demonstrating the power of the subcell circuit model. In this configuration, the capacitors become part a part of the field/wire system and a full simulation must be completed to obtain the input impedance. It is interesting to note that a similar setup

is simulated using an alternate thin-wire model (the so-called thin-rod approximation) for FDTD in [60]. In that work, the authors simulate a square thin-wire loop and also add series capacitors to tune the loop for MRI applications. The primary difference is that the thin-wire model used in [60] does not introduce charge and current as new variables (it falls into the “second” class of models discussed in Section 7.1). For such a model, it is not clear how to introduce a subcell circuit model that directly constrains the current and charge on the line. Instead, the capacitors are introduced as a modification of the permittivity in the *volumetric* discretization, localized to the capacitors. This makes the mesh dependent on the subcell model. On the other hand, our subcell circuit model places no such constraint on the mesh.

An FVTD simulation was performed on a discretized computational domain consisting of 450,000 volumetric cells and 48 wire segments. The parameter  $\rho_0$  was automatically computed as 1.5 [cm]. The simulation results are shown in Figure 9.10. The first resonance occurs at 63.53 [MHz] and corresponds exceptionally well to the desired MRI frequency of 63 MHz.

## 9.5 Chapter Summary

In this chapter, the thin-wire and lumped-circuit subcell models have been validated for a few interesting modeling problems. The results presented correspond quite well with theory and/or expectations. Specifically, the subcell models appear like excellent candidates for full field modeling of MRI systems. Future work will include modeling of such systems including MRI instrumentation cabling and the presence of physical inhomogeneities used to model MRI targets.

## FVTD Parallelization

---

*Sometimes one can improve the theories in the sense of discovering a quicker, more efficient way of doing a given calculation. – John Pople*

In this chapter an FVTD parallelization scheme for distributed parallel systems is presented. As we have seen in Chapter 2, the conservative form of Maxwell's Equations is hyperbolic, and consequently time-evolution of the solution is dependent only on local values. As will be shown, this makes parallelization straightforward provided that a suitable decomposition of the computational domain is available.

The parallelization is presented in three stages. First, in order to demonstrate the general parallelization strategy, the bare volumetric solution is parallelized without the local time-stepping (LTS) scheme presented in Section 4.3 or the global mesh-truncation scheme presented in Section 5.2. Second, a parallelization strategy is developed for when the volumetric algorithm is complicated by LTS and global mesh truncation.

Finally, a parallelization strategy that incorporates all aspects of the volumetric solver and the sub-cell models formulated in Chapters 7 and 8 is given.

In the aforementioned parallelization strategies, we are primarily interested in getting the algorithms to function in parallel. In each case, an appropriate weighting scheme is suggested that would theoretically permit equal load balancing in a parallel system. Only the general strategies have been implemented. Implementation of the precise load-balancing mechanisms discussed in this chapter is reserved for future work. Even in the absence of perfect load balancing, results presented at the end of this chapter demonstrate increased efficiency in a parallel environment.

### *10.1 A Brief History of Parallel FVTD*

Parallel FVTD, in the field of computational electromagnetics, dates back quite far. As early as 1995, Rowell *et. al.* had developed a characteristic-based FVTD implementation, parallelized on both structured and unstructured grids [61]. In 1997, Blake and Buter published a parallel FVTD implementation for body-fitted (conforming) grids [62]. Shang *et. al.* provide significant details of an MPI-based “ghost-layer” implementation on body-conforming grids [63]. Other implementations and details can be found in [64, 65, 66].

### *10.2 Distributed Parallel Environments - Message Passing Interface*

Herein, we consider parallelization schemes for distributed parallel environments, where by a distributed environment we mean that each processor has its own system memory (in contrast to shared memory environments where each processing core accesses the

same memory) [67, 68]. In this regard, we are interested in writing a Single-Program Distributed-Memory (SPDM) algorithm [68] in which each processing core receives the same program but contains only a portion of the volumetric domain in its local memory. Synchronization between processors is performed using the Message Passing Interface (MPI) [69], which although not technically a standard, is the most common way to communicate information for high-performance computing on distributed systems and supercomputers [70].

When discussing the parallelization strategies in this chapter we will approach the ideas generally. Therefore, the only important features of the selected parallelization environments are that 1) memory is distributed and not shared and 2) MPI permits communication over processors. Beyond these facts, we will not have cause to discuss implementation-specific details.

### 10.3 Volumetric Parallelization Strategy

We begin discussion of parallel FVTD by considering the volumetric domain  $V$  and its partition  $\{V_i\}$  into  $N_V$  volumetric elements. This partition is to be equally divided into  $P$  disjoint subsets  $V(p) = \{V_i\}_p$ ,  $p = 1, \dots, P$ , containing roughly  $N_V/P$  finite-volumes. If we momentarily assume that the time-evolution of a given cell  $V_i$  is independent of all other cells, then by assigning the partition  $V(p)$  to processor  $p$  in a  $P$ -processor distributed system, the computational time is reduced precisely to  $O(N_V/P)$ . The time-evolution of the cell  $V_i$  is not, however, independent of the global solution. Instead, for the upwind and flux-split FVTD scheme considered in this work, the flux integration around each  $V_i$  at each time-step is dependent on the reconstruction of the boundary states  $\vec{u}^*$  and  $\vec{u}^{**}$  as discussed in Chapter 3. The



spatial dependence of these states can be summarized as follows:

- For first-order upwinding, the state  $\vec{u}^*$  depends only on the solution in  $V_i$  while the states  $\vec{u}^{**}$  depend only on the solution in neighbouring elements.
- For the MUSCL scheme, the state  $\vec{u}^*$  depends on the solution in  $V_i$  and its gradient, the latter being a function of the solution in neighbouring elements. The states  $\vec{u}^{**}$  depend on the solution in the neighbouring elements and the neighbours of neighbouring elements used to produce the neighbouring gradients.

We recall from Section 3.7.4 that the elements required to update a given cell are referred to as the *stencil* for that cell. From the above conditions, it is clear that for first-order upwinding, the stencil for  $V_i$  includes  $V_i$  itself and its neighbours, while for MUSCL the stencil extends to the neighbours of neighbours of  $V_i$  as shown in Figure 3.4. This suggests that  $V(p)$  can be time-evolved (almost) independently of other processors by adding the additional constraint that the partitions  $V(p)$  are spatially continuous while covering  $V$ . Henceforth, each subdomain  $V(p)$  is assumed spatially continuous as shown for a simple 2D example in Figure 10.1; how we compute this partition is discussed in Section 10.4.2.

Given a partition  $V(p)$  and its boundary  $\partial V(p)$ , processor  $p$  can time-evolve the solution for every element  $V_i$  in  $V(p)$  (abbreviated  $V_i(p)$ ) when its stencil lies entirely in  $V(p)$ . For elements whose stencils cross beyond the local boundary  $\partial V(p)$  the neighbouring elements exist on neighbouring processors and are not available locally (in this sense the neighbours of a processor share a geometric boundary of the global partition). We denote the set  $V^*(p) \supset V(p)$  as the union of all stencil elements for  $V(p)$  – given the solution in  $V^*(p)$ ,  $V(p)$  can be time-evolved independently of the rest of  $V$ .

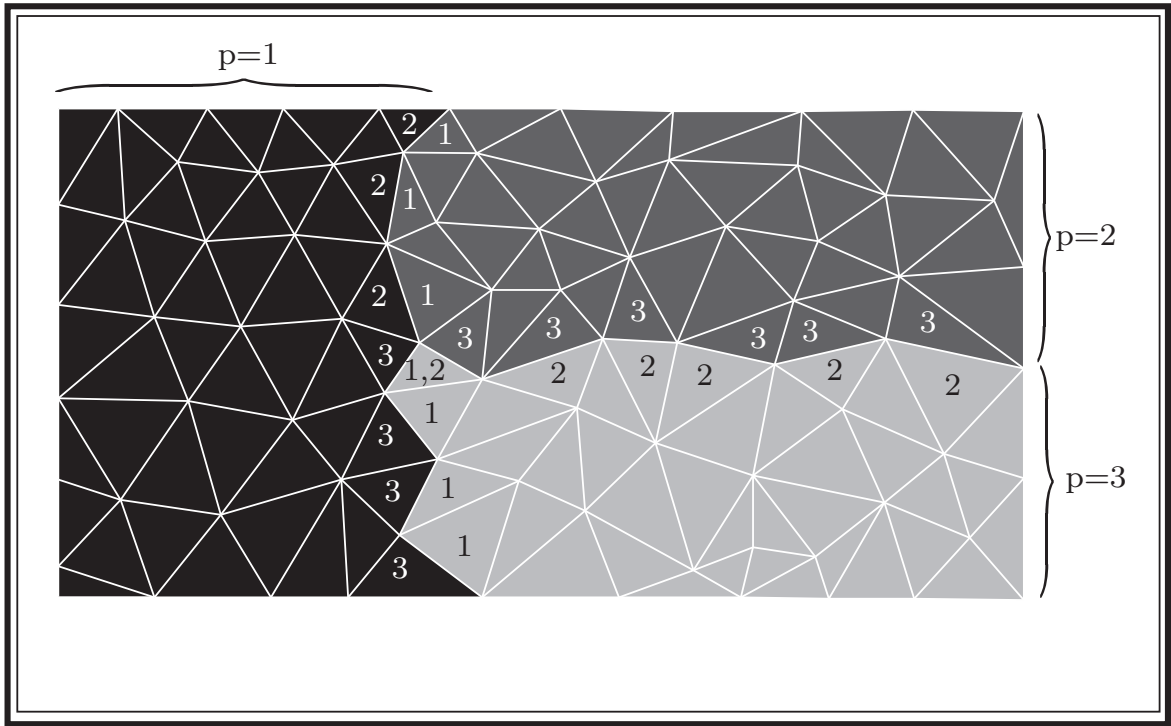


Fig. 10.1: A 2D depiction of the continuous parallel partitions for a three-processor distributed environment. For first-order stencils, the elements of a given domain are labeled with the number(s) of the processors where they are also present as halo elements.

Different techniques can be used to complete the set  $V^*(p)$  on processor  $p$ . Herein the concept of “halo”, or “ghost” cells are used [63]. The cells in the region  $V^*(p) \setminus V(p)$  do not belong to  $V(p)$  and therefore lie in the partitions of neighbouring processors. As the time-evolution of FVTD hinges on the geometric properties of the cells, the so-called collection of “halo” elements, denoted  $V^h(p)$  and consisting of  $V^*(p) \setminus V(p)$ , are obtained from the neighbouring processors and are *copied* on processor  $p$ . This “ghosting” or “haloing” of elements adds an additional (but generally small) global memory cost of duplicated geometric elements. The benefit is that the geometric information required to update all elements in  $V(p)$  is known. For the simple 2D example mesh presented in Figure 10.1, the elements on a given processor are numbered to denote the processor to which they will be copied as halos.

While the geometric information for each halo element is available directly to the processor, processor  $p$  is not responsible for time-evolving the elements inside  $V^h(p)$ . Therefore, some book-keeping is required to ensure that each processor knows which of its elements exist as halo elements on other processors. In what follows we will assume that such “halo-maps” exist.

#### *10.4 The Parallel Mesh Read and Domain Decomposition*

Having presented an appropriate algorithm for accelerating the bulk volumetric field-solver for distributed systems we are left with selecting an appropriate domain decomposition technique to determine the subdomains  $V(p)$ . There is a vast literature on this problem, so much so that it has its own dedicated website [71] where one can find an exhaustive BibTex bibliography file [72]. To select a method, let us consider the nature of the problems we are trying to solve: namely Maxwell’s Equations in a volume  $V$ . We first assume that the distributed parallel environment consists of  $P$  processors each with equal processing speed and that these processors are not running any other tasks. As we are not considering non-stationary media, the computational domain is fixed and if we have a partition  $V(p)$  that provides us with  $O(N_V/P)$  computational complexity for a given time-step, this complexity persists throughout the numerical simulation. Next, we assume that the volumetric problem is “sufficiently large” such that the computational time required to update the halo data at each (partial) time-step is negligible compared to local update of  $V(p)$ . Finally, as we are currently neglecting LTS, ERBC and subcell models, we can safely assume that the computational cost associated with updating each element  $V_i$  in the global  $V$  is constant. (While it is true that for the update of boundary elements, absorbing or perfect conductors, there is a reduction in the computational cost, this can be neglected.) Combining these

conditions leads to the following conclusion: *a subdivision of  $V$  into a covering of  $P$  continuous, non-overlapping subdomains  $V(p)$  each containing approximately  $N_V/P$  elements will provide  $O(N_V/P)$  computational complexity.*

Before discussing domain decomposition techniques that provide a partitioning that satisfies this condition, we consider the memory complexity of the algorithm. As stated in Section 3.8, the FVTD algorithm requires  $O(N_V)$  memory to store the geometric information and solution at a given time-step. If the entire mesh is stored on each processor the memory will scale as  $O(N_V)$  independent of the number of processors used. There are benefits to having the global mesh available to each processor when dynamic load balancing is considered. For example, recomputing the parallel subdomains to account for differences in the computational load and/or speed of each processor in the distributed environment does not require shifting elements from processor to processor. The cost of this luxury, however, is too great when problem sizes become large; irregardless of the savings in computational time, it may not even be possible to load large meshes into memory. Consequently we require that the memory required by the parallel algorithm also scales as  $O(N_V/P)$ . In order to accomplish this scaling, we require a parallel mesh-read.

#### 10.4.1 Parallel Mesh Read

We assume a mesh file that consists of a list of vertices followed by a list of elements defined in terms of vertex indices. It is assumed that the list of vertices is provided in order of increasing mesh vertex number. To ensure that the memory never grows beyond  $O(N_V/P)$  on a given processor we perform the following steps, depicted graphically in Figure 10.2:

- (a) Each processor reads an equal and sequential portion of the vertex list. These vertices are created in the local memory on each processor. This list of vertices does not change during the computation and is referred to as the *static vertex list*. As the list of vertices in the mesh file is assumed to be given in order of increasing vertex id, the range of static vertices do not overlap. On each processor the range of static mesh-vertex indices is computed and broadcast to all other processors. This provides each processor with the knowledge of which processor contains what vertices in its static vertex list.
- (a - continued) Next, each processor reads an equal portion of the element list. These elements are created in local memory and include the mesh ids of the element vertices. By assigning only a portion of the vertex and element lists to each processor upon reading the mesh the memory requirements on each processor are  $O(N_V/P)$ .
- (b) On each processor  $p$ , the positions of the vertices required to define the geometries of each element are not necessarily available locally as they are only available in the static vertex lists. We refer to the list of unique vertices required to define the local elements on processor  $p$  as the *dynamic vertices*. These dynamic vertices are obtained by making a request to the appropriate processor based on the static vertex ranges. The dynamic vertices are created locally and stored in memory completing the geometric information for the initial mesh read. At this point we have both a dynamic and static vertex list on each of the processors. While this involves additional memory, the memory complexity remains  $O(N_V/P)$ .

The benefit of the static vertex lists is that, as they do not change, they permit complete topological description of the mesh provided that they are supplemented

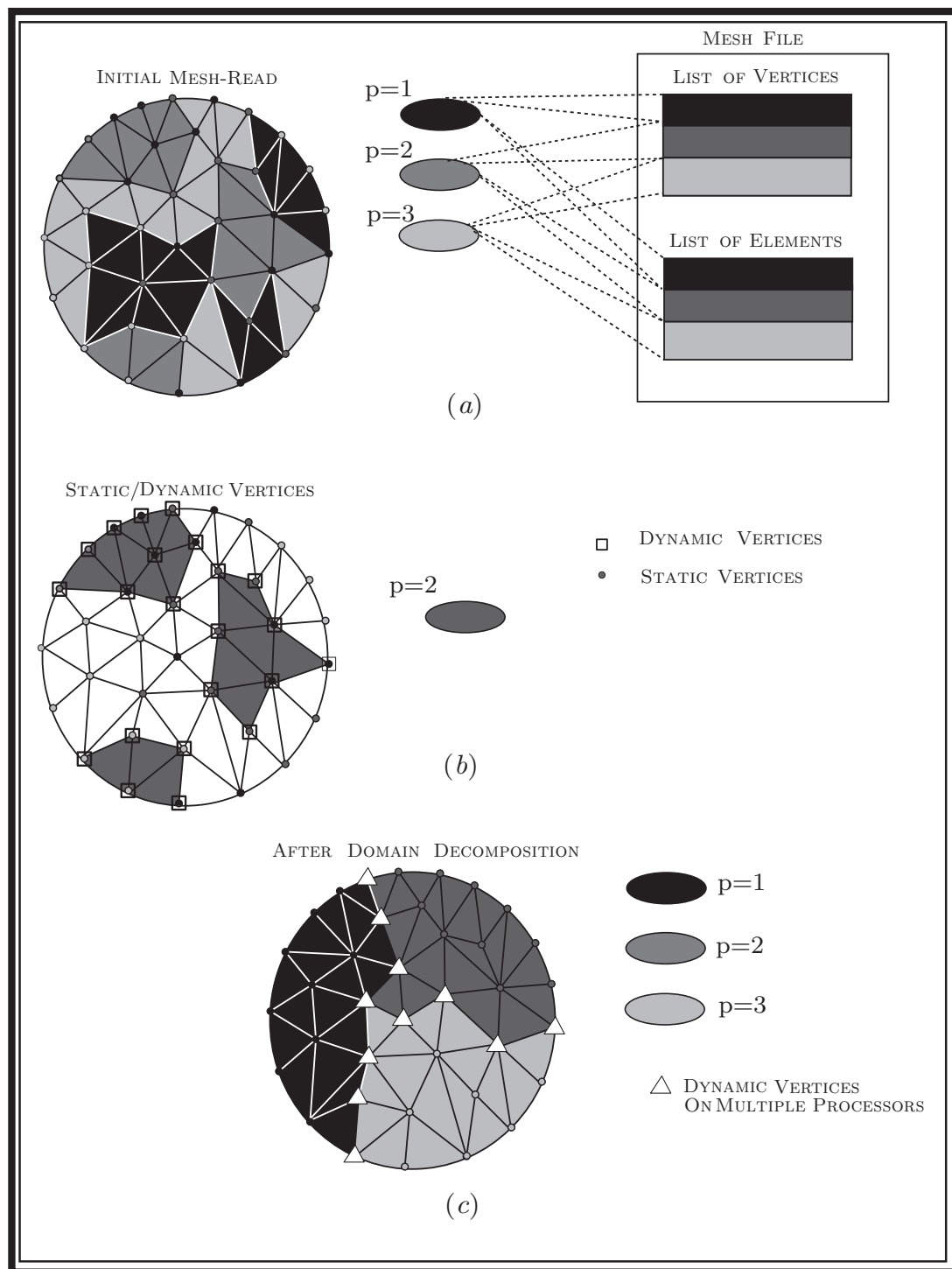
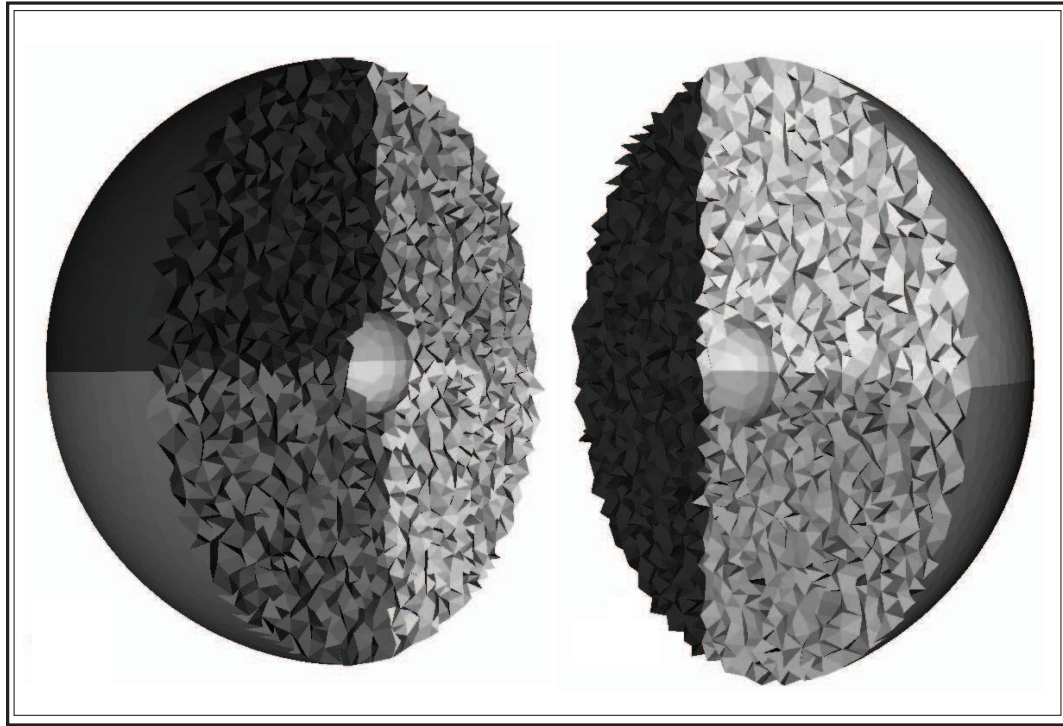


Fig. 10.2: Visual description of the parallel mesh read, illustrated in 2D. (a) Sequential sections of vertices and elements are read on each processor - the vertices make up the static vertex list. (b) By obtaining the dynamic vertices we can complete the geometric description of each element. (c) Using a domain decomposition technique we arrive at a continuous partition of the global domain.

with a global list of all elements connected to each vertex. Such a list is easily created in  $O(N_V/P)$  time by constructing a list of local elements connected to each dynamic vertex on each processor and communicating these lists to the corresponding static vertex. Amalgamating the list of all elements connected to a static vertex results in a global list of elements (and the processors they exist on) connected to that vertex. For well-structured meshes this results in a constant number of integers (element ids) associated with each vertex and leaves the computational complexity unchanged.

