Finite-Element Time-Domain Methods for Nonlinear Dispersive Media

David S. Abraham

Presented in partial fulfillment of the requirements of the degree of
Doctor of Philosophy

Department of Electrical & Computer Engineering
McGill University, Montréal, Québec, Canada
July 2020

Abstract

Today's telecommunications infrastructure is increasingly reliant upon complex material interactions with the electromagnetic field. For instance, effects such as dispersion, in which a material's response to an applied field depends on its frequency, and nonlinearity, where the response is a complex function of field strength, form the cornerstones of fields such as nonlinear fibre optics. In consequence, efficient, accurate, and reliable numerical simulation tools capable of modeling these complex interactions are increasingly in demand, as cost effective alternatives to physical experimentation and prototyping.

In this thesis, a family of Finite-Element Time-Domain (FETD) based numerical methods for the simulation of electromagnetic problems containing electrically complex material interactions is presented. Making use of both the mixed and vector wave equation formulations, the derived methods are capable of modeling very general combinations of linear dispersion, instantaneous nonlinearity, and dispersive nonlinearity within the nonlinear Maxwell's Equations, free from any simplifying assumptions about the nature of the field solutions. In contrast to existing methods, these techniques permit increased geometric freedom, improved stability, and are capable of handling arbitrarily high nonlinear and dispersive orders.

This thesis also presents several additional tools and methods to increase the effectiveness and versatility of these techniques. For instance, a Perfectly Matched Layer (PML) is derived which is compatible with nonlinear dispersive media, permitting the emulation of infinite domains as well as the truncation of finite systems within the nonlinear FETD method. More specifically, by utilizing the stretched coordinate formulation of the PML, the resulting technique not only saves computational resources, but does so without significantly altering

the original underlying algorithms.

Furthermore, while the derived techniques permit a much more accurate and general solution to Maxwell's Equations for complex media, they unfortunately do so with a significant added computational burden. To mitigate this fact, this thesis also presents an analysis and breakdown of the computational overhead and bottlenecks associated with these methods. Building upon this analysis, a scheme is then presented by which these algorithms may be accelerated via parallelism and implemented on Graphics Processing Units (GPUs) to help alleviate some of the burden they pose.

Lastly, in each case the FETD, PML, and GPU algorithms proposed in this dissertation are tested via numerical studies to verify their proper functioning, convergence, and accuracy. These include convergence studies as well as the demonstration of several well-known and physically significant nonlinear phenomena, such as spatial solitons, temporal solitons, and supercontinuum generation. Moreover, a parallel GPU implementation of the nonlinear algorithm is benchmarked against an equivalent traditional serial Central Processing Unit (CPU) version, and is shown to perform up to 150 times faster, significantly increasing the applicability and usefulness of these algorithms.

Résumé

L'infrastructure des télécommunications d'aujourd'hui dépend de plus en plus des interactions complexes entre les matériaux et les champs électromagnétiques. Par exemple, des effets tels que la dispersion, où la réponse d'un matériel à un champ appliqué dépend de sa fréquence, et la non-linéarité, où la réponse est une fonction complexe de l'ampleur du champ, forment la base de plusieurs domaines d'études tels que les fibres optiques non-linéaires. En conséquence, la demande d'outils de simulation numérique capables de modéliser ces interactions complexes d'une manière efficace, précise et fiable augmente en tant d'alternatives abordables à l'expérimentation et au prototypage.

Dans cette thèse, une famille de méthodes numériques basées sur la méthode des éléments finis au domaine temporel (FETD) est proposée pour la simulation des problèmes électromagnétiques contenant des interactions matérielles complexes. En utilisant à la fois une formulation mixte et une formulation de l'équation d'ondes vectorielles, les méthodes présentées sont capables de modéliser des combinaisons très générales de dispersion linéaire, de non-linéarité instantanée et de non-linéarité dispersive dans les équations de Maxwell d'une manière complète et libre de toute hypothèse simplificatrice sur la nature des solutions. Contrairement aux méthodes existantes, ces techniques permettent une liberté géométrique accrue, une stabilité améliorée et sont capables de gérer des ordres dispersifs et non-linéaire arbitrairement élevés.

Cette thèse présente également plusieurs outils et méthodes supplémentaires pour augmenter l'efficacité et la polyvalence de ces techniques. Par exemple, une couche absorbante

parfaitement adaptée (PML) compatible avec les milieux dispersifs et non-linéaire est dérivée, permettant l'émulation des domaines infinies ainsi que la troncature de systèmes finis au sein de la méthode FETD non-linéaire. Plus précisément, en utilisant la formulation de coordonnées étirées du PML, la technique qui résulte permet non seulement d'économiser des ressources, mais fait ceci sans altérer de manière significative les algorithmes originaux sous-jacents.

De plus, bien que les techniques dérivées permettent une solution beaucoup plus précise et générale aux équations de Maxwell aux milieux complexes, elles le font malheureusement avec une charge de calcul supplémentaire importante. Pour mitiger ce fait, cette thèse présente également une analyse des goulots d'étranglement associés à ces méthodes. Sur la base de cette analyse, un schéma est ensuite présenté par lequel les algorithmes proposés peuvent être parallélisés et mis en oeuvre sur des processeurs graphiques (GPU), apaisant largement le fardeau qu'ils posent.

Dernièrement, en chaque cas, les algorithmes FETD, PML et GPU proposés dans cette thèse sont évalués via des études numériques pour vérifier leurs bons fonctionnements, leur convergence et leur précision. Celles-ci incluent des études de convergence ainsi que la démonstration de plusieurs phénomènes non-linéaires bien connus et significatifs, tels que les solitons spatiaux, les solitons temporels et la génération de supercontinuum. De plus, une mise en oeuvre parallèle sur GPU de l'algorithme non-linéaire est comparée à une version traditionnelle exécutée en série sur un processeur central (CPU), et démontre un temps d'exécution 150 fois plus rapide, ce qui augmente considérablement l'application et l'utilité de