#### 10.4.2 Domain Decomposition

Having completed an initial parallel mesh read that requires  $O(N_V/P)$  memory storage, we can now apply a domain decomposition algorithm with the simple constraint that the decomposition produces  $P$  subdomains  $V(p)$  containing approximately  $N_V/P$  elements each. Three straight-forward and computationally efficient algorithms that accomplish this task are: Orthogonal Recursive Bisection (ORB) [73, 74, 75], Hypergraph Partitioning (HG) [76, 77], and Space-Filling Curves (SFC) [78, 79]. In an initial implementation of an FVTD solver, ORB was implemented, from scratch, directly in the FVTD code [80]. More recently, the *Zoltan* library has been adopted as it provides implementations of all three of the considered algorithms [81, 82, 83]. A complete discussion of these algorithms is beyond the scope of this thesis and the reader is referred to the references above. What will be mentioned here is that the ORB and SFC algorithms require the geometric location of each element which is available by completing the dynamic vertex list for the initial mesh read. On the other hand, the HG algorithm only requires the topological connectivity between elements. This connectivity is computed in constant time using the global static-vertex-to-element list computed during the initial mesh read.



*Fig. 10.3:* An ORB domain decomposition for 7 processors. The volume has been split, and shifted for perspective. The left side of the image contains 4/7ths of the total volume.

By running one of the Zoltan domain decomposition algorithms we obtain a partition index  $p$  associated with each element on each processor. The following steps localize the partitions  $V(p)$  on processor  $p$  as shown in Figure 10.2 (c):

- Using the computed partition indices, each element with index  $p$  is transferred to processor  $p$ . This communication phase results in a momentarily doubling of the number of elements on each processor (the elements being distributed in addition to the elements received from other processors corresponding to  $V(p)$ ). The memory complexity remains  $O(N_V/P)$ .
- Upon receiving the elements in  $V(p)$  the initial element list is destroyed and the dynamic vertex lists are reconstructed to correspond to  $V(p)$ .



- The static-vertex-to-element list is updated to reflect the processor locations of each element.
- The elements are connected to their neighbours completing the description of  $V(p)$ . If an element in  $V(p)$  is missing a neighbour on some facet then it is either a boundary element of the computational domain or the neighbour lies on a neighbouring processor.

This completes obtaining the list of local elements  $V(p)$  on each processor  $p$  and we are left, for example, with the local partitions shown in Figure 10.3 where a 7-processor partition of the global domain  $V$  is shown. We must now augment this local list to include the halo elements.

#### 10.4.3 Halo Construction

As discussed in Section 10.3,  $V(p)$  must be supplemented with  $V^h(p)$ . This is accomplished layer-by-layer as follows:

1. For each element in  $V(p)$  that is missing a neighbour we determine the processor that owns the neighbour from the available topological connectivity through the static vertex lists. (This processor will not exist if the element is a true boundary element on  $\partial V$ .)
2. Each missing neighbour is copied to processor  $V(p)$  and appended to the end of the element list. Book-keeping is maintained to ensure that each processor is aware of which processors its elements have been copied to.
3. The element connectivity on  $V(p)$  is updated to include the newly acquired elements.

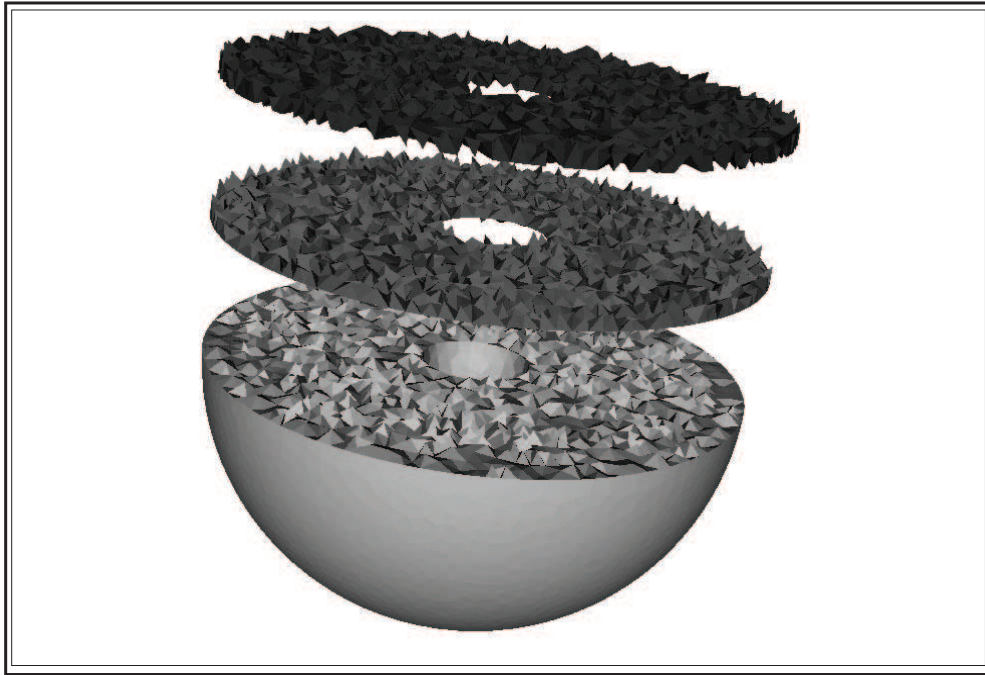


Fig. 10.4: A visualization of two halo layers required on a single processor for the MUSCL scheme. The layers have been offset for clarity.

4. Repeat the three steps above until the appropriate number of stencil layers are acquired.

An example of this procedure is shown in Figure 10.4. By building the halo elements up layer-by-layer, it is easy to modify the halos to support different spatial reconstruction schemes.

### 10.5 The Parallel Volumetric Solution

Having appropriately decomposed the volumetric domain  $V$  into  $P$  subdomains  $V(p)$  of equal size  $O(N_V/P)$  on  $P$  processors, the parallel FVTD volumetric solution of Maxwell's Equations is straightforward. We note that the effect of the domain decom-

position is to partition the global system of equations (4.1):

$$d_t \vec{\mathbf{u}}(t) = -\vec{\mathcal{F}}(\vec{\mathbf{u}}(t)) + \underline{\underline{\Sigma}} \cdot \vec{\mathbf{u}}(t) + \vec{\mathcal{S}}(t) \quad (10.1)$$

such that each processor contains a subset of the *rows* of the global matrix. Therefore, that portion of the matrix-vector product required at each time-step (or partial time-step if higher-order schemes are considered), can be performed locally provided the halo elements are updated to the current time  $n\Delta t$ . This update occurs once before each (partial) time-step. The overhead incurred is fractional compared to the  $O(N_V/P)$  update time required on each processor provided that the processor boundaries contain much fewer elements than the bulk volume.

### 10.6 Volumetric Parallelization and Local Time Stepping

When the local-time-stepping (LTS) scheme presented in Section 4.3 is desired in a parallel framework, the computational cost associated with each volumetric element is not equal. A straightforward partitioning of the problem into subdomains having  $N_V/P$  elements will yield acceleration, but not ideal acceleration. If  $O(N_V/P)$  memory scaling is the primary concern, say for very large meshes, then straightforward partitioning is warranted. If, however, execution time is the primary constraint, we offer the following obvious solution.

All of the domain partitioning algorithms available in Zoltan permit weights to be assigned to each element. Then, the partitioning scheme attempts to balance the total weight associated with a given subdomain. In the LTS scheme, the maximum level  $\ell_{max}$  operates at a time step  $2^{\ell_{max}-1}\Delta t$ . To avoid fractional weights we will define the

LTS parallel weight  $\Gamma_{V_i,LTS}(\ell)$  associated with each element at level  $\ell$ :

$$\boxed{\Gamma_{V_i,LTS}(\ell) \triangleq 2^{\ell_{max}-\ell_i}, \quad \text{where } V_i \text{ operates at level } \ell} \quad (10.2)$$

In a three-level LTS scheme, the weights associated with each level are then:

$$\Gamma_{V_i,LTS}(1) = 2^{3-1} = 4$$

$$\Gamma_{V_i,LTS}(2) = 2^{3-2} = 2$$

$$\Gamma_{V_i,LTS}(3) = 2^{3-3} = 1$$

The highest weight is associated with the lowest level, as these will require twice as many computations as the elements one level above.

A comment should be made regarding the LTS pre-processing algorithm required to determine the LTS domains. As presented in Section 4.3.1, the algorithm on page 57 requires checking all elements connected to a vertex of a given element  $V_i$ . If  $V_i \in V(p)$  on processor  $p$  is on the boundary of processor  $p$ , this information can be obtained through the global topology available via the static vertex lists.

### 10.7 Volumetric Parallelization and Global Mesh Truncation

While developing an appropriate parallel modification for local time-stepping was straight-forward, things are not so simple when the global mesh truncation scheme presented in Section 5.2 is included. We suppose that we have an integration surface  $\partial V_S$  that is made up of  $M_{\partial V_S}$  facets and that  $M_{\partial V}$  facets exist on the boundary  $\partial V$  of the global computational domain as shown in Figure 10.5. For each of these global boundary facets, the evaluation of the integrals (5.2) and (5.3) over  $\partial V_S$  scales

as  $O(M_{\partial V_S})$ . For completeness, we should include the fact that the evaluation of the integrals over a single facet is some constant  $q_{ERBC}$  times more expensive than a single volumetric update. Therefore, the computational weight required to evaluate the surface integrals for each boundary element is:

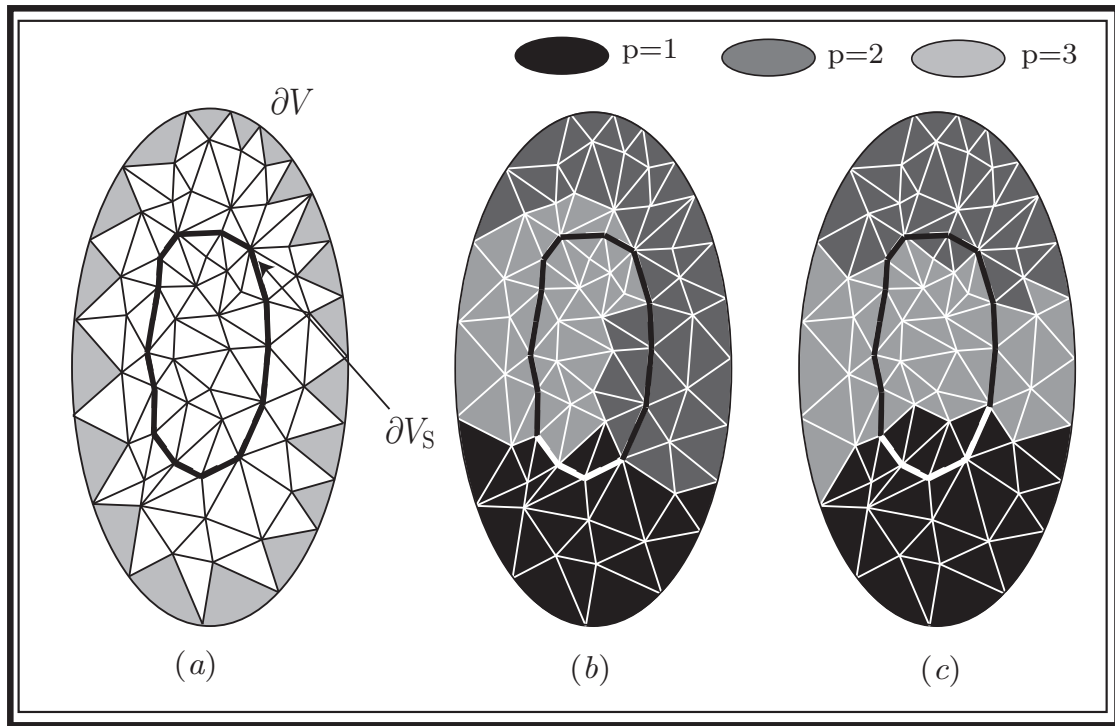
$$\boxed{\Gamma_{V_i,ERBC} = q_{ERBC} \cdot M_{\partial V_S}, \quad \text{if } V_i \in \partial V} \quad (10.3)$$

and is zero otherwise. The total weight associated with each element, where we permit an LTS scheme, is

$$\Gamma_{V_i} = (1 + \delta(\partial V) \cdot q_{ERBC} \cdot M_{\partial V_S}) \cdot \Gamma_{V_i,LTS}(\ell) \quad (10.4)$$

where  $\delta(\partial V)$  is unity when  $V_i \in \partial V$  and is zero otherwise. Imposing this weight on each element and performing a domain decomposition will provide a set of processor subdomains that require approximately equal computational time. The difference between a straightforward volumetric partition and the application of this weighting scheme is shown in Figure 10.5. In part (a) of the figure, we see the global computational domain and the Huygens' surface  $\partial V_S$ . The boundary elements are highlighted in grey. A simple volumetric partition (LTS excluded) could produce the volumetrically balanced partitions shown in part (b). Clearly, however, the number of boundary cells associated with each subdomain is not the same. In part (c), we see the effect of adding the weighting in equation , where now, both the volumetric and boundary computations are balanced.

While the above parallelization scheme does equally divide the computational weight on each processor, we have not yet discussed the evaluation of the surface integrals (5.2) and (5.3). To ensure that each processor is capable of evaluating these integrals,



*Fig. 10.5:* A 2D visualization of the parallelization of the global mesh-truncation scheme. (a) The global domain and the integration surface. The grey boundary elements require evaluation of the surface integrals. (b) A partition of the global domain that does not balance the computationally intensive boundary element. (c) A better partition that equally balances the number of boundary elements on each processor.

we propose a duplication of the surface  $\partial V_S$  and the back-stored solution values, on each processor as shown in Figure 10.6.

In the figure we see that each processor has its associated volumetric elements in addition to the surface  $\partial V_S$ . The memory duplication on each processor for the surface  $\partial V_S$  is permissible provided that the number of required back-stored time-samples is not exceedingly large. The back-stored surface values on  $\partial V_S$  must be updated across all processors at each time-step, resulting in a small global communication of  $O(M_{\partial V_S})$  values across all processors at each time step.

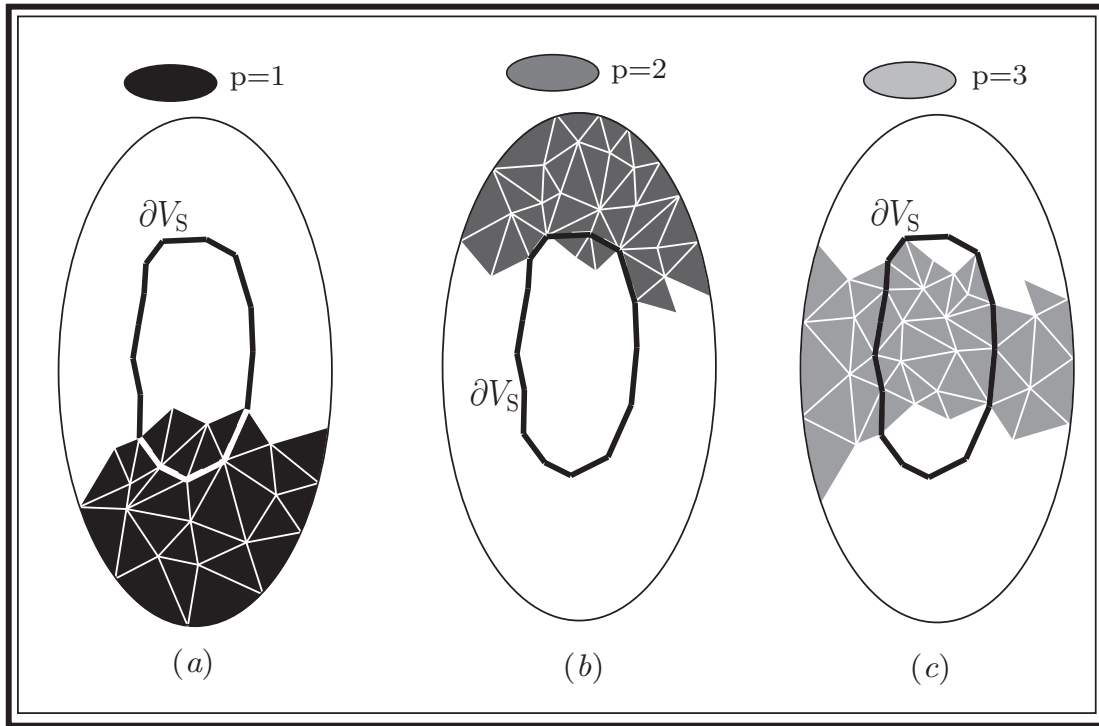


Fig. 10.6: Parallel partitions including global mesh-truncation in a three-processor environment. (a) Processor 1, (b) Processor 2, (c) Processor 3. In each case the Huygens' surface is duplicated on the processor.

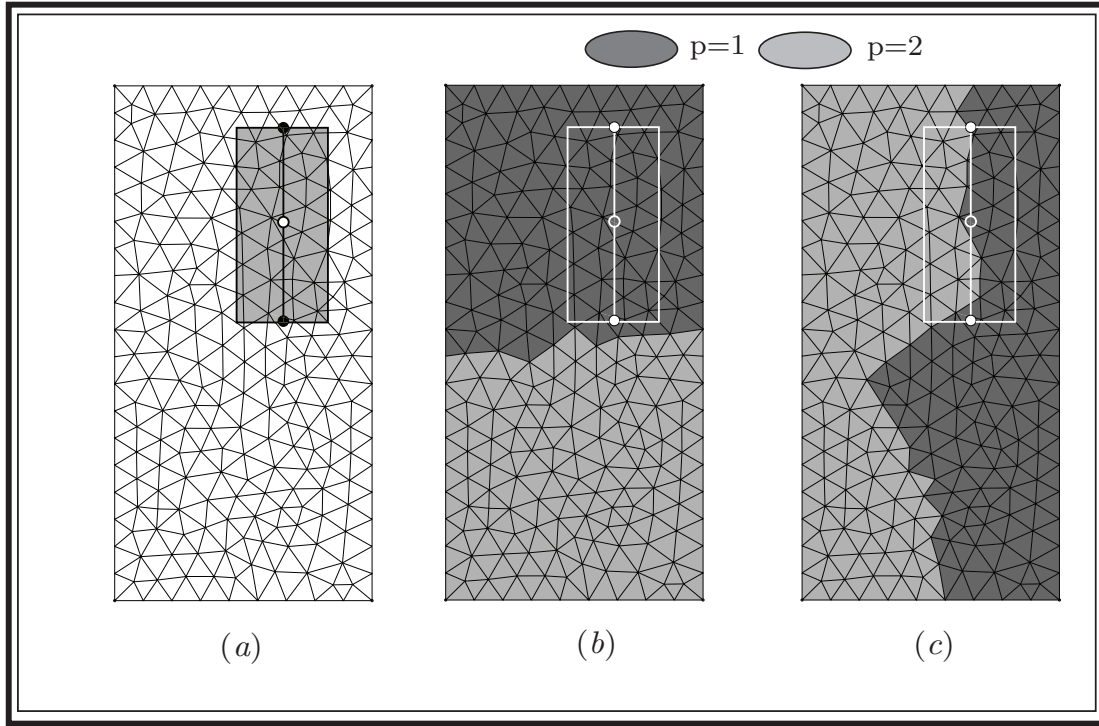
## 10.8 Subcell Model Parallelization

We now consider parallelization of the final aspect of the FVTD implementation considered in this work: the thin-wire and circuit subcell models described in Chapters 7 and 8 respectively. We begin by making the assumption that there is no field-to-wire or wire-to-field coupling: that is, the one dimensional thin-wire equations (8.14) are solved without the source term  $\mathcal{S}_j(t)$  and that the volumetric equations (3.7) are solved without the source term  $\vec{\mathcal{S}}_i(t)$  (as all coupling between the two models occurs through these source terms as described at length in Section 7.5). While this assumption completely defeats the purpose of introducing the thin-wire subcell model into the FVTD volumetric scheme, it clarifies the following argument: *Under the assumption*

*of decoupled field and subcell models, we assume that the volumetric field solution is the computational, and memory, bottleneck. Therefore, we are free to duplicate all thin-wire segments and all circuits on each processor without affecting the asymptotic memory and computational complexity of the volumetric solver.*

Now we return to the question of coupling the subcell and field models. From the perspective of each volume inside the interpolation cylinder  $\rho_0$  of a wire element  $W_j$  there is the added cost of evaluating the source term  $\vec{\mathcal{J}}_{(i)}(t)$  (7.56), contained in  $\vec{\mathcal{S}}_{(i)}(t)$ . We are assuming that the thin-wire segments exist on each processor. If we also assume that the solution at time  $n\Delta t$  for each wire segment is available locally, this cost is minimal, and can be neglected. We will return to the assumption that the wire solutions are synchronized momentarily. On the other hand, the evaluation of the source term  $\mathcal{S}_{(j)}(t)$ , for each wire segment, requires evaluating the longitudinal field integral (7.54). Each volume inside the interpolation cylinder contributes to this integral as shown, for the simple case of a single wire segment, in Figure 10.7 (a). Clearly, if we have a domain decomposition such as the one shown for two processors in Figure 10.7 (b), all elements contributing to the wire source-terms are contained on a single processor, and while the number of volumes is equal on each processor,  $p = 1$  is responsible for all the coupling computations, offsetting its computational load. A more desirable picture is shown in Figure 10.7 (c), where again we have an equal number of volumes on each of two processors, but the number of elements affecting the wire-segment are also roughly equal. In this case, the total contribution to  $\mathcal{S}_{(j)}(t)$  is the sum of the contributions on processor 1 and 2. This operation can easily be performed, and distributed to each processor using MPI. This fact maintains the synchronization of the wire solutions over all processors. To enforce a partition such as that shown in Figure 10.7, we can introduce an additional weighting scheme. For each volumetric element, we assume that the computation associated with the evaluation of





*Fig. 10.7:* A 2D visualization of the subcell-model parallelization strategy. (a) A single wire segment inside a global computational domain. The affected volumes are shown in grey. (b) A two-processor partition that is unbalanced as all wire-operations exist on one processor. (c) A two-processor partition that is balanced as an equal number of wire-influence volumes exist on each processor.

the thin-wire source term is  $q_{TW}$ . For simplicity, we will assume that each volumetric element only affects a single wire segment (though this is not generally the case in practice, and the resulting expressions can easily be modified accordingly). Then, to each element introduce the thin-wire weight:

$$\Gamma_{V_i, TW}(\ell) = \delta(TW) \cdot q_{TW} \cdot \Gamma_{V_i, LTS}(\ell) \quad (10.5)$$

where  $\delta(TW)$  is unity only if  $V_i$  affects a thin-wire segment and where  $V_i$  operates at LTS level  $\ell$ . Adding this result into the ERBC and LTS weights computed in the

previous sections we obtain the total weight associated with each volumetric element:

$$\boxed{\Gamma_{V_i} = (1 + \delta(\partial V)q_{ERBC} \cdot M_{\partial V_S} + \delta(TW) \cdot q_{TW}) \cdot \Gamma_{V_i,LTS}(\ell)} \quad (10.6)$$

### 10.9 Determination of the Volumetric Weights

In order to determine the weights  $\Gamma_{V_i}$  associated with each volumetric element, we must pre-process the global domain  $V$ . This could be accomplished as follows:

- Perform a parallel mesh read.
- Perform a domain decomposition using unit-weight associated with each volume and distribute the volumetric elements.
- *Do Not* compute the halos.
- Determine the LTS levels.
- Duplicate the ERBC surfaces and subcell models on each processor.
- Determine the number of global boundary elements and ERBC surface elements.
- Determine the elements influenced by thin-wires.
- Compute the weights according to (10.6).
- Repartition and redistribute the mesh using the computed weights.
- Compute and obtain the appropriate halo levels.

The constants  $q_{ERBC}$  and  $q_{TW}$  are both dependent on the accuracy of the implementation and on the computational architecture. They can easily be estimate by considering the operations required, but can also be empirically determined by simulation.

### 10.10 Validation

The parallelization strategies presented in this section have been implemented using MPI and Zoltan for a high-performance computing cluster running a Linux flavoured operating system. The cluster has 32 compute nodes, each with 2 quad-core Intel Xeon X5355 processors running at 2.66 [GHz]. In total there are 256 cores. Each node is equipped with 16 [GB] of memory and are connected by a high-speed Infiniband network permitting roughly 2 [GB/s] communication rates.

As mentioned at the beginning of this chapter, only the general parallelization schemes have been performed. That is, in all schemes a weight of unity is associated with each volumetric element. This will not generally result in ideal load-balancing but there is no theoretical or practical reason why the weighting schemes presented in this chapter will not remedy this issue.

For all aspects of the FVTD engine, including local-time-stepping, global mesh-truncation and subcell models, the numerical results produced by the serial and parallel codes are equal to machine precision and no comparative simulation results are provided in what follows. Instead the validation focuses on the scalability of the parallel algorithms.

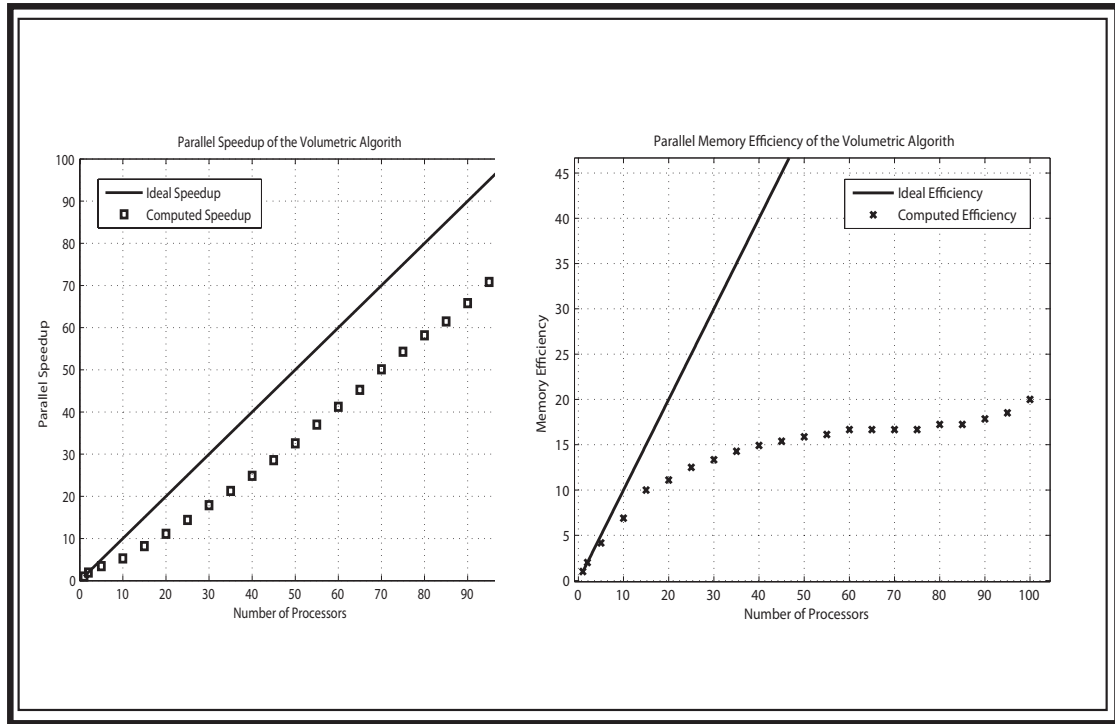


Fig. 10.8: Parallel efficiency scaling of the volumetric solver. (a) Parallel speedup and (b) memory scaling, normalized to the serial case. An ideal scaling curve is provided for reference.

### 10.10.1 Parallel Efficiency of the Volumetric Solver

A summary of the parallel scalability of the volumetric solver (without LTS or mesh-truncation) is shown in Figure 10.8. Simulations were run up to 100 processing cores for the PEC scattering from a sphere where the computational domain consisted of 600,000 volumetric elements. The left side of the figure shows the acceleration as a function of the number of processors up to 100 processing cores. The plot is normalized to the computational time on a single processor, which required nearly one hour and 1 [Gb] of RAM to compute 500 time-steps (of a MUSCL, predictor-corrector scheme). The acceleration is quite good, with the deviation from the ideal curve being accounted for by the overhead of updating the halo elements at each time-step. The right side of the figure shows the memory scaling, again normalized to the case of a single processor

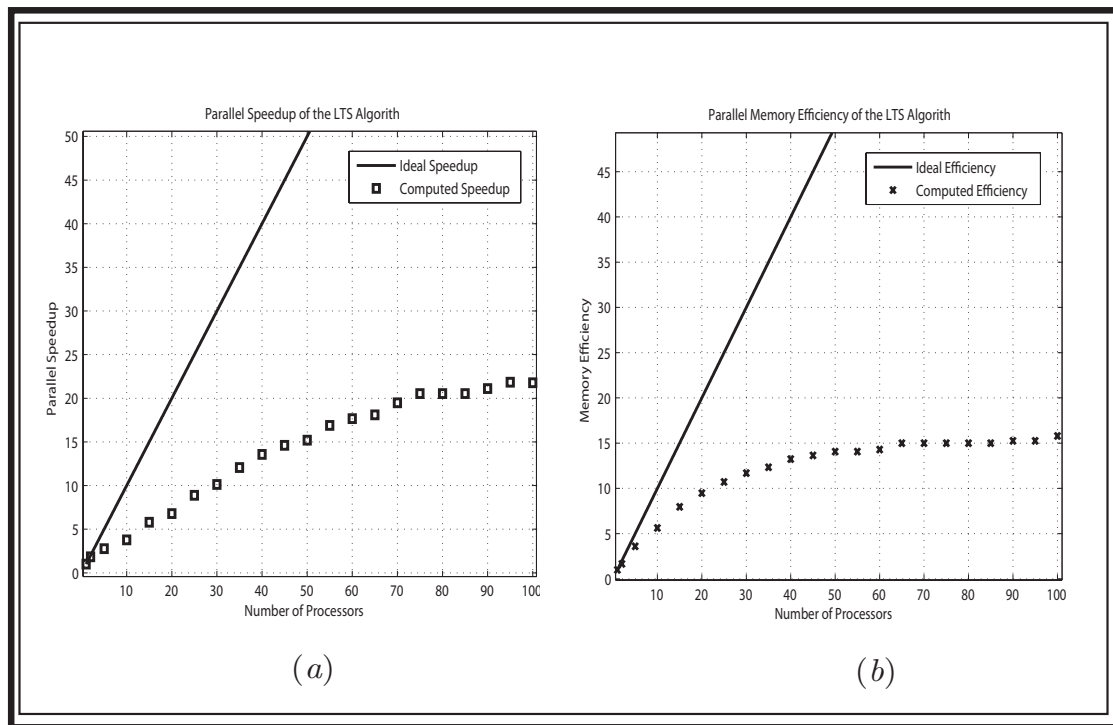


Fig. 10.9: Parallel efficiency scaling of the LTS Scheme. (a) Parallel speedup and (b) memory scaling (right), normalized to the serial case. An ideal scaling curve is provided for reference.

and taken as the worst-case memory over all processors. The memory scaling shows the required memory overhead of the halo elements. Using 100 processing cores, this problem requires less than one minute to compute 500 time-steps at a cost of less than 50 [Mb] per core.

### 10.10.2 Parallel Efficiency of the Local-Time-Stepping Scheme

We now consider the LTS parallelization by running the same mesh used in Section (6.3) to validate the serial algorithm. As in the previous example, this mesh has approximately 600,000 elements. The scaling curves are shown in Figure 10.9. Clearly, the LTS scaling is not nearly as good as the straight volumetric scaling. This is because

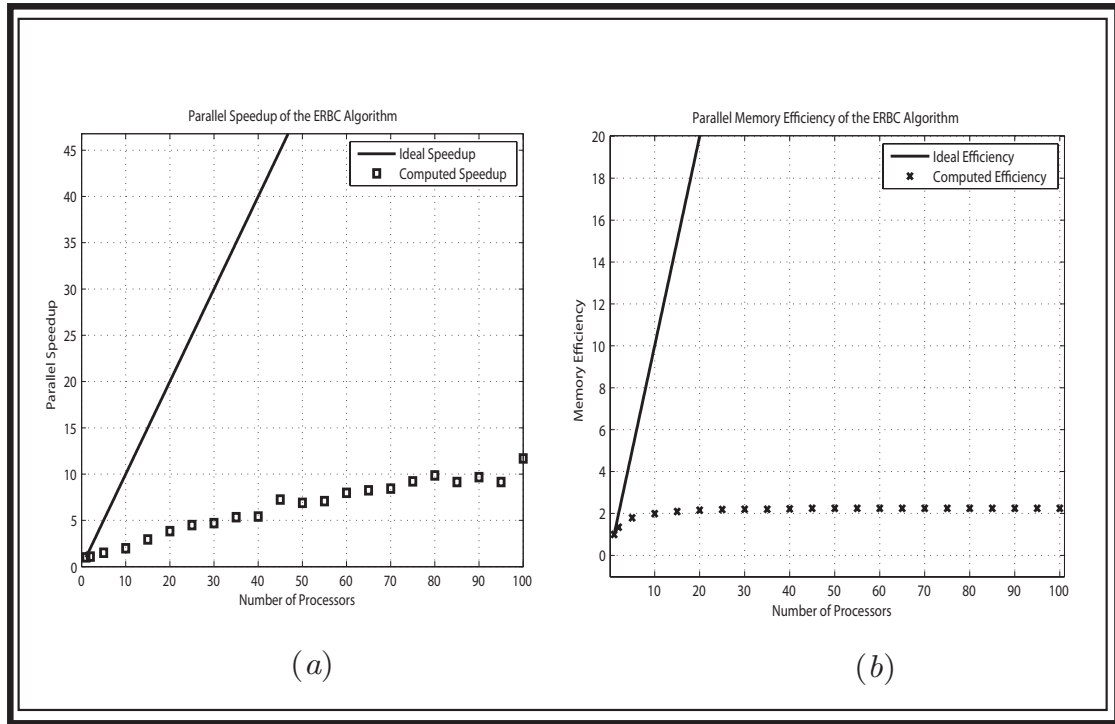


Fig. 10.10: Parallel efficiency scaling of the global mesh-truncation Scheme. (a) Parallel speedup and (b) memory scaling (right), normalized to the serial case. An ideal scaling curve is provided for reference.

the appropriate weights for each volume have not been used. In the serial case, this problem required 6 minutes and 900 [Mb] of memory to compute 500 time steps. Above 50 processors, this is reduced to 30 seconds and 70 [Mb] of memory per processing core.

### 10.10.3 Parallel Efficiency of the Global Mesh-Truncation Scheme

We now return to the example run of the global mesh-truncation scheme validated in Section 6.4. This example had 500,000 volumetric elements, 6,300 integration surface elements and 2700 boundary surface elements. The parallel scaling results are shown in Figure 10.10. It is not surprising that neither the speedup, nor the memory efficiencies are terribly good. The maximum back-storage of the solution required 1,100 time

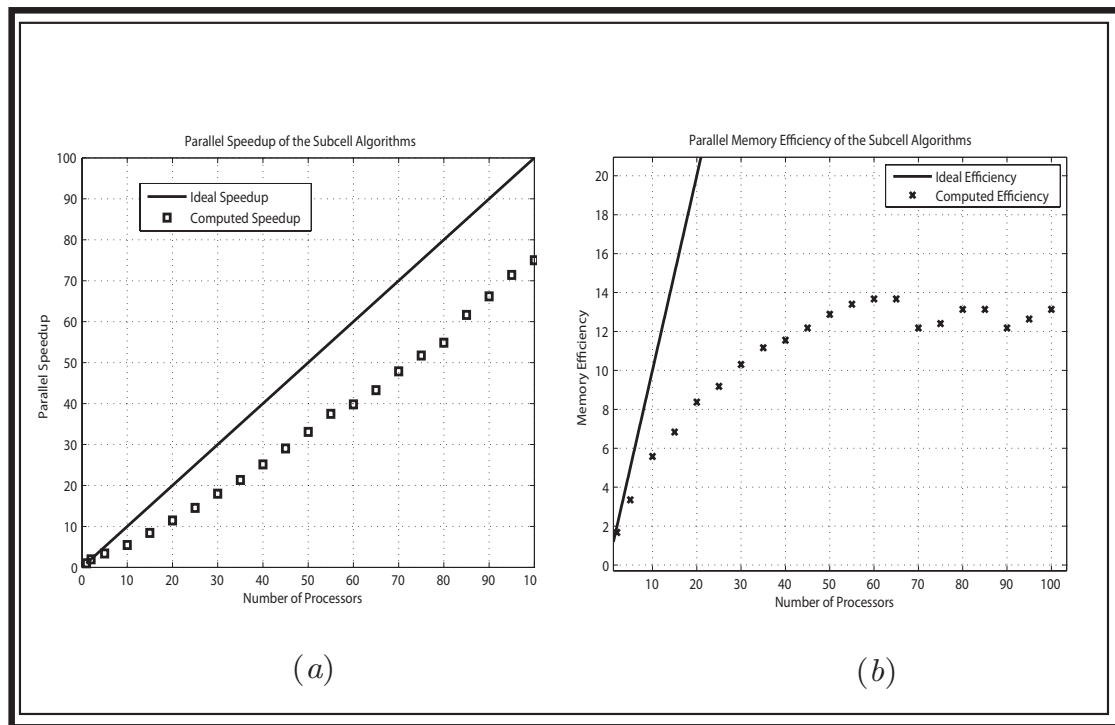


Fig. 10.11: Parallel efficiency scaling of the subcell models. (a) Parallel speedup and (b) memory scaling (right), normalized to the serial case. An ideal scaling curve is provided for reference.

samples. Multiplying this by 2,700 boundary surface elements and 6 field components requires hundreds of megabytes of storage for the ERBC integrals alone. The serial case required 1.1 [Gb] of memory and 10 minutes to compute 10 time steps. Above 20 processors, this same run requires roughly 400 [Gb] of memory. Using 100 cores, the 10 time-steps were computed in 45 [s]. The non-monotonic nature of the speedup is explained by the fact that the number of surface elements is not equally balanced on all processors and the maximum number of boundary facets on a single processor actually increases, in some cases, as the number of processors is increased.

*10.10.4 Parallel Efficiency of the Subcell Models*

Finally, we investigate the parallel efficiency obtained for the subcell models. We reconsider the example of the MRI-tuned loop presented in Section 9.4. The volume contained 450,000 volumetric cells and 48 wire segments. The efficiencies are shown in Figure 10.11. The speedup is quite good, mostly because the subcell-models are localized and do not introduce that much overhead for this example consisting of only 48 wire segments. We do, however, see that the memory scaling is affected by the partitioning. A large amount of the thin-wire coupling is pre-computed and stored. Therefore, as the wire-segments become more-or-less localized on each processor, the worst-case memory can vary according to the number of elements affected by wires on a given processor. In serial, this example required 450 [s] to compute 100 time steps and 600 [Mb] of memory. For 100 processing cores, the memory is reduced to 45 [Mb] and 6 [s].

*10.11 Chapter Summary*

In this chapter, all aspects of the FVTD solver have been parallelized. While the ideal load-balancing schemes presented in this section have not been validated, each part of the FVTD scheme has been parallelized for uniform weighting over the computational domain. Though the FVTD solver is not heavily optimized, the parallel scalability shown herein suggests that it is capable of solving problems involving hundreds of million of unknowns with fewer than 100 processors using both reasonable memory and computational time. Pushing the limits of the solver will be the focus of future work.



# Finite-Volume Frequency Domain Solutions of Maxwell's Equations

---

*When I was a boy, I had a clock with a pendulum that could be lifted off. I found that the clock went very much faster without the pendulum. If the main purpose of a clock is to go, the clock was the better for losing its pendulum. True, it could no longer tell the time, but that did not matter if one could teach oneself to be indifferent to the passage of time. – Bertrand Russell*

The FVTD scheme presented in this thesis is capable of producing broad-band system response in a single execution. When discrete frequency solutions are required (e.g. in the field of microwave tomography for biomedical imaging [84]), a frequency-domain solution may be more efficient. It is interesting that the upwind, flux-split FVTD scheme can be converted to the frequency domain under time-harmonic assumptions without any major modifications to the underlying theory. The result is a Finite-Volume Frequency-Domain (FVFD) method for solving the time-harmonic Maxwell's

---

Equations. Since its apparent introduction in [85], FVFD has begun to garner attention due to its geometric modeling capabilities [17]. FVFD is not, however, a common solution tool in computational electromagnetics.

Returning, briefly, to the time domain, one of the benefits of FVTD over finite-element based time-domain codes (FETD) is that a sparse matrix inversion is not required at each time-step [10]. As will be made clear shortly, when frequency domain solvers are considered, Maxwell's curl equations result in an elliptic system of ordinary differential equations and FVFD requires the solution to a sparse matrix equation. Frequency-domain finite-element methods (FEM) [86] also require solving a sparse matrix equation. As FEM theory for solving electromagnetic problems is so rich, one might ask why an FVFD solver is necessary. Two features of the FVFD formulation make it an attractive alternative to FEM: first, the numerical algorithm is easily implemented from existing FVTD codes, and second, the FVFD formulation simultaneously solves for both electric and magnetic fields (unlike FEM which typically solves for one or the other [86]).

In this chapter, the FVFD solution to Maxwell's Equations is developed directly from the FVTD theory presented in this thesis. The resulting global sparse matrix system is solved iteratively. As will be shown, the spatial reconstruction schemes can be maintained from FVTD. A brief, empirical study of the effects of flux-reconstruction on the accuracy and convergences of the iterative solution will be presented and a possible benefit of using different flux-reconstruction schemes during the iterative solution process is proposed. For completeness, the solution is formulated in parallel using the domain-decomposition techniques discussed in Chapter 10.

The FVFD discussion herein focuses solely on the volumetric solution of Maxwell's Equations. Global mesh-truncation schemes and the conversion of the subcell models

to the frequency domain have been attempted, but insufficient theory and/or positive results are available to warrant discussion herein.

### 11.1 The Global FVFD System

We recall from Chapter 4 that the global system of FVTD equations can be written as:

$$d_t \underline{\vec{u}}(t) = -\underline{\vec{\mathcal{F}}}(\underline{\vec{u}}(t)) + \underline{\underline{\Sigma}} \cdot \underline{\vec{u}}(t) + \underline{\vec{\mathcal{S}}}(t) \quad (11.1)$$

where the  $6N_V$  length global solution vector  $\underline{\vec{u}}(t)$  contains the volumetric average of the electric and magnetic field components in each of the  $N_V$  volumes in the partition  $\{V_i\}$  of the global domain  $V$ . The global flux vector  $\underline{\vec{\mathcal{F}}}$  is the result of spatially integrating the reconstructed boundary flux over the surface of each  $V_i$  and can be viewed, according to (3.46), as a sparse global matrix multiplying  $\underline{\vec{u}}(t)$ :

$$\underline{\vec{\mathcal{F}}}(\underline{\vec{u}}(t)) = \underline{\underline{\mathcal{F}}} \cdot \underline{\underline{\mathcal{R}}} \cdot \underline{\vec{u}}(t) \quad (11.2)$$

The matrix  $\underline{\underline{\mathcal{R}}}$  is responsible for the reconstruction of boundary states and is dependent on the reconstruction scheme selected. The matrix  $\underline{\underline{\mathcal{F}}}$  performs the integration of the flux (a function of the boundary states) over each finite-volume. We can convert the system of equations (11.1) to the frequency domain by assuming that Maxwell's Equations are time-harmonic with radial frequency  $\omega = 2\pi f$  where  $f$  is the time-harmonic frequency [35]. According to Fourier Theory, and assuming an time-dependence  $e^{j\omega t}$ ,  $e$  being Euler's number and where  $j = \sqrt{-1}$ , the time-derivative in (11.1) can be replaced by  $j\omega$  [35] resulting in:

$$j\omega \underline{\vec{u}}(\omega) = -\underline{\vec{\mathcal{F}}}(\underline{\vec{u}}(\omega)) + \underline{\underline{\Sigma}} \cdot \underline{\vec{u}}(\omega) + \underline{\vec{\mathcal{S}}}(\omega) \quad (11.3)$$

where, without having to introduce additional notation, it is to be understood that by simply replacing  $t$  by  $\omega$  in  $\vec{\mathbf{U}}(t)$  we imply a Fourier Transform [87]. Collecting terms and defining the global identity matrix  $\underline{\mathbf{I}}$  we have:

$$\boxed{(j\omega\underline{\mathbf{I}} - \underline{\mathbf{\Sigma}} + \underline{\mathbf{F}} \cdot \underline{\mathbf{R}}) \cdot \vec{\mathbf{U}}(\omega) = \vec{\mathbf{S}}(\omega)} \quad (11.4)$$

which is a global matrix equation for determining  $\vec{\mathbf{U}}(\omega)$  due to the sources  $\vec{\mathbf{S}}(\omega)$ . By iteratively inverting the matrix we obtain the cell-average fields in each volume which can be used to reconstruct the solution at any location inside  $V$ .

## 11.2 Spatial Reconstruction and Electromagnetic Boundary Conditions

Having converted the FVTD formulation to an FVFD formulation, we note that all of the spatial reconstruction schemes discussed in Chapter 3 are still applicable if we permit a “upwind” and flux-split FVFD formulation. Here the concept of upwinding is in the context of time-harmonic field-propagation, but holds nonetheless. Under time-harmonic assumptions, another type of flux-reconstruction is possible, as presented in [85], the so-called “harmonic” reconstruction. This approach uses the method of characteristics to account for the *phase* change between barycentric values at  $\vec{r}_i$  in  $V_i$  and the boundary evaluation point  $\vec{r}$  on facet  $k$  of volume  $i$  [85] (recall that the barycentric value can be approximated to second-order accuracy from the volumetric average - see Section 3.7)

$$\boxed{\vec{\mathbf{U}}_{i,k}^+ = \vec{\mathbf{U}}_i \exp(-j\omega\sqrt{\epsilon_i\mu_i}|\vec{r} - \vec{r}_i|)} \quad (11.5)$$

This reconstruction is slightly more computationally expensive than the first-order implementation but less so than MUSCL reconstruction. It is not strictly second order accurate. We might interpret it as lying somewhere between first- and second-order accuracy.

### 11.3 *Iterative Solutions, Parallelization and (a lack of) Preconditioners*

The iterative solution to the global system of equations (11.4) requires an iterative matrix solver. First, we note that in order to maintain code-scalability for a large number of finite-volumes, we are not computing or storing a global matrix. Instead, this matrix is computed by producing the rows corresponding to  $V_i$  on-the-fly. As we do not have a global matrix, it is not immediately obvious how to obtain its Hermitian and so we do not consider the conjugate-gradient algorithm [88]. Instead, both the GMRES algorithm [88] and BICGSTAB algorithm [89] have been implemented. In the pioneering work [85], Bonnet *et. al* suggest that BICGSTAB converges faster than GMRES. While this may be true, we have elected to use GMRES simply because GMRES converges monotonically whereas BICGSTAB does not.

Either of the algorithms GMRES or BICGSTAB function by computing matrix-vector products and evaluating appropriate norms of the resulting solution estimates. In the time-domain, the time-evolution of the global system (11.1) can be interpreted analogously to a single matrix-vector-product of the frequency-domain system (11.4). Therefore, all of the parallelization developed for the time-domain volumetric solution in Chapter 10, holds. The only modification required to ensure global functionality of the iterative algorithms is to determine global vector norms from local quantities.

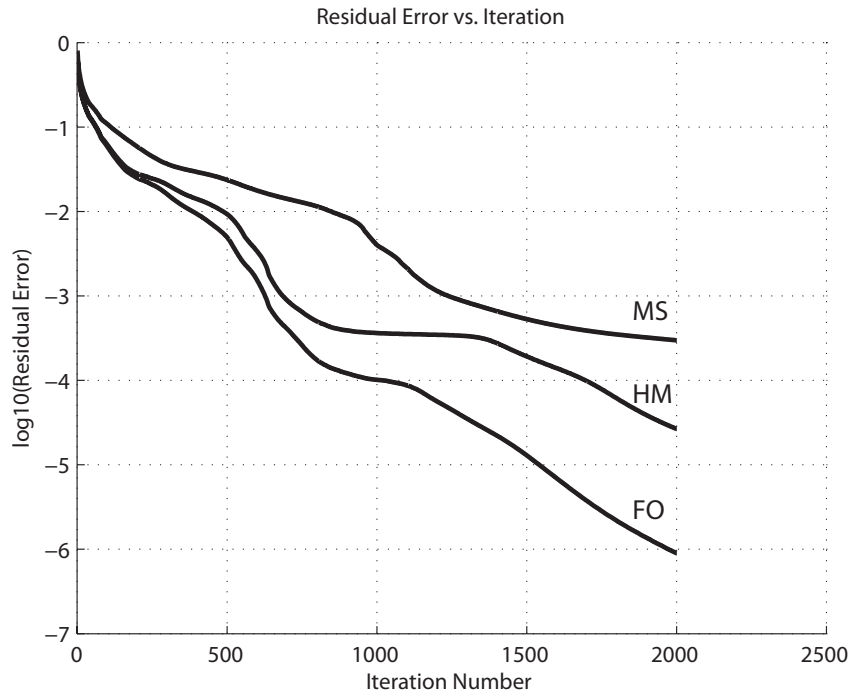
This is easily accomplished in constant time using MPI.

One of the greatest challenges when solving elliptic problems using iterative solvers is to construct efficient preconditioners that minimize the number of iterations required to obtain a global solution to a desired level of accuracy [88]. While the applicability of FVFD for real-world problems may be greatly hindered by a lack of preconditioning, we do not consider preconditioners in this work.

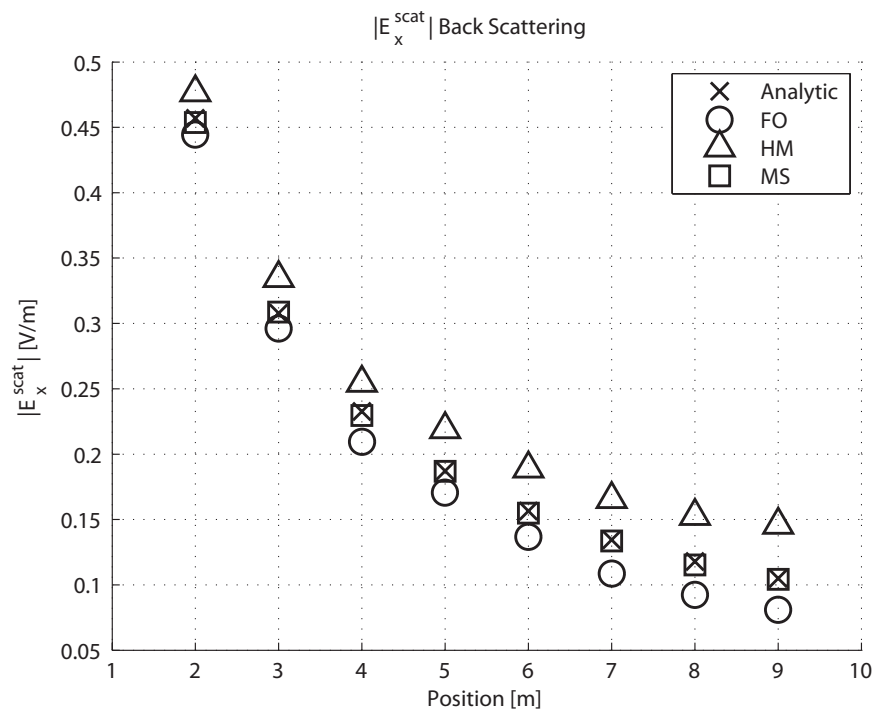
### 11.4 Numerical Results

To validate the conversion of the volumetric FVTD code to a volumetric FVFD code, we investigate the simple problem of incident field scattering from a PEC sphere, as an analytic solution is available [35]. In addition, we are interested in a brief empirical study of the effects of flux reconstruction on both the accuracy and iterative convergence of the FVFD algorithm. We consider a 1 [m] radius PEC sphere centered in a 10 [m] radius spherical computational domain truncated with the local Silver-Müller mesh truncation scheme presented in Section 5.1. A mesh of the volumetric domain was produced using Gmsh [36] and the discretization resulted in 2612 triangular facets on the PEC surface and approximately 1.16 million first-order tetrahedral elements in the computational domain. Correspondingly, the global matrix equation relates roughly 7 million unknowns. As the mesh supports a maximum frequency of approximately 75 [MHz] at one-tenth of a wavelength we compute the scattered field of a 50 [MHz]  $z$ -incident and  $x$ -polarized plane-wave. The iterative solution was obtained using GMRES without any preconditioning.

In Figure 11.1(a) we show the residual relative norm of the matrix equation as a function of GMRES iteration for the first-order (FO), harmonic (HM) and MUSCL

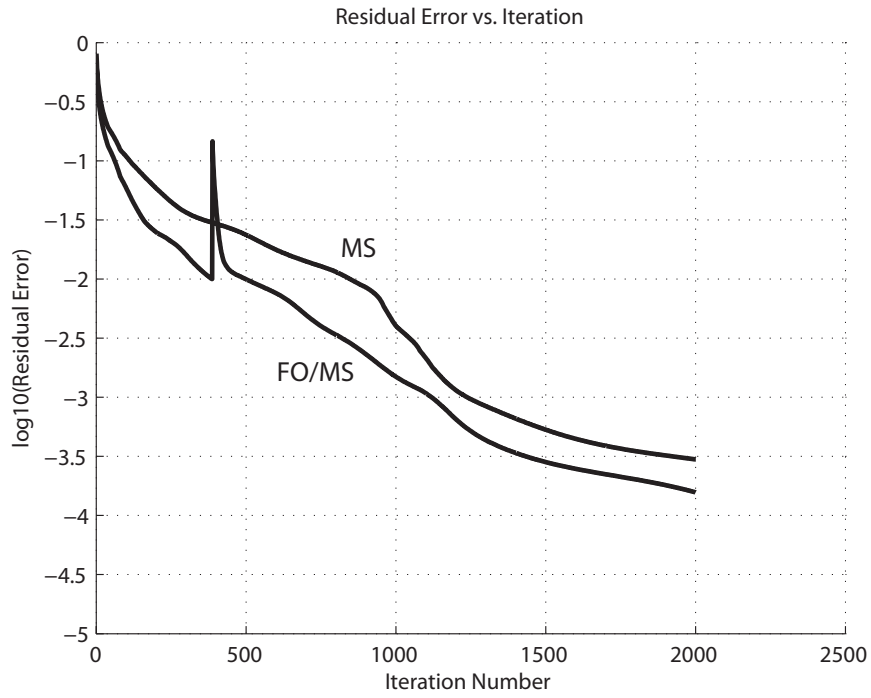


(a)

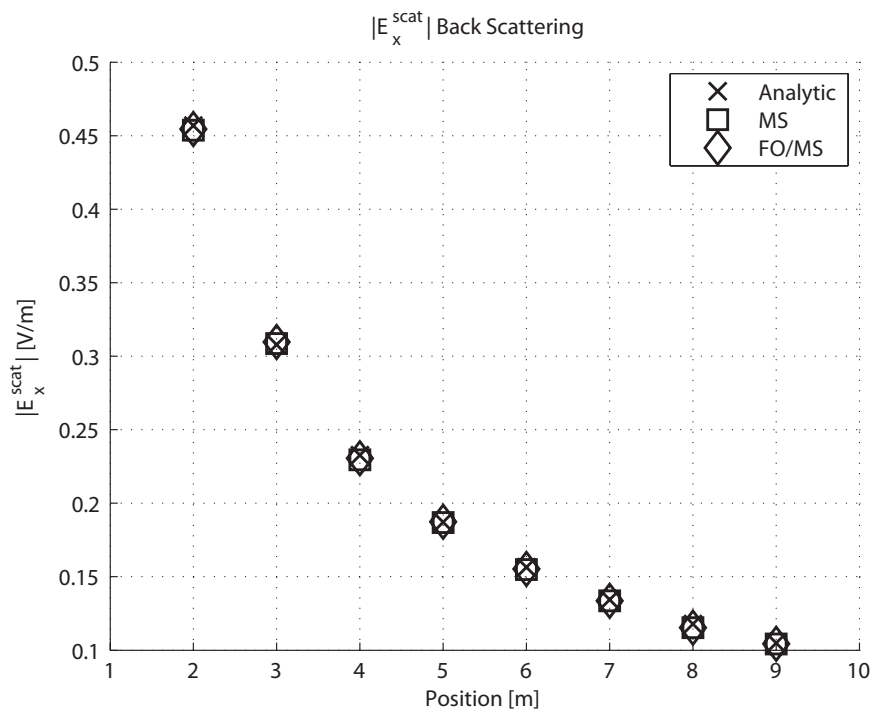


(b)

Fig. 11.1: (a) FVFD GMRES convergence rate for different flux reconstruction schemes. (b) Backscattered solution for different flux reconstruction schemes. The analytic solution is provided for reference.



(a)



(b)

Fig. 11.2: (a) FVFD GMRES convergence rate for the FO/MS hybrid scheme vs. the MS scheme. The discontinuity shows where the FO solution is used as an initial guess for the MS system. (b) Backscattered results for the FO/MS and MS systems.



(MS) reconstructions to 2000 iterations. In Figure 11.1(b) we show the reconstructed scattered field  $|\vec{\mathcal{E}}_x^{scat}|$  at back-scatter locations from  $z = 2$  [m] to  $z = 9$  [m] in unit steps. The analytic solution is provided for reference. From the plots we see that both the FO and HM solutions converge much faster than then MS reconstruction but to an inaccurate solution. The inaccuracies appear to increase away from the scatterer (sources). We also note that the MS reconstruction is visually more accurate despite having a significantly higher residual after 2000 iterations. A cross-sectional image of the scattered electric and magnetic fields, produced using the parallel visualization tool ParaView [38] is shown for the MUSCL reconstruction in Figure 11.3.

### 11.5 A Novel Iterative Approach

Motivated by the quick convergence of the computationally less expensive FO reconstruction, we consider solving the matrix equation by first using the FO reconstruction to obtain an initial guess for the MS reconstruction. As we are not storing the global matrix, there is no need to re-evaluate the flux-reconstruction part of the system, and we can simply switch reconstruction schemes dynamically during the iterative solution process. We denote this hybrid reconstruction scheme as FO/MS. Figure 11.2(a) shows the convergence rate of a FO/MS scheme. The discontinuity of the convergence indicates the location where the FO solution was used as an initial guess to the MS system. For this simple example, the FO/MS scheme converges to a lower residual error in the same number of iterations than the MS scheme alone.

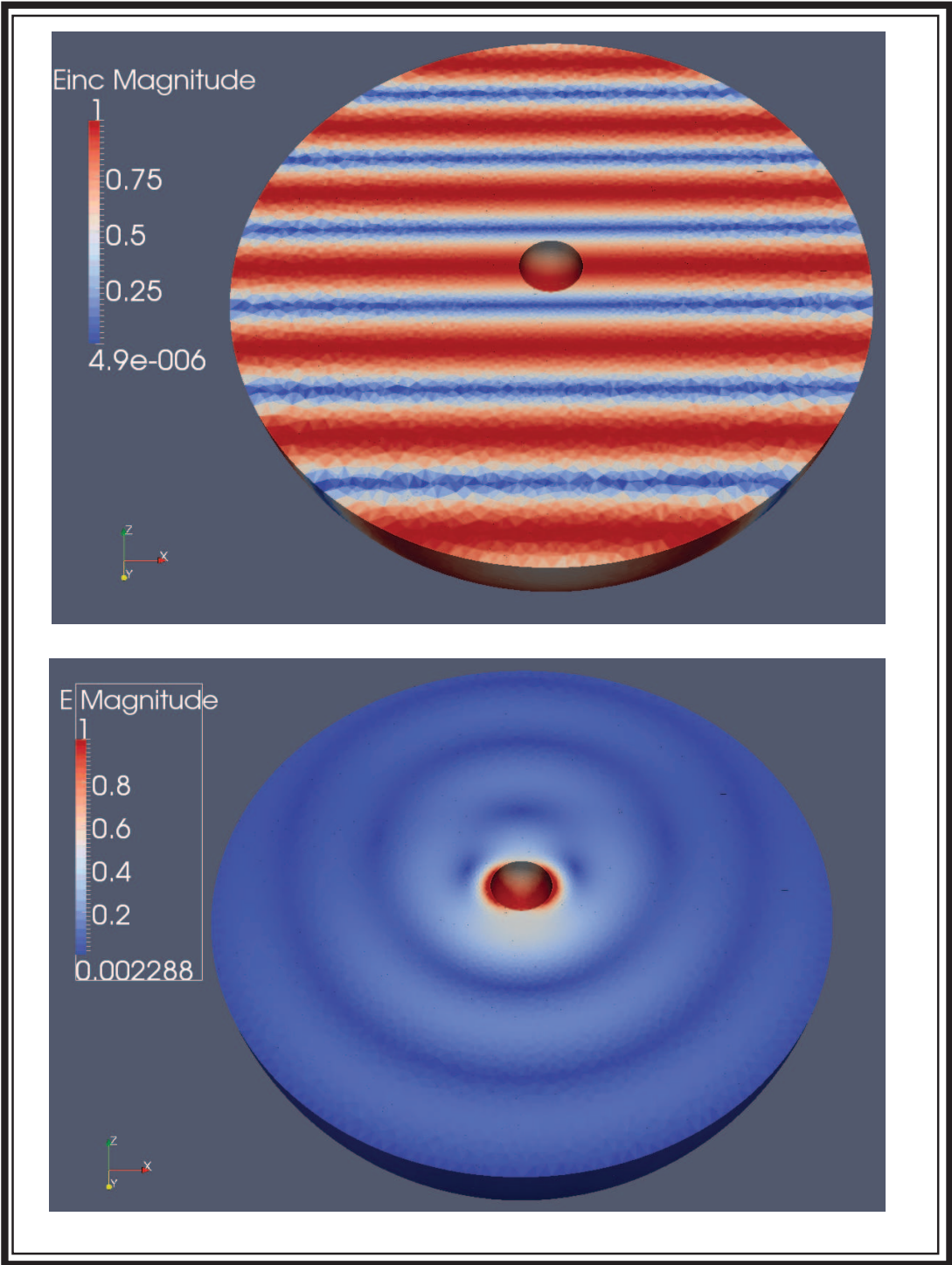


Fig. 11.3: (Colour) FVFD solution of the scattering from a PEC sphere. The magnitude of the incident field (top) and scattered field (bottom) are shown.

*11.6 Chapter Summary*

In this chapter, the small modifications required to convert an upwind and flux-split FVTD scheme to the frequency domain have been provided. The implemented algorithm is completely parallelized for distributed systems, but is limited to the consideration of simple volumetric problems in the absence of global mesh truncation and subcell-models. We have empirically investigated the effects of first-order, harmonic, and MUSCL flux reconstruction schemes on the iterative convergence of the FVFD solution to Maxwell's Equations for the simple problem of scattering from a PEC sphere. While low-order reconstructions converge faster than the MUSCL scheme, they do not accurately model the underlying system of differential equations. It has been demonstrated that (for the scattering problem considered herein) using a hybrid low-order/MUSCL solution can improve on the iterative convergence rate. Determining either an analytical or empirical rule for such hybrid schemes will be the focus of future work. Furthermore, the iterative solution to the resulting global system of equations has been performed without preconditioning. For real-world problems, FVFD convergence rates are typically poor [17] and determining efficient preconditioners and/or model or reduction methods (as in [17]) will be critical for making the proposed FVFD scheme a viable numerical algorithm.

# Conclusions and Future Improvements

---

*Un bon mot ne prouve rien – Voltaire, Translation: A witty saying proves nothing.*

## *12.1 Summary of Accomplishments*

The work presented in this dissertation constitutes the development of an upwind, and flux-split Finite-Volume Time-Domain numerical tool for solving Maxwell's Equations on unstructured grids and serves as the first steps towards a robust numerical tool for solving real-world engineering problems. While the initial theory, presented in Chapters 2 and Chapters 3, at first appears daunting, one quickly realizes that, by storing cell-average field values and ensuring that discrete jumps in constitutive parameters

resides precisely at cell-interfaces, this flavour of FVTD contains an inherent simplicity that makes geometric modeling of complicated structures straight-forward. Furthermore, unlike the standard FDTD algorithm where time- and spatial accuracies are inherently linked, the cell-centered collocated time approach used provides (at least theoretically) an obvious direction towards higher-order accuracy.

To improve the computational cost of the developed FVTD engine, the local-time stepping scheme of Fumeaux *et. al* [15] has been implemented without modification in Chapter 4. In addition a global mesh-truncation scheme, based on the evaluation of Huygens' surface integrals has been presented in Chapter 5 and has been shown to permit a reduction in the global computational volume for specific problems.

The primary contributions of this thesis were the development of the upwind and flux-split thin-wire model presented in Chapter 7. The model is the FVTD equivalent of the unconditionally stable models of Eldelvik *et. al* [18], [19], and permits thin wires to be arbitrarily oriented within the volumetric domain. Further, the theory developed in Chapter 8 of a novel flux-split boundary condition that permits the thin-wire model to be coupled to arbitrary lumped-element multi-port circuits introduces incredible flexibility when modeling thin-wires structures interconnected using electrically small circuits. Specifically, the combined application of these circuit models appears to have great promise for modeling the radio-frequency (RF) components of Magnetic Resonance Imaging (MRI) systems.

As parallel computational resources become more and more abundant, the parallelization of the FVTD algorithm makes the tool that much more capable. The general approach presented in Chapter 10 is the secondary contribution of this thesis and provides a straight-forward method that ensures parallel efficiencies for both computations and memory. Furthermore, the required steps to efficiently parallelize all of the FVTD

features presented in this work has been given in detail.

Finally, the frequency-domain FVFD equivalent of the flux-split and upwind FVTD scheme has been implemented, in parallel, and has shows some promise. As described in Chapter 11, the small but novel idea that different flux reconstruction schemes can be used to improve the iterative convergence of the method is the third, and final contribution of this work.

## *12.2 Future Work: Modeling Capabilities*

There are a countless number of additional features and improvements to the implemented FVTD/FVFD algorithms that are possible without modifying the underlying upwind and flux-split formulation. The most obvious are:

- **Source Modeling:** Other than the scattered-field and circuit-driven thin wire formulations provided in this thesis, the ability of the current implementation to model sources is somewhat limited. Baumann [13] provide a very interesting way of imposing incoming fluxes across specially defined port-planes. The planes are placed inside waveguide structures where an analytic description of a known field-mode can be imposed. Using the same plane to capture the time-evolving field provides a very elegant way of determining the system S-parameters.
- **Post-Processing:** For the computation of radar cross sections, a far-field transformation should be implemented. This could be achieved directly from the global mesh-truncation scheme under far-field approximations.
- **Boundary Conditions:** One type of boundary condition omitted from this work has been the Perfectly Matched Layer (PML) conditions originally pre-

sented by Berenger [90]. Recently, these conditions have been formulated for FVTD [34] and their incorporation into the developed code would provide greater functionality for problems involving nearly closed boundaries such as open-ended waveguides.

- **Material Modeling:** Throughout this work isotropic, stationary and simple media has been assumed. Including anisotropy could make the developed code applicable to a larger number of interesting problems.
- **Thin-Wire Modeling:** The primary draw-back of the thin-wire model developed in Chapter 7 is that at multi-wire junctions it is still unclear how to distribute the volumetric current density in a smooth manner. One possibility is to surround all wires with a conforming tube and to remove the volumetric discretization within the tube completely. It may then be possible to impose the effects of the wire directly as incoming boundary fluxes at the surface of the tube. The geometrical flexibility of FVTD certainly makes such a scheme possible.
- **Frequency-Domain Preconditioning:** Developing an appropriate preconditioner or model-order reduction technique to improve the iterative convergence of the FVFD algorithm presented in Chapter 11 is critical for making the FVFD algorithm applicable to real-world problems.
- **Frequency-Domain Subcell Models:** Developing the appropriate theory to solve the coupled circuit-thin-wire-field equations in the frequency domain would make an FVFD a suitable forward-solver for inverse imaging applications in Microwave Tomography [91].
- **Global Boundary Condition Acceleration:** The use of time-domain acceleration techniques such as the plane-wave time-domain (PWTD) method [92, 32] could be used to accelerate the evaluation of the global mesh truncation scheme.

### 12.3 Future Work: Algorithmic Considerations

At the fundamental level this work has assumed an upwind and flux-split FVTD formulation. The most stringent limitation imposed by such an approach is that upwind solutions for hyperbolic systems are inherently dissipative [11]. While this problem can, to some extent, be dealt with by using a sufficiently fine discretization of the computational domain, a MUSCL FVTD algorithm does not seem to be the appropriate way to solve electromagnetic problems involving very large electrical scales (either the geometry itself or the distance between disjoint geometries). “For problems of this type the (upwind FVTD methods) may not be appropriate... (as) they have a high cost per grid cell relative to simpler finite difference methods” [11]. While it has already been established in Chapter 1 that there is still a large amount of real-world problems that are solved efficiently by FVTD, higher-order approaches, algorithmic modifications and/or hybrid methods are required to solve electrically large problems. Recent developments in higher-order FVTD using either ENO or WENO reconstructions may resolve some of these issues [26, 20]. Alternative FVTD implementations that do not require upwinding such as the leap-from scheme presented in [16, 93] may also be of interest, although they appear to suffer from grid-decoupling problems [17] (and also witnessed first-hand in a preliminary implementation for scattered fields). Finally, the Discontinuous Galerkin Time-Domain (DGTD) method, which attempts to balance the benefits of FETD and FVTD in a higher-order framework [10], has recently been shown to have great potential [94, 95, 96]. Ultimately, however, it is likely that the most robust numerical solvers will incorporate a hybrid combination of available time-domain techniques: accurate geometric modeling can be achieved using unstructured algorithms such as FVTD, while long propagation distances can be efficiently computed by means of Huygens’ surface boundary conditions or computationally effective



finite-difference schemes [97, 98].

# APPENDIX

A

# Derivation Details for the Conservative Form of Maxwell's Equations

---

The purpose of this appendix is to derive equation 2.17 from equation 2.8. The notation and formulation is close to that in [9]. Equation (2.8), is repeated here for convenience:

$$\bar{\boldsymbol{\alpha}}(\vec{r}) \cdot \partial_t \vec{\boldsymbol{u}}(t, \vec{r}) + \vec{\boldsymbol{\kappa}}(t, \vec{r}) = \bar{\boldsymbol{\sigma}}(\vec{r}) \cdot \vec{\boldsymbol{u}}(t, \vec{r}) + \vec{\boldsymbol{s}}(t, \vec{r}). \quad (\text{A.1})$$

Re-writing the quantity  $\vec{\mathcal{K}}(t, \vec{r})$  as:

$$\vec{\mathcal{K}}(t, \vec{r}) = \partial_x \begin{bmatrix} 0 \\ \mathcal{H}_z \\ -\mathcal{H}_y \\ 0 \\ -\mathcal{E}_z \\ \mathcal{E}_y \end{bmatrix} + \partial_y \begin{bmatrix} -\mathcal{H}_z \\ 0 \\ \mathcal{H}_x \\ \mathcal{E}_z \\ 0 \\ -\mathcal{E}_x \end{bmatrix} + \partial_z \begin{bmatrix} \mathcal{H}_y \\ -\mathcal{H}_x \\ 0 \\ -\mathcal{E}_y \\ \mathcal{E}_x \\ 0 \end{bmatrix} = \partial_x \vec{\mathcal{G}}_x(t, \vec{r}) + \partial_y \vec{\mathcal{G}}_y(t, \vec{r}) + \partial_z \vec{\mathcal{G}}_z(t, \vec{r}) \quad (\text{A.2})$$

for the source-free and lossless case we get

$$\begin{aligned} \partial_t (\vec{\alpha}(\vec{r}) \cdot \vec{\mathcal{U}}(t, \vec{r})) + \partial_x \vec{\mathcal{G}}_x(t, \vec{r}) + \partial_y \vec{\mathcal{G}}_y(t, \vec{r}) + \partial_z \vec{\mathcal{G}}_z(t, \vec{r}) = \\ \partial_t (\vec{\alpha}(\vec{r}) \cdot \vec{\mathcal{U}}(t, \vec{r})) + \nabla \cdot \vec{\mathcal{G}}(t, \vec{r}) = \vec{0} \end{aligned} \quad (\text{A.3})$$

The divergence is to be taken across each row of  $\vec{\mathcal{G}}(t, \vec{r}) = [\vec{\mathcal{G}}_x \ \vec{\mathcal{G}}_y \ \vec{\mathcal{G}}_z]$ . Inside a finite-volume with constant material parameters we may write

$$\partial_t \vec{\mathcal{U}}(t, \vec{r}) + \nabla \cdot \vec{\mathcal{F}}(t, \vec{r}) = \vec{0} \quad (\text{A.4})$$

where the flux  $\vec{\mathcal{F}}(t, \vec{r}) = \vec{\alpha}^{-1} \cdot \vec{\mathcal{G}}(t, \vec{r})$ .

As in Section 2.5 we introduce a set of  $M$  unit-vectors  $\vec{m}_0, \vec{m}_1, \dots, \vec{m}_{M-1}$  that span  $\mathbb{R}^3$  such that  $|\vec{m}_0| = |\vec{m}_1| = \dots = |\vec{m}_{M-1}| = 1$ . We allow the set  $\{\vec{m}_k\}$ ,  $k = 0, 1, \dots, M-1$  to be linearly dependent (thus it may not be a minimal spanning set, or basis, for  $\mathbb{R}^3$ ). Any point  $\vec{r} \in \mathbb{R}^3$  can be written as a linear (though not necessarily unique) combination of the set  $\{\vec{m}_k\}$ . Then, for some set of parameters  $(\xi_0, \xi_1, \dots, \xi_{M-1}) \in \mathbb{R}^M$  we

have the following non-unique mapping:

$$(\xi_0, \xi_1, \dots, \xi_{M-1}) \in \mathbb{R}^M \mapsto \vec{r} = \sum_{k=0}^{M-1} \xi_k \vec{m}_k \in \mathbb{R}^3. \quad (\text{A.5})$$

It is clear from the definition of this mapping that the coordinates  $x$ ,  $y$  and  $z$  are functions of the coordinates  $\xi_k$ :

$$\begin{aligned} x &= x(\xi_0, \xi_1, \dots, \xi_{M-1}) \\ y &= y(\xi_0, \xi_1, \dots, \xi_{M-1}) \\ z &= z(\xi_0, \xi_1, \dots, \xi_{M-1}) \end{aligned} \quad (\text{A.6})$$

such that we may write partial derivatives with respect to the coordinates  $x$ ,  $y$  and  $z$  via the chain rule for partial differentiation:

$$\begin{aligned} \frac{\partial}{\partial x} &= \frac{\partial \xi_0}{\partial x} \frac{\partial}{\partial \xi_0} + \frac{\partial \xi_1}{\partial x} \frac{\partial}{\partial \xi_1} + \dots + \frac{\partial \xi_{M-1}}{\partial x} \frac{\partial}{\partial \xi_{M-1}} \\ \frac{\partial}{\partial y} &= \frac{\partial \xi_0}{\partial y} \frac{\partial}{\partial \xi_0} + \frac{\partial \xi_1}{\partial y} \frac{\partial}{\partial \xi_1} + \dots + \frac{\partial \xi_{M-1}}{\partial y} \frac{\partial}{\partial \xi_{M-1}} \\ \frac{\partial}{\partial z} &= \frac{\partial \xi_0}{\partial z} \frac{\partial}{\partial \xi_0} + \frac{\partial \xi_1}{\partial z} \frac{\partial}{\partial \xi_1} + \dots + \frac{\partial \xi_{M-1}}{\partial z} \frac{\partial}{\partial \xi_{M-1}}. \end{aligned}$$

Geometrical considerations can be used to show that:

$$\vec{r} = \sum_{k=0}^{M-1} \xi_k \frac{\partial \vec{r}}{\partial \xi_k}, \quad (\text{A.7})$$

from which we can show that:

$$\frac{\partial \vec{r}}{\partial \xi_k} = \frac{\partial \xi_k}{\partial \vec{r}} = \vec{m}_k. \quad (\text{A.8})$$

Using the partial differentials defined in (A.7) we write the divergence of  $\vec{\mathcal{G}}(t, \vec{r})$  as

$$\begin{aligned} \nabla \cdot \vec{\mathcal{G}}(t, \vec{r}) &= \frac{\partial \vec{\mathcal{G}}_x}{\partial x} + \frac{\partial \vec{\mathcal{G}}_y}{\partial y} + \frac{\partial \vec{\mathcal{G}}_z}{\partial z} \\ &= \sum_{k=0}^{M-1} \left( \frac{\partial \vec{\mathcal{G}}_x(\vec{\mathcal{U}})}{\partial \xi_k} \frac{\partial \xi_k}{\partial x} + \frac{\partial \vec{\mathcal{G}}_y(\vec{\mathcal{U}})}{\partial \xi_k} \frac{\partial \xi_k}{\partial y} + \frac{\partial \vec{\mathcal{G}}_z(\vec{\mathcal{U}})}{\partial \xi_k} \frac{\partial \xi_k}{\partial z} \right) \\ &= \sum_{k=0}^{M-1} \left( \frac{\partial \vec{\mathcal{G}}_x(\vec{\mathcal{U}})}{\partial \xi_k} m_{kx} + \frac{\partial \vec{\mathcal{G}}_y(\vec{\mathcal{U}})}{\partial \xi_k} m_{ky} + \frac{\partial \vec{\mathcal{G}}_z(\vec{\mathcal{U}})}{\partial \xi_k} m_{kz} \right) \end{aligned}$$

Noting that

$$\frac{\partial}{\partial \xi_k} \vec{\mathcal{G}}_x(\vec{\mathcal{U}}(t, \vec{r})) = \frac{\partial \vec{\mathcal{G}}_x}{\partial \vec{\mathcal{U}}} \cdot \frac{\partial \vec{\mathcal{U}}}{\partial \xi_k}, \quad (\text{A.9})$$

allows us to rewrite (A.9) as:

$$\nabla \cdot \vec{\mathcal{G}}(t, \vec{r}) = \sum_{k=0}^{M-1} \left( \frac{\partial \vec{\mathcal{G}}_x(\vec{\mathcal{U}})}{\partial \vec{\mathcal{U}}} m_{kx} + \frac{\partial \vec{\mathcal{G}}_y(\vec{\mathcal{U}})}{\partial \vec{\mathcal{U}}} m_{ky} + \frac{\partial \vec{\mathcal{G}}_z(\vec{\mathcal{U}})}{\partial \vec{\mathcal{U}}} m_{kz} \right) \cdot \frac{\partial \vec{\mathcal{U}}}{\partial \xi_k}. \quad (\text{A.10})$$

The vector derivative  $\partial \vec{\mathcal{G}}_x / \partial \vec{\mathcal{U}}$  is defined in the usual fashion as:

$$\begin{aligned} \frac{\partial \vec{\mathcal{G}}_x}{\partial \vec{\mathcal{U}}} &= \left( \left( \frac{\partial}{\partial \vec{\mathcal{U}}} \right)^T \cdot \vec{\mathcal{G}}_x^T \right)^T \\ &= \begin{bmatrix} \frac{\partial \vec{\mathcal{G}}_{x1}}{\partial \vec{\mathcal{U}}_1} & \frac{\partial \vec{\mathcal{G}}_{x1}}{\partial \vec{\mathcal{U}}_2} & \frac{\partial \vec{\mathcal{G}}_{x1}}{\partial \vec{\mathcal{U}}_3} & \frac{\partial \vec{\mathcal{G}}_{x1}}{\partial \vec{\mathcal{U}}_4} & \frac{\partial \vec{\mathcal{G}}_{x1}}{\partial \vec{\mathcal{U}}_5} & \frac{\partial \vec{\mathcal{G}}_{x1}}{\partial \vec{\mathcal{U}}_6} \\ \frac{\partial \vec{\mathcal{G}}_{x2}}{\partial \vec{\mathcal{U}}_1} & \frac{\partial \vec{\mathcal{G}}_{x2}}{\partial \vec{\mathcal{U}}_2} & \frac{\partial \vec{\mathcal{G}}_{x2}}{\partial \vec{\mathcal{U}}_3} & \frac{\partial \vec{\mathcal{G}}_{x2}}{\partial \vec{\mathcal{U}}_4} & \frac{\partial \vec{\mathcal{G}}_{x2}}{\partial \vec{\mathcal{U}}_5} & \frac{\partial \vec{\mathcal{G}}_{x2}}{\partial \vec{\mathcal{U}}_6} \\ \frac{\partial \vec{\mathcal{G}}_{x3}}{\partial \vec{\mathcal{U}}_1} & \frac{\partial \vec{\mathcal{G}}_{x3}}{\partial \vec{\mathcal{U}}_2} & \frac{\partial \vec{\mathcal{G}}_{x3}}{\partial \vec{\mathcal{U}}_3} & \frac{\partial \vec{\mathcal{G}}_{x3}}{\partial \vec{\mathcal{U}}_4} & \frac{\partial \vec{\mathcal{G}}_{x3}}{\partial \vec{\mathcal{U}}_5} & \frac{\partial \vec{\mathcal{G}}_{x3}}{\partial \vec{\mathcal{U}}_6} \\ \frac{\partial \vec{\mathcal{G}}_{x4}}{\partial \vec{\mathcal{U}}_1} & \frac{\partial \vec{\mathcal{G}}_{x4}}{\partial \vec{\mathcal{U}}_2} & \frac{\partial \vec{\mathcal{G}}_{x4}}{\partial \vec{\mathcal{U}}_3} & \frac{\partial \vec{\mathcal{G}}_{x4}}{\partial \vec{\mathcal{U}}_4} & \frac{\partial \vec{\mathcal{G}}_{x4}}{\partial \vec{\mathcal{U}}_5} & \frac{\partial \vec{\mathcal{G}}_{x4}}{\partial \vec{\mathcal{U}}_6} \\ \frac{\partial \vec{\mathcal{G}}_{x5}}{\partial \vec{\mathcal{U}}_1} & \frac{\partial \vec{\mathcal{G}}_{x5}}{\partial \vec{\mathcal{U}}_2} & \frac{\partial \vec{\mathcal{G}}_{x5}}{\partial \vec{\mathcal{U}}_3} & \frac{\partial \vec{\mathcal{G}}_{x5}}{\partial \vec{\mathcal{U}}_4} & \frac{\partial \vec{\mathcal{G}}_{x5}}{\partial \vec{\mathcal{U}}_5} & \frac{\partial \vec{\mathcal{G}}_{x5}}{\partial \vec{\mathcal{U}}_6} \\ \frac{\partial \vec{\mathcal{G}}_{x6}}{\partial \vec{\mathcal{U}}_1} & \frac{\partial \vec{\mathcal{G}}_{x6}}{\partial \vec{\mathcal{U}}_2} & \frac{\partial \vec{\mathcal{G}}_{x6}}{\partial \vec{\mathcal{U}}_3} & \frac{\partial \vec{\mathcal{G}}_{x6}}{\partial \vec{\mathcal{U}}_4} & \frac{\partial \vec{\mathcal{G}}_{x6}}{\partial \vec{\mathcal{U}}_5} & \frac{\partial \vec{\mathcal{G}}_{x6}}{\partial \vec{\mathcal{U}}_6} \end{bmatrix} \end{aligned}$$

For each of the three components  $\vec{\mathcal{G}}_x$ ,  $\vec{\mathcal{G}}_y$  and  $\vec{\mathcal{G}}_z$ , we can explicitly write the partial derivatives with respect to the generalized vector  $\vec{\mathcal{U}}$  as:

$$\frac{\partial \vec{\mathcal{G}}_x}{\partial \vec{\mathcal{U}}} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{\partial \mathcal{H}_z}{\partial \mathcal{E}_x} & \frac{\partial \mathcal{H}_z}{\partial \mathcal{E}_y} & \frac{\partial \mathcal{H}_z}{\partial \mathcal{E}_z} & \frac{\partial \mathcal{H}_z}{\partial \mathcal{H}_x} & \frac{\partial \mathcal{H}_z}{\partial \mathcal{H}_y} & \frac{\partial \mathcal{H}_z}{\partial \mathcal{H}_z} \\ -\frac{\partial \mathcal{H}_y}{\partial \mathcal{E}_x} & -\frac{\partial \mathcal{H}_y}{\partial \mathcal{E}_y} & -\frac{\partial \mathcal{H}_y}{\partial \mathcal{E}_z} & -\frac{\partial \mathcal{H}_y}{\partial \mathcal{H}_x} & -\frac{\partial \mathcal{H}_y}{\partial \mathcal{H}_y} & -\frac{\partial \mathcal{H}_y}{\partial \mathcal{H}_z} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -\frac{\partial \mathcal{E}_z}{\partial \mathcal{E}_x} & -\frac{\partial \mathcal{E}_z}{\partial \mathcal{E}_y} & -\frac{\partial \mathcal{E}_z}{\partial \mathcal{E}_z} & -\frac{\partial \mathcal{E}_z}{\partial \mathcal{H}_x} & -\frac{\partial \mathcal{E}_z}{\partial \mathcal{H}_y} & -\frac{\partial \mathcal{E}_z}{\partial \mathcal{H}_z} \\ \frac{\partial \mathcal{E}_y}{\partial \mathcal{E}_x} & \frac{\partial \mathcal{E}_y}{\partial \mathcal{E}_y} & \frac{\partial \mathcal{E}_y}{\partial \mathcal{E}_z} & \frac{\partial \mathcal{E}_y}{\partial \mathcal{H}_x} & \frac{\partial \mathcal{E}_y}{\partial \mathcal{H}_y} & \frac{\partial \mathcal{E}_y}{\partial \mathcal{H}_z} \end{bmatrix}$$

$$\frac{\partial \vec{\mathcal{G}}_y}{\partial \vec{\mathcal{U}}} = \begin{bmatrix} -\frac{\partial \mathcal{H}_z}{\partial \mathcal{E}_x} & -\frac{\partial \mathcal{H}_z}{\partial \mathcal{E}_y} & -\frac{\partial \mathcal{H}_z}{\partial \mathcal{E}_z} & -\frac{\partial \mathcal{H}_z}{\partial \mathcal{H}_x} & -\frac{\partial \mathcal{H}_z}{\partial \mathcal{H}_y} & -\frac{\partial \mathcal{H}_z}{\partial \mathcal{H}_z} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{\partial \mathcal{H}_x}{\partial \mathcal{E}_x} & \frac{\partial \mathcal{H}_x}{\partial \mathcal{E}_y} & \frac{\partial \mathcal{H}_x}{\partial \mathcal{E}_z} & \frac{\partial \mathcal{H}_x}{\partial \mathcal{H}_x} & \frac{\partial \mathcal{H}_x}{\partial \mathcal{H}_y} & \frac{\partial \mathcal{H}_x}{\partial \mathcal{H}_z} \\ \frac{\partial \mathcal{E}_z}{\partial \mathcal{E}_x} & \frac{\partial \mathcal{E}_z}{\partial \mathcal{E}_y} & \frac{\partial \mathcal{E}_z}{\partial \mathcal{E}_z} & \frac{\partial \mathcal{E}_z}{\partial \mathcal{H}_x} & \frac{\partial \mathcal{E}_z}{\partial \mathcal{H}_y} & \frac{\partial \mathcal{E}_z}{\partial \mathcal{H}_z} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -\frac{\partial \mathcal{E}_x}{\partial \mathcal{E}_x} & -\frac{\partial \mathcal{E}_x}{\partial \mathcal{E}_y} & -\frac{\partial \mathcal{E}_x}{\partial \mathcal{E}_z} & -\frac{\partial \mathcal{E}_x}{\partial \mathcal{H}_x} & -\frac{\partial \mathcal{E}_x}{\partial \mathcal{H}_y} & -\frac{\partial \mathcal{E}_x}{\partial \mathcal{H}_z} \end{bmatrix}$$

$$\frac{\partial \vec{\mathcal{G}}_z}{\partial \vec{\mathcal{U}}} = \begin{bmatrix} \frac{\partial \mathcal{H}_y}{\partial \mathcal{E}_x} & \frac{\partial \mathcal{H}_y}{\partial \mathcal{E}_y} & \frac{\partial \mathcal{H}_y}{\partial \mathcal{E}_z} & \frac{\partial \mathcal{H}_y}{\partial \mathcal{H}_x} & \frac{\partial \mathcal{H}_y}{\partial \mathcal{H}_y} & \frac{\partial \mathcal{H}_y}{\partial \mathcal{H}_z} \\ -\frac{\partial \mathcal{H}_x}{\partial \mathcal{E}_x} & -\frac{\partial \mathcal{H}_x}{\partial \mathcal{E}_y} & -\frac{\partial \mathcal{H}_x}{\partial \mathcal{E}_z} & -\frac{\partial \mathcal{H}_x}{\partial \mathcal{H}_x} & -\frac{\partial \mathcal{H}_x}{\partial \mathcal{H}_y} & -\frac{\partial \mathcal{H}_x}{\partial \mathcal{H}_z} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -\frac{\partial \mathcal{E}_y}{\partial \mathcal{E}_x} & -\frac{\partial \mathcal{E}_y}{\partial \mathcal{E}_y} & -\frac{\partial \mathcal{E}_y}{\partial \mathcal{E}_z} & -\frac{\partial \mathcal{E}_y}{\partial \mathcal{H}_x} & -\frac{\partial \mathcal{E}_y}{\partial \mathcal{H}_y} & -\frac{\partial \mathcal{E}_y}{\partial \mathcal{H}_z} \\ \frac{\partial \mathcal{E}_x}{\partial \mathcal{E}_x} & \frac{\partial \mathcal{E}_x}{\partial \mathcal{E}_y} & \frac{\partial \mathcal{E}_x}{\partial \mathcal{E}_z} & \frac{\partial \mathcal{E}_x}{\partial \mathcal{H}_x} & \frac{\partial \mathcal{E}_x}{\partial \mathcal{H}_y} & \frac{\partial \mathcal{E}_x}{\partial \mathcal{H}_z} \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Equation (A.1) can be interpreted as a conservation law on a finite-volume with constant material parameters. In this case, Maxwell's equations satisfy the wave equation and the field components become uncoupled. This means that the above partial dif-

ferential matrices may be simplified to:

$$\frac{\partial \vec{\mathcal{G}}_x}{\partial \vec{\mathcal{U}}} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \quad \frac{\partial \vec{\mathcal{G}}_y}{\partial \vec{\mathcal{U}}} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\frac{\partial \vec{\mathcal{G}}_z}{\partial \vec{\mathcal{U}}} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Therefore:

$$\frac{\partial \vec{\mathcal{G}}_x}{\partial \vec{\mathcal{U}}} m_{kx} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & m_{kx} \\ 0 & 0 & 0 & 0 & -m_{kx} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -m_{kx} & 0 & 0 & 0 \\ 0 & m_{kx} & 0 & 0 & 0 & 0 \end{bmatrix}$$



$$\frac{\partial \vec{\mathcal{G}}_y}{\partial \vec{\mathcal{U}}} m_{ky} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & -m_{ky} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & m_{ky} & 0 & 0 \\ 0 & 0 & m_{ky} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -m_{ky} & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\frac{\partial \vec{\mathcal{G}}_z}{\partial \vec{\mathcal{U}}} m_{kz} = \begin{bmatrix} 0 & 0 & 0 & 0 & m_{kz} & 0 \\ 0 & 0 & 0 & -m_{kz} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -m_{kz} & 0 & 0 & 0 & 0 \\ m_{kz} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

Noting that

$$\frac{\partial \vec{\mathcal{U}}}{\partial \xi_k} = \frac{\partial \vec{\mathcal{U}}}{\partial \vec{r}} \cdot \frac{\partial \vec{r}}{\partial \xi_k} = \frac{\partial \vec{\mathcal{U}}}{\partial \vec{r}} \cdot \vec{m}_k \equiv \frac{\partial \vec{\mathcal{U}}}{\partial m_k}$$

and using the previously derived matrices of partial derivatives we re-write (A.10) as

$$\nabla \cdot \vec{\mathcal{G}}(t, \vec{r}) = \sum_{k=0}^{M-1} \bar{\bar{\mathbf{L}}}(\vec{m}_k) \cdot \frac{\partial \vec{\mathcal{U}}(t, \vec{r})}{\partial m_k} \quad (\text{A.11})$$

where the matrix  $\bar{\bar{\mathbf{L}}}(\vec{m}_k) \in \mathbb{R}^6 \times \mathbb{R}^6$  is defined as:

$$\bar{\bar{\mathbf{L}}}(\vec{m}_k) \triangleq \begin{bmatrix} 0 & 0 & 0 & 0 & m_{kz} & -m_{ky} \\ 0 & 0 & 0 & -m_{kz} & 0 & m_{kx} \\ 0 & 0 & 0 & m_{ky} & -m_{kx} & 0 \\ 0 & -m_{kz} & m_{ky} & 0 & 0 & 0 \\ m_{kz} & 0 & -m_{kx} & 0 & 0 & 0 \\ -m_{ky} & m_{kx} & 0 & 0 & 0 & 0 \end{bmatrix} \quad (\text{A.12})$$

allowing us to re-write the conservative form of Maxwell's equation in terms of the selected basis  $\{\vec{m}_k\}$  on the finite-volume  $V_i$  with material parameters  $\bar{\bar{\boldsymbol{\alpha}}}_i$  as

$$\frac{\partial}{\partial t} \vec{\mathbf{u}}(t, \vec{r}) + \sum_{k=0}^{M-1} \bar{\bar{\mathbf{A}}}_i(\vec{m}_k) \cdot \frac{\partial \vec{\mathbf{u}}(t, \vec{r})}{\partial m_k} = \vec{\mathbf{0}} \quad (\text{A.13})$$

where  $\bar{\bar{\mathbf{A}}}_i(\vec{m}_k) = \bar{\bar{\boldsymbol{\alpha}}}_i^{-1} \cdot \bar{\bar{\mathbf{L}}}(\vec{m}_k)$ . This is precisely the form of equation (2.17).

# B

## Time Domain Integral Solutions To Maxwell's Equations

---

In this chapter, integral forms of Maxwell's Equations in the time-domain are developed. The theory primarily follows that presented in D. S. Jones classic text [33].

### *B.1 The Vector and Scalar Potentials*

In the absence of magnetic sources, Maxwell's Equations are:

$$\nabla \times \vec{\mathcal{H}}(t, \vec{r}) = \frac{\partial}{\partial t} \vec{\mathcal{D}}(t, \vec{r}) + \vec{\mathcal{J}}(t, \vec{r}) \quad (\text{B.1a})$$

$$\nabla \times \vec{\mathcal{E}}(t, \vec{r}) = -\frac{\partial}{\partial t} \vec{\mathcal{B}}(t, \vec{r}) \quad (\text{B.1b})$$

$$\nabla \cdot \vec{\mathcal{D}}(t, \vec{r}) = \rho_e(t, \vec{r}) \quad (\text{B.1c})$$

$$\nabla \cdot \vec{\mathcal{B}}(t, \vec{r}) = 0 \quad (\text{B.1d})$$

where the electric conduction current is included in  $\vec{\mathcal{J}}(t, \vec{r})$ . As the magnetic flux  $\vec{\mathcal{B}}(t, \vec{r})$  is source free and can be written as the curl of the vector potential  $\vec{\mathcal{A}}(t, \vec{r})$ :

$$\vec{\mathcal{B}}(t, \vec{r}) = \nabla \times \vec{\mathcal{A}}(t, \vec{r}) \quad (\text{B.2})$$

Substituting this relationship into Faraday's Law (B.1b) relates the electric field to  $\vec{\mathcal{A}}(t, \vec{r})$  as:

$$\nabla \times \vec{\mathcal{E}}(t, \vec{r}) + \nabla \times \frac{\partial}{\partial t} \vec{\mathcal{A}}(t, \vec{r}) + \nabla \times \nabla \phi(t, \vec{r}) = 0$$

where the addition of  $\nabla \times \nabla \phi(t, \vec{r})$  does not contribute to the equality. This equality is zero for all space and time provided that

$$\vec{\mathcal{E}}(t, \vec{r}) = -\frac{\partial}{\partial t} \vec{\mathcal{A}}(t, \vec{r}) - \nabla \phi(t, \vec{r}) \quad (\text{B.3})$$

where  $\phi(t, \vec{r})$  is the scalar potential. The equations (B.2) and (B.3) relate the field quantities  $\vec{\mathcal{B}}(t, \vec{r})$  and  $\vec{\mathcal{E}}(t, \vec{r})$  to the potentials. These relationships are not unique. If we consider the following potentials:

$$\begin{aligned} \vec{\mathcal{A}}_0(t, \vec{r}) &= \vec{\mathcal{A}}(t, \vec{r}) - \nabla \psi(t, \vec{r}) \\ \phi_0(t, \vec{r}) &= \phi(t, \vec{r}) + \frac{\partial}{\partial t} \psi(t, \vec{r}) \end{aligned} \quad (\text{B.4})$$

for some sufficiently differentiable scalar function  $\psi(t, \vec{r})$ , then upon substitution into (B.2) and (B.3) we get:

$$\begin{aligned}
\vec{\mathcal{B}}(t, \vec{r}) &= \nabla \times (\vec{\mathcal{A}}(t, \vec{r}) - \nabla\psi(t, \vec{r})) \\
&= \nabla \times (\vec{\mathcal{A}}(t, \vec{r})) \\
\vec{\mathcal{E}}(t, \vec{r}) &= -\frac{\partial}{\partial t}(\vec{\mathcal{A}}(t, \vec{r}) - \nabla\psi(t, \vec{r})) - \nabla\phi(t, \vec{r}) - \nabla\left(\phi(t, \vec{r}) + \frac{\partial}{\partial t}\psi(t, \vec{r})\right) \quad (\text{B.5}) \\
&= -\frac{\partial}{\partial t}(\vec{\mathcal{A}}(t, \vec{r}) + \frac{\partial}{\partial t}\nabla\psi(t, \vec{r}) - \nabla\phi(t, \vec{r})) - \nabla\frac{\partial}{\partial t}\psi(t, \vec{r}) \\
&= -\frac{\partial}{\partial t}\vec{\mathcal{A}}(t, \vec{r}) - \nabla\phi(t, \vec{r})
\end{aligned}$$

Therefore, the potentials (B.4) also satisfy Maxwell's Equations and we say that Maxwell's Equations are invariant under the *gauge transformation* (B.4).

If we now assume that the magnetic and electric properties of the medium are simple and stationary for all  $\vec{r}$  of interest such that  $\vec{\mathcal{D}}(t, \vec{r}) = \epsilon\vec{\mathcal{E}}(t, \vec{r})$  and  $\vec{\mathcal{B}}(t, \vec{r}) = \mu\vec{\mathcal{H}}(t, \vec{r})$ , then Ampères law (B.1a) can be written in terms of the potentials as:

$$\begin{aligned}
\nabla \times \nabla \times \vec{\mathcal{A}}(t, \vec{r}) &= \mu\epsilon\frac{\partial}{\partial t}\vec{\mathcal{E}}(t, \vec{r}) + \mu\vec{\mathcal{J}}(t, \vec{r}) \\
&= -\mu\epsilon\frac{\partial^2}{\partial t^2}\vec{\mathcal{A}}(t, \vec{r}) - \mu\epsilon\nabla\frac{\partial}{\partial t}\phi(t, \vec{r}) + \mu\vec{\mathcal{J}}(t, \vec{r}) \quad (\text{B.6}) \\
&= -\frac{1}{c^2}\frac{\partial^2}{\partial t^2}\vec{\mathcal{A}}(t, \vec{r}) - \frac{1}{c^2}\nabla\frac{\partial}{\partial t}\phi(t, \vec{r}) + \mu\vec{\mathcal{J}}(t, \vec{r})
\end{aligned}$$

where the velocity  $c^2 = (\mu\epsilon)^{-1}$ . This provides three equations in the four unknown potential quantities. We will obtain a fourth equation shortly. In the meantime, the Lapacian operator in Cartesian coordinates can be written as:

$$\begin{aligned}
\nabla^2\vec{\mathcal{A}}(t, \vec{r}) &= \hat{x}\frac{\partial^2}{\partial x^2}A_x(t, \vec{r}) + \hat{y}\frac{\partial^2}{\partial x^2}A_y(t, \vec{r}) + \hat{z}\frac{\partial^2}{\partial x^2}A_z(t, \vec{r}) \\
&\equiv \nabla(\nabla \cdot \vec{\mathcal{A}}(t, \vec{r})) - \nabla \times \nabla \times \vec{\mathcal{A}}(t, \vec{r}) \quad (\text{B.7})
\end{aligned}$$

Using (B.7) in (B.6) gives:

$$\nabla(\nabla \cdot \vec{\mathcal{A}}(t, \vec{r})) - \nabla^2 \vec{\mathcal{A}}(t, \vec{r}) + \mu\epsilon \left( \frac{\partial^2}{\partial t^2} \vec{\mathcal{A}}(t, \vec{r}) + \nabla \frac{\partial}{\partial t} \phi(t, \vec{r}) \right) = \mu \vec{\mathcal{J}}(t, \vec{r})$$

As we have only specified the curl of  $\vec{\mathcal{A}}(t, \vec{r})$  we are free to choose its divergence. By choosing:

$$\nabla \cdot \vec{\mathcal{A}}(t, \vec{r}) = -\mu\epsilon \frac{\partial}{\partial t} \phi(t, \vec{r}) \quad (\text{B.8})$$

we arrive at:

$$\nabla^2 \vec{\mathcal{A}}(t, \vec{r}) - \mu\epsilon \partial_{tt} \vec{\mathcal{A}}(t, \vec{r}) = -\mu \vec{\mathcal{J}}(t, \vec{r}) \quad (\text{B.9})$$

and we see that the vector potential satisfies the vector wave equation.

We now seek a fourth equation relating the vector and scalar potentials. Substituting (B.3) into Gauss's law gives:

$$-\nabla \cdot \frac{\partial}{\partial t} \vec{\mathcal{A}}(t, \vec{r}) - \nabla^2 \phi(t, \vec{r}) = \frac{\rho_e(\vec{r})}{\epsilon} \quad (\text{B.10})$$

Interchanging the temporal and spatial derivatives of the vector potential and substituting for the divergence of  $\vec{\mathcal{A}}(t, \vec{r})$  from (B.8) we obtain:

$$\nabla^2 \phi(t, \vec{r}) - \frac{1}{c^2} \partial_{tt} \phi(t, \vec{r}) = -\frac{\rho_e(\vec{r})}{\epsilon} \quad (\text{B.11})$$

and so the scalar potential satisfies the scalar wave equation.

In Cartesian coordinates the three components of the vector wave equation for simple, stationary media, (B.9), leaving three equations of the form (B.11). Therefore we seek the solution to (B.11). It can be shown that the causal impulse response of this PDE

due to a point charge at  $(t', \vec{r}')$ , *i.e.*:

$$\nabla^2 G(t, \vec{r}; t', \vec{r}') - \frac{1}{v^2} \partial_{tt} G(t, \vec{r}; t', \vec{r}') = -\delta(t - t') \delta(\vec{r} - \vec{r}') \quad (\text{B.12})$$

is equal to:

$$G(t, \vec{r}; t', \vec{r}') = \frac{\delta(t - t' - |\vec{r} - \vec{r}'|/c)}{4\pi |\vec{r} - \vec{r}'|} \quad (\text{B.13})$$

where  $\delta(\xi)$  is the Dirac delta function.  $G$  is often referred to as the scalar Green's function for homogeneous and stationary media and can be verified by direct substitution into (B.12). The notation “;” should be interpreted as denoting that  $G(t, \vec{r})$  is due to a source at  $(t', \vec{r}')$

If all sources are inside some volume  $V$  then, from the definition of the delta function, we have:

$$\rho(t, \vec{r}) = \iiint_V \int_{-\infty}^{\infty} \rho_e(t', \vec{r}') \delta(t - t') \delta(\vec{r} - \vec{r}') dt' dV'$$

and therefore the solution to the scalar wave equation (B.11) can be written as:

$$\begin{aligned} \phi(t, \vec{r}) &= \frac{1}{\epsilon} \iiint_V \int_{-\infty}^{\infty} G(t, \vec{r}; t', \vec{r}') \rho(t', \vec{r}') dV' \\ &= \frac{1}{4\pi\epsilon} \iiint_V \int_{-\infty}^{\infty} \frac{\delta(t - t' - |\vec{r} - \vec{r}'|/c)}{|\vec{r} - \vec{r}'|} \rho_e(t', \vec{r}') dt' dV' \\ &= \frac{1}{4\pi\epsilon} \iiint_V \frac{\rho_e(t - t' - |\vec{r} - \vec{r}'|/c, \vec{r}')}{|\vec{r} - \vec{r}'|} dV' \\ &= \frac{1}{4\pi\epsilon} \iiint_V \frac{[[\rho_e(t, \vec{r}')] ]}{|\vec{r} - \vec{r}'|} dV' \end{aligned} \quad (\text{B.14})$$

where the notation  $[[\cdot]]$  is used to denote temporal evaluation at the retarded time

$t = t' - |\vec{r} - \vec{r}'|/c$ . In a completely analogous manner for the vector potential we obtain:

$$\vec{\mathcal{A}}(t, \vec{r}) = \frac{\mu}{4\pi} \iiint_V \frac{[[J(t, \vec{r}')] ]}{|\vec{r} - \vec{r}'|} dV' \quad (\text{B.15})$$

where once again all sources lie on the interior of  $V$ .

### B.2 The Kirchoff Solution of the Scalar Wave Equation

For a sufficiently smooth spatial vector field the divergence theorem hold:

$$\iiint_V \nabla' \cdot \vec{a}(\vec{r}') dV' = \iint_S \vec{a}(\vec{r}') \cdot \vec{n}(\vec{r}') dS' \quad (\text{B.16})$$

where  $\vec{n}(\vec{r}')$  is the outward normal to the boundary  $S$  of the volume  $V$ . The notation  $\nabla'$  denotes that the operator is with respect to the primed position  $\vec{r}'$ . We consider the following vector quantities:

$$\begin{aligned} \vec{a}(t', \vec{r}') &= G(t, \vec{r}; t', \vec{r}') \nabla' \phi(t', \vec{r}') \\ \vec{b}(t', \vec{r}') &= \phi(t', \vec{r}') \nabla' G(t, \vec{r}; t', \vec{r}') \end{aligned} \quad (\text{B.17})$$

with divergences:

$$\begin{aligned} \nabla' \cdot \vec{a}(t', \vec{r}') &= \nabla' G(t, \vec{r}; t', \vec{r}') \cdot \nabla' \phi(t', \vec{r}') + G(t, \vec{r}; t', \vec{r}') \nabla'^2 \phi(t', \vec{r}') \\ \nabla' \cdot \vec{b}(t', \vec{r}') &= \nabla' \phi(t', \vec{r}') \cdot \nabla' G(t, \vec{r}; t', \vec{r}') + \phi(t', \vec{r}') \nabla'^2 G(t, \vec{r}; t', \vec{r}') \end{aligned} \quad (\text{B.18})$$



Substituting the difference of these divergences into the divergence theorem (B.16) and integrating with respect to  $t'$  gives:

$$\begin{aligned} & \iiint_V \int_{-\infty}^{\infty} \left( G(t, \vec{r}; t', \vec{r}') \nabla'^2 \phi(t', \vec{r}') - \phi(t', \vec{r}') \nabla'^2 G(t, \vec{r}; t', \vec{r}') \right) dt' dV' = \\ & \iiint_S \int_{-\infty}^{\infty} \left( G(t, \vec{r}; t', \vec{r}') \partial_{n'} \phi(t', \vec{r}') - \phi(t', \vec{r}') \partial_{n'} G(t, \vec{r}; t', \vec{r}') \right) dt' dS' \end{aligned} \quad (\text{B.19})$$

where  $\vec{r}' \in V$  and where  $\partial_{n'} \phi(t', \vec{r}') \triangleq \nabla' \phi(t', \vec{r}') \cdot \vec{n}(\vec{r}')$ . Within this context the space-time source location of for the Green's function is considered variable. The first term in the surface integral is:

$$\begin{aligned} I_1(t, \vec{r}) & \triangleq \iint_S \int_{-\infty}^{\infty} G(t, \vec{r}; t', \vec{r}') \partial_{n'} \phi(t', \vec{r}') dt' dS' \\ & = \frac{1}{4\pi} \iint_S \frac{[[\partial_{n'} \phi(t, \vec{r}')] ]}{|\vec{r} - \vec{r}'|} dS' \end{aligned} \quad (\text{B.20})$$

due to the sifting property of the impulse function.

The second surface integral is the negative of:

$$I_2(t, \vec{r}) \triangleq \iint_S \int_{-\infty}^{\infty} \phi(t', \vec{r}') \partial_{n'} G(t, \vec{r}; t', \vec{r}') dt' dS' \quad (\text{B.21})$$

So we must evaluate the derivative:

$$\begin{aligned} 4\pi \partial_{n'} G(t, \vec{r}; t', \vec{r}') & = \partial_{n'} \frac{\delta(t - t' - |\vec{r} - \vec{r}'|/c)}{|\vec{r} - \vec{r}'|} \\ & = -\frac{1}{c|\vec{r} - \vec{r}'|} \delta'(t - t' - |\vec{r} - \vec{r}'|/c) \partial_{n'} |\vec{r} - \vec{r}'| \\ & \quad + \delta(t - t' - |\vec{r} - \vec{r}'|/c) \partial_{n'} \frac{1}{|\vec{r} - \vec{r}'|} \end{aligned} \quad (\text{B.22})$$

so the integral (B.21) becomes:

$$\begin{aligned}
 I_2(t, \vec{r}) &\triangleq \iint_S \int_{-\infty}^{\infty} \phi(t', \vec{r}') \partial_{n'} G(t, \vec{r}; t', \vec{r}') dt' dS' \\
 &= \frac{1}{4\pi} \iint_S \left( [[\phi(t, \vec{r}')] ] \partial_{n'} \frac{1}{|\vec{r} - \vec{r}'|} - \frac{1}{c|\vec{r} - \vec{r}'|} [[\partial_t \phi(t, \vec{r}')] ] \partial_{n'} |\vec{r} - \vec{r}'| \right) dS'
 \end{aligned} \tag{B.23}$$

and the surface integral in (B.19) can now be written as:

$$\begin{aligned}
 I_S(t, \vec{r}) &\triangleq I_1(t, \vec{r}) - I_2(t, \vec{r}) \\
 &= \frac{1}{4\pi} \iint_S \left( \frac{[[\partial_{n'} \phi(t, \vec{r}')] ]}{|\vec{r} - \vec{r}'|} - [[\phi(t, \vec{r}')] ] \partial_{n'} \frac{1}{|\vec{r} - \vec{r}'|} + \frac{1}{c|\vec{r} - \vec{r}'|} [[\partial_t \phi(t, \vec{r}')] ] \partial_{n'} |\vec{r} - \vec{r}'| \right) dS'
 \end{aligned} \tag{B.24}$$

We now seek to simplify the volumetric integral in (B.19). From the definition of the Green's function (B.12) it can be shown by direct evaluation that

$$\nabla'^2 G(t, \vec{r}; t', \vec{r}') = \frac{1}{c^2} \partial_{t't'} G(t, \vec{r}; t', \vec{r}') - \delta(t - t') \delta(\vec{r} - \vec{r}')$$

while from the definition of the scalar potential we have:

$$\nabla'^2 \phi(t', \vec{r}') = \frac{1}{c^2} \partial_{t't'} \phi(t', \vec{r}') - \frac{\rho_e(t', \vec{r}')}{\epsilon}$$

and the volumetric part of (B.19)

$$\begin{aligned}
 I_V(t, \vec{r}) &\triangleq \iiint_V \int_{-\infty}^{\infty} \left( G(t, \vec{r}; t', \vec{r}') \nabla'^2 \phi(t', \vec{r}') - \phi(t', \vec{r}') \nabla'^2 G(t, \vec{r}; t', \vec{r}') \right)' dt' dV' \\
 &= \iiint_V \int_{-\infty}^{\infty} G(t, \vec{r}; t', \vec{r}') \left( \frac{1}{c^2} \partial_{t't'} \phi(t', \vec{r}') - \frac{\rho_e(t', \vec{r}')}{\epsilon} \right) - \\
 &\quad \iiint_V \int_{-\infty}^{\infty} \phi(t', \vec{r}') \left( \frac{1}{c^2} \partial_{t't'} G(t, \vec{r}; t', \vec{r}') - \delta(t-t') \delta(\vec{r}-\vec{r}') \right) dt' dV'
 \end{aligned}$$

This temporal integration above can be evaluated directly by noting that due to the derivative and sifting properties of the delta function the second order time derivatives cancel in the evaluation of the integral and therefore:

$$I_V(t, \vec{r}) = \phi(t, \vec{r}) - \frac{1}{4\pi\epsilon} \iiint_V \frac{[[\rho_e(t, \vec{r}')] ]}{|\vec{r} - \vec{r}'|} dV' \quad (\text{B.25})$$

where, finally, equating  $I_V(t, \vec{r})$  and  $I_S(t, \vec{r})$  from equations (B.25) and (B.24) according to (B.19) gives:

$$\begin{aligned}
 \phi(t, \vec{r}) &= \frac{1}{4\pi\epsilon} \iiint_V \frac{[[\rho_e(t, \vec{r}')] ]}{|\vec{r} - \vec{r}'|} dV' + \\
 &\quad \frac{1}{4\pi} \iint_S \left( \frac{[[\partial_{n'} \phi(t, \vec{r}')] ]}{|\vec{r} - \vec{r}'|} - [[\phi(t, \vec{r}')] ] \partial_{n'} \frac{1}{|\vec{r} - \vec{r}'|} + \frac{1}{v|\vec{r} - \vec{r}'|} [[\partial_t \phi(t, \vec{r}')] ] \partial_{n'} |\vec{r} - \vec{r}'| \right) dS'
 \end{aligned} \quad (\text{B.26})$$

and we arrive at an integral formula for the scalar potential. The physical interpretation of this equation is as follows: charges inside the volume  $V$  contribute to the scalar potential from the volume integral as the scalar potential for a domain  $V$  containing all sources internal to  $V$  given in in (B.14) is identical to the volume integral in (B.26). Therefore, the surface integral must be the contribution to the potential due to sources

outside of  $V$ .

### B.3 Time-Domain Integral Field Solutions

The fact that the electric field  $\vec{\mathcal{E}}(t, \vec{r})$  satisfies the vector wave equation in simple, stationary media, can easily be deduced from Maxwell's Equations (B.1a)-(B.1d) by taking the curl of Faraday's law (B.1b):

$$\begin{aligned}
 \nabla \times \nabla \times \vec{\mathcal{E}}(t, \vec{r}) &= -\partial_t \left( \nabla \times \vec{\mathcal{B}}(t, \vec{r}) \right) \\
 &= -\mu \partial_t \left( \nabla \times \vec{\mathcal{H}}(t, \vec{r}) \right) \\
 &= -\mu \partial_t \left( \epsilon \partial_t \vec{\mathcal{E}}(t, \vec{r}) + \vec{\mathcal{J}}(t, \vec{r}) \right) \\
 &= -\mu \epsilon \partial_{tt} \vec{\mathcal{E}}(t, \vec{r}) - \mu \partial_t \vec{\mathcal{J}}(t, \vec{r})
 \end{aligned} \tag{B.27}$$

and applying the Cartesian definition of the Laplacian we have:

$$\nabla(\nabla \cdot \vec{\mathcal{E}}(t, \vec{r})) - \nabla^2 \vec{\mathcal{E}}(t, \vec{r}) = -\mu \epsilon \partial_{tt} \vec{\mathcal{E}}(t, \vec{r}) - \mu \partial_t \vec{\mathcal{J}}(t, \vec{r})$$

so by re-arranging and applying Gauss's Law we arrive at:

$$\nabla^2 \vec{\mathcal{E}}(t, \vec{r}) - \frac{1}{c^2} \partial_{tt} \vec{\mathcal{E}}(t, \vec{r}) = \frac{\nabla \rho_e(t, \vec{r})}{\epsilon} + \mu \partial_t \vec{\mathcal{J}}(t, \vec{r}) \tag{B.28}$$

Because the three components of the Laplacian are independent the solution for each of the vector components is independently of the form (B.26) (where the source term

is modified accordingly) giving:

$$\begin{aligned}
\vec{\mathcal{E}}(t, \vec{r}) &= -\frac{1}{4\pi} \iiint_V \frac{[[\epsilon^{-1} \nabla' \rho_e(t, \vec{r}') + \mu \partial_t \vec{\mathcal{J}}(t, \vec{r}')]]}{|\vec{r} - \vec{r}'|} dV' + \\
&\frac{1}{4\pi} \iint_S \left( \frac{[[\partial_{n'} \vec{\mathcal{E}}(t, \vec{r}')]]}{|\vec{r} - \vec{r}'|} - [[\vec{\mathcal{E}}(t, \vec{r}')]] \partial_{n'} \frac{1}{|\vec{r} - \vec{r}'|} + \frac{1}{c|\vec{r} - \vec{r}'|} [[\partial_t \vec{\mathcal{E}}(t, \vec{r}')]] \partial_{n'} |\vec{r} - \vec{r}'| \right) dS' \\
&= -\frac{1}{4\pi} \iiint_V \frac{[[\epsilon^{-1} \nabla' \rho_e(t, \vec{r}') + \mu \partial_t \vec{\mathcal{J}}(t, \vec{r}')]]}{|\vec{r} - \vec{r}'|} dV' + \\
&\frac{1}{4\pi} \iint_S \left( \frac{(\vec{n}' \cdot \nabla') [[\vec{\mathcal{E}}(t, \vec{r}')]]}{|\vec{r} - \vec{r}'|} - [[\vec{\mathcal{E}}(t, \vec{r}')]] \partial_{n'} \frac{1}{|\vec{r} - \vec{r}'|} + \frac{2c^{-1}}{|\vec{r} - \vec{r}'|} [[\partial_t \vec{\mathcal{E}}(t, \vec{r}')]] \partial_{n'} |\vec{r} - \vec{r}'| \right) dS'
\end{aligned} \tag{B.29}$$

by means of the identity:

$$\begin{aligned}
[[\partial_{n'} \vec{\mathcal{E}}(t, \vec{r}')]] &= [[(\vec{n}' \cdot \nabla') \vec{\mathcal{E}}(t, \vec{r}')]] \\
&= (\vec{n}' \cdot \nabla') [[\vec{\mathcal{E}}(t, \vec{r}')]] + \frac{1}{c} [[\partial_t \vec{\mathcal{E}}(t, \vec{r}')]] (\vec{n}' \cdot \nabla') |\vec{r} - \vec{r}'|
\end{aligned}$$

where the last equality can be shown to hold by summing the differentials of each argument of the field. The surface integral then involves a term of the form:

$$\iint_S \frac{(\vec{n}' \cdot \nabla') [[\vec{\mathcal{E}}(t, \vec{r}')]]}{|\vec{r} - \vec{r}'|} dS'$$

As the surface is closed, for arbitrary vector  $\vec{a}(\vec{r}')$  we have:

$$\iint_S ((\vec{n}' \cdot \nabla') \vec{a}(\vec{r}') + \vec{n}' \times \nabla' \times \vec{a}(\vec{r}') - \vec{n}' \nabla' \cdot \vec{a}(\vec{r}')) dS = 0$$

and so we let  $\vec{a}(t, \vec{r}') = [[\vec{\mathcal{E}}(t, \vec{r}')] ] / |\vec{r} - \vec{r}'|$  to get:

$$\iint_S \left( \frac{1}{R} (\vec{n}' \cdot \nabla') [[\vec{\mathcal{E}}]] + \left( \vec{n}' \times \nabla' \frac{1}{R} \right) \times [[\vec{\mathcal{E}}]] + \frac{1}{R} \vec{n}' \times \nabla' \times [[\vec{\mathcal{E}}]] - \frac{1}{R} \vec{n}' \nabla' \cdot [[\vec{\mathcal{E}}]] \right) dS' = 0 \quad (\text{B.30})$$

where  $\vec{\mathcal{E}} = \vec{\mathcal{E}}(t, \vec{r}')$  and where  $R = |\vec{r} - \vec{r}'|$ . This provides an expression for the surface integral of  $(\vec{n}' \cdot \nabla') [[\vec{\mathcal{E}}]] / R$  which can be substituted into (B.29) to give the following for the surface integral contribution to (B.29) (the first equality is simply the surface integral in (B.29) repeated).

$$\begin{aligned} I_{ES}(t, \vec{r}) &\triangleq \iint_S \left( \frac{(\vec{n}' \cdot \nabla') [[\vec{\mathcal{E}}]]}{R} - [[\vec{\mathcal{E}}]] \partial_{n'} \frac{1}{|\vec{r} - \vec{r}'|} + \frac{2}{cR} [[\partial_t \vec{\mathcal{E}}]] \partial_{n'} R \right) dS' \\ &= \iint_S \left( \frac{1}{R} \vec{n}' \nabla' \cdot [[\vec{\mathcal{E}}]] \right. \\ &\quad \left. - \left( \vec{n}' \times \nabla' \frac{1}{R} \right) \times [[\vec{\mathcal{E}}]] - \frac{1}{R} \vec{n}' \times \nabla' \times [[\vec{\mathcal{E}}]] - [[\vec{\mathcal{E}}]] \partial_{n'} \frac{1}{R} + \frac{2}{cR} [[\partial_t \vec{\mathcal{E}}]] \partial_{n'} R \right) dS' \end{aligned} \quad (\text{B.31})$$

We now use the two following differential/time-retardation relationships:

$$\begin{aligned} \nabla' \cdot [[\vec{\mathcal{E}}]] &= [[\nabla' \cdot \vec{\mathcal{E}}]] - \frac{1}{c} [[\partial_t \vec{\mathcal{E}}]] \cdot \nabla' R \\ \nabla' \times [[\vec{\mathcal{E}}]] &= [[\nabla' \times \vec{\mathcal{E}}]] - \frac{1}{c} \nabla' R \times [[\partial_t \vec{\mathcal{E}}]] \end{aligned} \quad (\text{B.32})$$

which can be obtained by adding the contribution of the spatial differentials of each argument of  $[[\vec{\mathcal{E}}]]$ . Therefore  $I_{ES}$  can be written as:

$$\begin{aligned}
 I_{ES}(t, \vec{r}) = & \iint_S \left( \frac{\vec{n}'}{R} \nabla' \cdot [[\vec{\mathcal{E}}]] \right. \\
 & - \left( \vec{n}' \times \nabla' \frac{1}{R} \right) \times [[\vec{\mathcal{E}}]] - \frac{1}{R} \vec{n}' \times \nabla' \times [[\vec{\mathcal{E}}]] - [[\vec{\mathcal{E}}]] \partial_{n'} \frac{1}{R} + \frac{2}{cR} [[\partial_t \vec{\mathcal{E}}]] \partial_{n'} R \Big) dS' = \\
 & \iint_S \left( \frac{\vec{n}'}{R} [[\nabla' \cdot \vec{\mathcal{E}}]] - \frac{\vec{n}'}{cR} ([[ \partial_t \vec{\mathcal{E}} ]] \cdot \nabla' R) - \left( \vec{n}' \times \nabla' \frac{1}{R} \right) \times [[\vec{\mathcal{E}}]] \right. \\
 & \quad \left. - \frac{1}{R} \vec{n}' \times [[\nabla' \times \vec{\mathcal{E}}]] + \vec{n}' \times \left( \frac{1}{cR} \nabla' R \times [[\partial_t \vec{\mathcal{E}}]] \right) \right. \\
 & \quad \left. - [[\vec{\mathcal{E}}]] \left( \vec{n}' \cdot \nabla' \frac{1}{R} \right) + \frac{2}{cR} [[\partial_t \vec{\mathcal{E}}]] (\vec{n}' \cdot \nabla' R) \right) dS'
 \end{aligned} \tag{B.33}$$

By the vector triple product:

$$\begin{aligned}
 - \left( \vec{n}' \times [[\vec{\mathcal{E}}]] \right) \times \nabla' \frac{1}{R} &= \nabla' \frac{1}{R} \times \left( \vec{n}' \times [[\vec{\mathcal{E}}]] \right) \\
 &= \left( \nabla' \frac{1}{R} \cdot [[\vec{\mathcal{E}}]] \right) \vec{n} - \left( \nabla' \frac{1}{R} \cdot \vec{n} \right) [[\vec{\mathcal{E}}]]
 \end{aligned} \tag{B.34}$$

while

$$\begin{aligned}
 - \left( \vec{n}' \times \nabla' \frac{1}{R} \right) \times [[\vec{\mathcal{E}}]] &= [[\vec{\mathcal{E}}]] \times \left( \vec{n}' \times \nabla' \frac{1}{R} \right) \\
 &= \left( [[\vec{\mathcal{E}}]] \cdot \nabla' \frac{1}{R} \right) \vec{n} - \left( [[\vec{\mathcal{E}}]] \cdot \vec{n} \right) \nabla' \frac{1}{R} \\
 &= - \left( \vec{n}' \times [[\vec{\mathcal{E}}]] \right) \times \nabla' \frac{1}{R} + \left( \nabla' \frac{1}{R} \cdot \vec{n} \right) [[\vec{\mathcal{E}}]] - \left( [[\vec{\mathcal{E}}]] \cdot \vec{n} \right) \nabla' \frac{1}{R}
 \end{aligned} \tag{B.35}$$

Substituting this last result into (B.33) gives:

$$\begin{aligned}
I_{ES}(t, \vec{r}) = & \iint_S \left( \frac{\vec{n}'}{R} [[\nabla' \cdot \vec{\mathcal{E}}]] - \frac{\vec{n}'}{Rc} \left( [[\partial_t \vec{\mathcal{E}}]] \cdot \nabla' R \right) - \left( \vec{n}' \times [[\vec{\mathcal{E}}]] \right) \times \nabla' \frac{1}{R} \right. \\
& - \left( [[\vec{\mathcal{E}}]] \cdot \vec{n} \right) \nabla' \frac{1}{R} - \frac{1}{R} \vec{n}' \times [[\nabla' \times \vec{\mathcal{E}}]] + \vec{n}' \times \left( \frac{1}{cR} \nabla' R \times [[\partial_t \vec{\mathcal{E}}]] \right) \\
& \left. + \frac{2}{cR} [[\partial_t \vec{\mathcal{E}}]] (\vec{n}' \cdot \nabla' R) \right) dS' \tag{B.36}
\end{aligned}$$

Similarly the triple product

$$\vec{n}' \times \left( \frac{1}{cR} \nabla' R \times [[\partial_t \vec{\mathcal{E}}]] \right) = \left( \vec{n}' \cdot [[\partial_t \vec{\mathcal{E}}]] \right) \frac{1}{cR} \nabla' R - \left( \frac{1}{cR} \vec{n}' \cdot \nabla' R \right) [[\partial_t \vec{\mathcal{E}}]] \tag{B.37}$$

can be substituted into (B.36) to give:

$$\begin{aligned}
I_{ES}(t, \vec{r}) = & \iint_S \left( \frac{\vec{n}'}{R} [[\nabla' \cdot \vec{\mathcal{E}}]] - \frac{\vec{n}'}{Rc} \left( [[\partial_t \vec{\mathcal{E}}]] \cdot \nabla' R \right) - \left( \vec{n}' \times [[\vec{\mathcal{E}}]] \right) \times \nabla' \frac{1}{R} \right. \\
& - \left( [[\vec{\mathcal{E}}]] \cdot \vec{n} \right) \nabla' \frac{1}{R} - \frac{1}{R} \vec{n}' \times [[\nabla' \times \vec{\mathcal{E}}]] + \left( \vec{n}' \cdot [[\partial_t \vec{\mathcal{E}}]] \right) \frac{1}{cR} \nabla' R \\
& \left. + \frac{1}{cR} [[\partial_t \vec{\mathcal{E}}]] (\vec{n}' \cdot \nabla' R) \right) dS' \tag{B.38}
\end{aligned}$$

A final triple product:

$$\frac{1}{cR} \nabla' R \times \left( \vec{n}' \times [[\partial_t \vec{\mathcal{E}}]] \right) = \frac{\vec{n}'}{Rc} \left( [[\partial_t \vec{\mathcal{E}}]] \cdot \nabla' R \right) - \left( \frac{1}{cR} \vec{n}' \cdot \nabla' R \right) [[\partial_t \vec{\mathcal{E}}]] \tag{B.39}$$

simplifies (B.38) further to:

$$\begin{aligned}
I_{ES}(t, \vec{r}) = & \iint_S \left( \frac{\vec{n}'}{R} [[\nabla' \cdot \vec{\mathcal{E}}]] + \frac{1}{cR} \left( \vec{n}' \times [[\partial_t \vec{\mathcal{E}}]] \right) \times \nabla' R - \left( \vec{n}' \times [[\vec{\mathcal{E}}]] \right) \times \nabla' \frac{1}{R} \right. \\
& \left. - \left( [[\vec{\mathcal{E}}]] \cdot \vec{n} \right) \nabla' \frac{1}{R} - \frac{1}{R} \vec{n}' \times [[\nabla' \times \vec{\mathcal{E}}]] + \left( \vec{n}' \cdot [[\partial_t \vec{\mathcal{E}}]] \right) \frac{1}{cR} \nabla' R \right) dS' \tag{B.40}
\end{aligned}$$



The term involving the curl of  $\vec{\mathcal{E}}$  can be re-written in terms of the magnetic field from Faraday's law:

$$\begin{aligned}
\vec{\mathcal{E}}(t, \vec{r}) = & -\frac{1}{4\pi} \iiint_V \frac{[[\epsilon^{-1}\nabla'\rho_e(t, \vec{r}') + \mu\partial_t\vec{\mathcal{J}}(t, \vec{r}')] ]}{R} dV' + \frac{1}{4\pi} \iint_S \frac{\vec{n}'}{R} [[\nabla' \cdot \vec{\mathcal{E}}]] dS' \\
& + \frac{1}{4\pi} \iint_S \left( \frac{1}{cR} (\vec{n}' \times [[\partial_t\vec{\mathcal{E}}]]) \times \nabla'R - (\vec{n}' \times [[\vec{\mathcal{E}}]]) \times \nabla'\frac{1}{R} - ([[ \vec{\mathcal{E}}]] \cdot \vec{n}) \nabla'\frac{1}{R} \right. \\
& \left. + \frac{\mu}{R} \vec{n}' \times [[\partial_t\vec{\mathcal{H}}]] + (\vec{n}' \cdot [[\partial_t\vec{\mathcal{E}}]]) \frac{1}{cR} \nabla'R \right) dS'
\end{aligned} \tag{B.41}$$

The divergence of  $\vec{\mathcal{E}}$  can be re-written as a volume integral using Gauss's Law and the gradient theorem such that:

$$\begin{aligned}
\frac{1}{4\pi} \iint_S \frac{\vec{n}'}{R} [[\nabla' \cdot \vec{\mathcal{E}}]] dS' &= \frac{1}{4\pi\epsilon} \iint_S \frac{\vec{n}'}{R} [[\rho_e]] dS' \\
&= \frac{1}{4\pi\epsilon} \iiint_V \frac{1}{R} \nabla' [[\rho_e]] dV'
\end{aligned}$$

And we have:

$$\begin{aligned}
\vec{\mathcal{E}}(t, \vec{r}) = & -\frac{1}{4\pi} \iiint_V \frac{[[\epsilon^{-1}\nabla'\rho_e(t, \vec{r}')]] + [[\mu\partial_t\vec{\mathcal{J}}(t, \vec{r}')] ] - \epsilon^{-1}\nabla'[[\rho_e]]}{R} dV' \\
& + \frac{1}{4\pi} \iint_S \left( \frac{1}{cR} (\vec{n}' \times [[\partial_t\vec{\mathcal{E}}]]) \times \nabla'R - (\vec{n}' \times [[\vec{\mathcal{E}}]]) \times \nabla'\frac{1}{R} - ([[ \vec{\mathcal{E}}]] \cdot \vec{n}) \nabla'\frac{1}{R} \right. \\
& \left. + \frac{\mu}{R} \vec{n}' \times [[\partial_t\vec{\mathcal{H}}]] + (\vec{n}' \cdot [[\partial_t\vec{\mathcal{E}}]]) \frac{1}{cR} \nabla'R \right) dS'
\end{aligned} \tag{B.42}$$

While further simplification of the volumetric integral is possible we will omit it here.

Instead we evaluate the  $R$  derivatives by using the following equalities:

$$\nabla' R = -\frac{\vec{R}}{R} \quad \nabla' \frac{1}{R} = \frac{\vec{R}}{R^3} \quad (\text{B.43})$$

where  $\vec{R} \triangleq |\vec{r} - \vec{r}'|$  equation (B.33) we arrive finally at:

$$\begin{aligned} \vec{\mathcal{E}}(t, \vec{r}) = & -\frac{1}{4\pi} \iiint_V \frac{[[\epsilon^{-1} \nabla' \rho_e(t, \vec{r}')] + [[\mu \partial_t \vec{\mathcal{J}}(t, \vec{r}')] - \epsilon^{-1} \nabla' [[\rho_e]]] dV'}{R} \\ & + \frac{1}{4\pi} \iint_S \left( -\frac{1}{cR} (\vec{n}' \times [[\partial_t \vec{\mathcal{E}}]]) \times \frac{\vec{R}}{R} - (\vec{n}' \times [[\vec{\mathcal{E}}]]) \times \frac{\vec{R}}{R^3} - ([[ \vec{\mathcal{E}} ]]) \cdot \vec{n} \right) \frac{\vec{R}}{R^3} \\ & + \frac{\mu}{R} \vec{n}' \times [[\partial_t \vec{\mathcal{H}}]] - \left( \vec{n}' \cdot [[\partial_t \vec{\mathcal{E}}]] \right) \frac{1}{cR} \frac{\vec{R}}{R} \Big) dS' \end{aligned} \quad (\text{B.44})$$

Collecting like terms we get:

$$\begin{aligned} \vec{\mathcal{E}}(t, \vec{r}) = & -\frac{1}{4\pi} \iiint_V \frac{[[\epsilon^{-1} \nabla' \rho_e(t, \vec{r}')] + [[\mu \partial_t \vec{\mathcal{J}}(t, \vec{r}')] - \epsilon^{-1} \nabla' [[\rho_e]]] dV'}{R} \\ & + \frac{1}{4\pi} \iint_S \left( -\vec{n}' \times \left( [[\vec{\mathcal{E}}]] + c^{-1} R [[\partial_t \vec{\mathcal{E}}]] \right) \times \frac{\vec{R}}{R^3} \right. \\ & \left. - \left( \vec{n}' \cdot \left( [[\vec{\mathcal{E}}]] + c^{-1} R [[\partial_t \vec{\mathcal{E}}]] \right) \right) \frac{\vec{R}}{R^3} + \frac{\mu}{R} \vec{n}' \times [[\partial_t \vec{\mathcal{H}}]] \right) dS' \end{aligned} \quad (\text{B.45})$$

Finally, if we have no sources inside  $V$  then the electric field inside  $V$  is completely specified by the surface fields:

$$\begin{aligned} \vec{\mathcal{E}}(t, \vec{r}) = & \\ & + \frac{1}{4\pi} \iint_S \left( -\vec{n}' \times \left( [[\vec{\mathcal{E}} + c^{-1} R \partial_t \vec{\mathcal{E}}]] \right) \times \frac{\vec{R}}{R^3} \right. \\ & \left. - \left( \vec{n}' \cdot \left( [[\vec{\mathcal{E}} + c^{-1} R \partial_t \vec{\mathcal{E}}]] \right) \right) \frac{\vec{R}}{R^3} + \frac{\mu}{R} \vec{n}' \times [[\partial_t \vec{\mathcal{H}}]] \right) dS' \end{aligned} \quad (\text{B.46})$$

A corresponding integral solution to Maxwell's Equations can be obtained for  $\vec{\mathcal{H}}(t, \vec{r})$  when  $V$  is source-free by duality:

$$\begin{aligned}
 \vec{\mathcal{H}}(t, \vec{r}) = & \\
 & + \frac{1}{4\pi} \iint_S \left( -\vec{n}' \times \left( [[\vec{\mathcal{H}} + c^{-1}R\partial_t\vec{\mathcal{H}}]] \right) \times \frac{\vec{R}}{R^3} \right. \\
 & \left. - \left( \vec{n} \cdot \left( [[\vec{\mathcal{H}} + c^{-1}R\partial_t\vec{\mathcal{E}}]] \right) \right) \frac{\vec{R}}{R^3} - \frac{\epsilon}{R} \vec{n}' \times [[\partial_t\vec{\mathcal{E}}]] \right) dS'
 \end{aligned} \tag{B.47}$$

# Bibliography

---

- [1] J. C. Maxwell, *A treatise on electricity and magnetism / by James Clerk Maxwell*. Dover Publications, New York :, 1954.
- [2] ———, *A dynamical theory of the electromagnetic field [microform] / by J. Clerk Maxwell ; read December 8, 1864*. The Society, [London :, 1865.
- [3] J. S. W. Chew, E. Michielssen and J. Jin, Eds., *Fast and Efficient Algorithms in Computational Electromagnetics*. Norwood, MA, USA: Artech House, 2001.
- [4] C. Hirsch, *Numerical Computation of Internal and External Flows - Volume 1: Fundamentals of Numerical Discretization*. New York: John Wiley & Sons, 1992.
- [5] K. 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] G. Rodrigue and D. White, “A vector finite element time-domain method for solving Maxwell’s equations on unstructured hexahedral grids,” *SIAM Journal on Scientific Computing*, vol. 23, no. 3, pp. 683–706, 2002.
- [7] C. Christopoulos, “The transmission-line modeling method: TLM,” *Antennas and Propagation Magazine, IEEE*, vol. 39, no. 1, p. 90, 2002.
- [8] M. Clemens and T. Weiland, “Discrete electromagnetism with the finite integration technique,” *Progress In Electromagnetics Research, PIER*, vol. 32, pp. 65–87, 2001.
- [9] P. Bonnet, X. Ferrières, B. Michielsen, and P. Klotz, “Finite-volume time-domain method,” in *Time Domain Electromagnetics*, S. M. Rao, Ed. New York: Academic, 1997, pp. 307–367, ch. 9.

- 
- [10] J. S. Hesthaven and T. Warburton, *Nodal Discontinuous Galerkin Methods, Algorithms, Analysis, and Applications*, ser. Text in Applied Mathematics. New York: Springer, 2008, vol. 54.
- [11] R. J. LeVeque, *Finite Volume Methods for Hyperbolic Problems*. Cambridge, U.K.: Cambridge University Press, 2002.
- [12] V. Shankar, A. Mohammadian, and W. Hall, "A time-domain, finite-volume treatment for the Maxwell equations," *Electromagnetics*, vol. 10, no. 1, pp. 127–145, 1990.
- [13] D. Baumann, "A 3-D numerical field solver based on the finite-volume time-domain method," Ph.D. dissertation, Swiss Federal Institute of Technology Zurich, 2006.
- [14] J. LoVetri and D. Firsov, "The University of Manitoba Finite-Volume Time-Domain Engine," Defence Research and Development, Ottawa, Canada, Technical Report, 2007.
- [15] C. Fumeaux, D. Baumann, P. Leuchtman, and R. Vahldieck, "A generalized local time-step scheme for efficient FVTD simulations in strongly inhomogeneous meshes," *IEEE Transactions on Microwave Theory and Techniques*, vol. 52, no. 3, pp. 1067–1076, March 2004.
- [16] S. Piperno, M. Remaki, and L. Fezoui, "A nondiffusive finite volume scheme for the three-dimensional Maxwell's equations on unstructured meshes," *SIAM J. Numer. Anal.*, vol. 39, no. 6, pp. 2089–2108, 2002.
- [17] K. Krohne, D. Baumann, C. Fumeaux, E. Li, and R. Vahldieck, "Frequency-domain finite-volume simulations," in *Microwave Conference, 2007. European*. IEEE, 2007, pp. 158–161.
- [18] F. Edelvik, "A new technique for accurate and stable modeling of arbitrarily oriented thin wires in the FDTD method," *IEEE Transactions on Electromagnetic Compatibility*, vol. 45, no. 2, pp. 416–423, May 2003.
- [19] F. Edelvik, G. Ledfelt, P. Lötstedt, and D. J. Riley, "An unconditionally stable subcell model for arbitrarily oriented thin wires in the FETD method," *IEEE Transactions on Antennas and Propagation*, vol. 51, no. 8, pp. 1797–1805, August 2003.
- [20] Y. Zhang and C. Shu, "Third order WENO scheme on three dimensional tetrahedral meshes," *Communications in Computational Physics*, vol. 5, pp. 836–848, 2009.
- [21] C. Bommaraju, "Investigating Finite Volume Time Domain Methods in Computational Electromagnetics," 2009.

- [22] D. Firsov and J. LoVetri, "FVTD-integral equation hybrid for Maxwell's Equations," *Int*, vol. 21, no. 1-2, pp. 29–42, January-April 2008.
- [23] J. Stratton, *Electromagnetic theory*. Wiley-IEEE Press, 2007.
- [24] B. Van Leer, "Flux-vector splitting for the euler equations," *Lecture Notes in Physics*, vol. 170, pp. 507–512, 1982.
- [25] —, "Towards the ultimate conservative difference scheme, v. a second order sequel to godunov's method," *Journal of Computational Physics*, vol. 32, pp. 101–136, 1979.
- [26] D. Firsov, J. LoVetri, I. Jeffrey, V. Okhmatovski, and C. G. W. Chamma, "High-order FVTD on unstructured grids using an object-oriented finite-volume time-domain computational engine," *ACES Journal*, vol. 22, no. 1, pp. 71–82, March 2007.
- [27] G. Strang, *Introduction to applied mathematics*. Wellesley-Cambridge Press, 1986.
- [28] G. Dahlquist and A. Bjorck, *Numerical Methods*. Dover Publications, 2003.
- [29] S. Piperno, "L2-stability of the upwind first order finite volume scheme for the Maxwell equations in two and three dimensions on arbitrary unstructured meshes," *Mathematical Modelling and Numerical Analysis*, vol. 34, no. 1, pp. 139–158, 2000.
- [30] D. Firsov and J. LoVetri, "New stability criterion for unstructured mesh upwinding FVTD schemes for Maxwell's equations," *ACES Journal*, vol. 23, no. 3, pp. 193–199, September 2008.
- [31] A. Sommerfeld, *Partial differential equations in physics*. Academic Press, 1964.
- [32] B. Shanker, M. Lu, A. Ergin, and E. Michielssen, "Plane-wave time-domain accelerated radiation boundary kernels for FDTD analysis of 3-D electromagnetic phenomena," *Antennas and Propagation, IEEE Transactions on*, vol. 53, no. 11, pp. 3704–3716, 2005.
- [33] D. Jones, *The theory of electromagnetism*. Macmillan New York, 1964.
- [34] K. Sankaran, C. Fumeaux, and R. Vahldieck, "Cell-centered finite-volume-based perfectly matched Layer for time-domain Maxwell system," *Microwave Theory and Techniques, IEEE Transactions on*, vol. 54, no. 3, pp. 1269–1276, 2006.
- [35] R. Harrington, *Time-harmonic electromagnetic fields*, ser. IEEE Press series on electromagnetic wave theory. IEEE Press, 2001.

- [36] C. Geuzaine and J. Remacle, "Gmsh: A 3-D finite element mesh generator with built-in pre-and post-processing facilities," *International Journal for Numerical Methods in Engineering*, vol. 79, no. 11, pp. 1309–1331, 2009.
- [37] Gmsh website. [Online]. Available: <http://geuz.org/gmsh/>
- [38] A. Squillacote, *The ParaView guide: a parallel visualization application*. Kitware, 2007.
- [39] Paraview website. [Online]. Available: <http://www.paraview.org/>
- [40] D. Isleifson, I. Jeffrey, L. Shafai, J. LoVetri, and D. G. Barber, "An efficient scattered-field formulation for objects in layered media using the fvtd method," *IEEE Trans. Antennas Propag.*, (submitted for publication) 2010.
- [41] G. Ledfelt, "A stable subcell model for arbitrarily oriented thin wires for the FDTD method," *Int. J. Numer. Model.*, vol. 15, pp. 503–515, 2002.
- [42] C. J. Railton, B. P. Koh, and I. J. Craddock, "The treatment of thin wires in the FDTD method using a weighted residuals approach," *IEEE Transactions on Antennas and Propagation*, vol. 52, no. 11, pp. 2941–2949, November 2004.
- [43] R. Holland and L. Simpson, "Finite-difference analysis of EMP coupling to thin struts and wires," *IEEE Transactions on Electromagnetic Compatibility*, vol. EMC-23, no. 2, pp. 88–97, May 1981.
- [44] K. R. Umashankar, A. Taflove, and B. Becker, "Calculation and experimental verification of induced currents on coupled wires in an arbitrarily shaped cavity," *IEEE Journal on Antennas and Propagation*, vol. 35, pp. 1248–1257, 1987.
- [45] R. Makinen, J. Juntunen, and M. Kivikoski, "An improved thin-wire model for FDTD," *Microwave Theory and Techniques, IEEE Transactions on*, vol. 50, no. 5, pp. 1245–1255, 2002.
- [46] A. Bretones, R. Mittra, and R. Martín, "A hybrid technique combining the method of moments in the time domain and FDTD," *Microwave and Guided Wave Letters, IEEE*, vol. 8, no. 8, pp. 281–283, 2002.
- [47] A. Monorchio, A. Bretones, R. Mittra, G. Manara, and R. Martín, "A hybrid time-domain technique that combines the finite element, finite difference and method of moment techniques to solve complex electromagnetic problems," *Antennas and Propagation, IEEE Transactions on*, vol. 52, no. 10, pp. 2666–2674, 2004.
- [48] G. Ledfelt, "Hybrid time-domain methods and wire models for computational electromagnetics," Ph.D. dissertation, Royal Institute of Technology, Stockholm, Sweden, 2001.

- [49] B. Van Leer, "Towards the ultimate conservative difference scheme. II. Monotonicity and conservation combined in a second-order scheme," *Journal of Computational Physics*, vol. 14, no. 4, pp. 361–370, 1974.
- [50] C. Shu, "Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws," Tech. Rep., 1998.
- [51] C. Balanis, *Antenna theory*. Wiley New York, 1997.
- [52] J. Jin, *Electromagnetic analysis and design in magnetic resonance imaging*, ser. Biomedical engineering series. CRC Press, 1999.
- [53] Z. Liang, P. Lauterbur, I. E. in Medicine, and B. Society, *Principles of magnetic resonance imaging: a signal processing perspective*. SPIE Optical Engineering Press, 2000.
- [54] C. Paul, *Introduction to electromagnetic compatibility*, ser. Wiley series in microwave and optical engineering. Wiley, 1992.
- [55] J. LoVetri and T. Lapohos, "Explicit upwind schemes for lossy MTL's with linear terminations," *IEEE Transactions on Electromagnetic Compatibility*, vol. 39, no. 3, pp. 189–200, August 1997.
- [56] J. Vlach and K. Singhal, *Computer Methods for Circuit Analysis and Design*, 2nd ed. New York, NY: Van Nostrand Reinhold, 1994.
- [57] A. Zakaria, C. Kaye, I. Jeffrey, and J. LoVetri, "Experimental validation of thin-wire fvtd models," in *ANTEM/URSI*, Banff, Alberta, February 2009.
- [58] Numerical Electromagnetics Code (NEC). [Online]. Available: <http://www.nec2.org/>
- [59] F. Terman, *Radio engineers' handbook*. McGraw-Hill Publishing Co. Ltd, 1950.
- [60] D. M. Sullivan, P. Wust, and J. Nadobny, "Accurate FDTD simulation of RF coils for MRI using the thin-rod approximation," *IEEE Transactions on Antennas and Propagation*, vol. 58, no. 6, pp. 2004–2011, June 2010.
- [61] C. Rowell, V. Shankar, W. Hall, and A. Mohammadian, "Algorithmic aspects and computing trends in computational electromagnetics using massively parallel architectures," in *Algorithms and Architectures for Parallel Processing, 1995. ICAPP 95. IEEE First ICA/sup 3/PP., IEEE First International Conference on*, vol. 2, Apr. 1995, pp. 770–779 vol.2.
- [62] D. Blake and T. Buter, "Electromagnetic scattering simulations using overset grids on massively parallel computing platforms," in *Antennas and Propagation Society International Symposium, 1997. IEEE., 1997 Digest*, vol. 1. IEEE, 2002, pp. 106–109.



- [63] J. Shang, M. Wagner, Y. Pan, and D. Blake, "Strategies for adopting FVTD on multicomputers [finite-volume time-domain analysis]," *Computing in Science & Engineering*, vol. 2, no. 1, pp. 10–21, 2002.
- [64] L. Baduel, F. Baude, D. Caromel, C. Delbe, N. Gama, S. Kasmi, and S. Lanteri, "A parallel object-oriented application for 3d electromagnetism," in *Parallel and Distributed Processing Symposium, 2004. Proceedings. 18th International*, 2004, p. 7.
- [65] J. Camberos and P. Wheat, "A finite-volume, time-domain CEM code for unstructured-grids on parallel computers," in *Computational Electromagnetics in Time-Domain, 2005. CEM-TD 2005. Workshop on*. IEEE, 2005, pp. 40–43.
- [66] V. Ahuja and L. Long, "Scattering from complex geometries using a parallel FVTD algorithm," in *12th Annual Review of Progress in Applied Computational Electromagnetics*, vol. 2, pp. 1072–1082.
- [67] G. Andrews, *Foundations of multithreaded, parallel, and distributed programming*. Addison-Wesley, 2000.
- [68] B. Wilkinson and C. Allen, *Parallel programming: techniques and applications using networked workstations and parallel computers*. Pearson Prentice Hall, 2005.
- [69] M. Snir, *MPI—the Complete Reference: The MPI core*, ser. Scientific and engineering computation. MIT Press, 1998.
- [70] S. Sur, M. Koop, and D. Panda, "High-performance and scalable MPI over InfiniBand with reduced memory usage: an in-depth performance analysis," in *Proceedings of the 2006 ACM/IEEE conference on Supercomputing*. ACM, 2006, p. 105.
- [71] "The official domain decomposition website," accessed December, 2010. [Online]. Available: <http://www.ddm.org>
- [72] "The official domain decomposition bibtex bibliography," accessed Dec, 2010. [Online]. Available: <http://www.ddm.org/Bibtex/bibtex.html>
- [73] S. Baden, "Run-time partitioning of scientific continuum calculations running on multiprocessors," 1987.
- [74] J. Salmon, *Parallel hierarchical N-body methods*. California Institute of Technology, 1991.
- [75] S. Warren, "Astrophysical N-body simulations using hierarchical tree data structures," 1992.

- [76] K. Devine, E. Boman, R. Heaphy, R. Bisseling, and U. Catalyurek, "Parallel hypergraph partitioning for scientific computing," in *Parallel and Distributed Processing Symposium, 2006. IPDPS 2006. 20th International*. IEEE, 2006, p. 10.
- [77] U. Catalyurek, D. Bozdag, E. Boman, K. Devine, R. Heaphy, and L. Riesen, "Hypergraph-Based Dynamic Partitioning and Load Balancing," *Advanced Computational Infrastructures for Parallel and Distributed Applications*, p. 313, 2010.
- [78] J. Hungershöfer and J. Wierum, "On the quality of partitions based on space-filling curves," *Computational Science-ICCS 2002*, pp. 36–45, 2009.
- [79] J. Luitjens, M. Berzins, and T. Henderson, "Parallel space-filling curve generation through sorting," *Concurrency and Computation: Practice and Experience*, vol. 19, no. 10, pp. 1387–1402, 2007.
- [80] I. Jeffrey, D. Firsov, C. Gilmore, V. Okhmatovski, and J. LoVetri, "Parallel higher-order EM-FVTD on an unstructured mesh," in *ACES*, Verona, Italy, March 2007.
- [81] K. Devine, E. Boman, R. Heaphy, B. Hendrickson, and C. Vaughan, "Zoltan data management services for parallel dynamic applications," *Computing in Science and Engineering*, vol. 4, no. 2, pp. 90–97, 2002.
- [82] E. Boman, K. Devine, L. A. Fisk, R. Heaphy, B. Hendrickson, V. Leung, C. Vaughan, U. Catalyurek, D. Bozdag, and W. Mitchell, "Zoltan home page," 1999, <http://www.cs.sandia.gov/Zoltan>.
- [83] E. Boman, K. Devine, L. A. Fisk, R. Heaphy, B. Hendrickson, C. Vaughan, U. Catalyurek, D. Bozdag, W. Mitchell, and J. Teresco, *Zoltan 3.0: Parallel Partitioning, Load-balancing, and Data Management Services; User's Guide*, Sandia National Laboratories, Albuquerque, NM, 2007, tech. Report SAND2007-4748W [http://www.cs.sandia.gov/Zoltan/ug\\_html/ug.html](http://www.cs.sandia.gov/Zoltan/ug_html/ug.html).
- [84] C. Gilmore, P. Mojabi, and J. LoVetri, "Comparison of an enhanced distorted born iterative method and the multiplicative-regularized contrast source inversion method," *Antennas and Propagation, IEEE Transactions on*, vol. 57, no. 8, pp. 2341–2351, 2009.
- [85] P. Bonnet, X. Ferrieres, J. Grando, J. Alliot, and J. Fontaine, "Frequency-domain finite volume method for electromagnetic scattering," in *Antennas and Propagation Society International Symposium, 1998. IEEE*, vol. 1. IEEE, 2002, pp. 252–255.
- [86] J. Jin, *The finite element method in electromagnetics*, ser. A Wiley-Interscience publication. Wiley, 2002.
- [87] C. Phillips, J. Parr, and E. Riskin, *Signals, systems, and transforms*. Prentice Hall, 2003.

- [88] Y. Saad, *Iterative methods for sparse linear systems*, ser. The PWS series in computer science. PWS Pub. Co., 1996.
- [89] G. Sleijpen and D. Fokkema, “BiCGStab (l) for linear equations involving unsymmetric matrices with complex spectrum,” *Electronic Transactions on Numerical Analysis*, vol. 1, no. 11, p. 2000, 1993.
- [90] J. Berenger, “A perfectly matched layer for the absorption of electromagnetic waves,” *Journal of computational physics*, vol. 114, no. 2, pp. 185–200, 1994.
- [91] C. Gilmore, P. Mojabi, A. Zakaria, M. Ostadrahimi, C. Kaye, S. Noghianian, L. Shafai, S. Pistorius, and J. LoVetri, “A wideband microwave tomography system with a novel frequency selection procedure,” *Biomedical Engineering, IEEE Transactions on*, vol. 57, no. 4, pp. 894–904, 2010.
- [92] A. Ergin, B. Shanker, and E. Michielssen, “The plane-wave time-domain algorithm for the fast analysis of transient wave phenomena,” *Antennas and Propagation Magazine, IEEE*, vol. 41, no. 4, pp. 39–52, 2002.
- [93] S. Lohrengel and M. Remaki, “A FV scheme for Maxwell’s equations: Convergence analysis on unstructured meshes,” in *Finite Volumes for Complex Applications III*, pp. 219–226.
- [94] G. Cohen, X. Ferrieres, and S. Pernet, “A spatial high-order hexahedral discontinuous Galerkin method to solve Maxwell’s equations in time domain,” *Journal of Computational Physics*, vol. 217, no. 2, pp. 340–363, 2006.
- [95] H. Fahs, L. Fezoui, S. Lanteri, and F. Rapetti, “Preliminary investigation of a non-conforming discontinuous Galerkin method for solving the time-domain Maxwell equations,” *Magnetics, IEEE Transactions on*, vol. 44, no. 6, pp. 1254–1257, 2008.
- [96] L. Pebernet, X. Ferrieres, S. Pernet, B. Michielsen, F. Rogier, and P. Degond, “Discontinuous Galerkin method applied to electromagnetic compatibility problems: introduction of thin wire and thin resistive material models,” *Science, Measurement & Technology, IET*, vol. 2, no. 6, pp. 395–401, 2008.
- [97] P. Sewell, T. Benson, A. Vukovic, and A. Jarro, “The challenges for numerical time domain simulations of optical resonators,” in *Transparent Optical Networks (ICTON), 2010 12th International Conference on*. IEEE, 2010, pp. 1–4.
- [98] S. Garcia, F. Pantoja, R. Bretones, G. Martin *et al.*, “A new hybrid DGTD/FDTD method in 2-D,” *Microwave and Wireless Components Letters, IEEE*, vol. 18, no. 12, pp. 764–766, 2008.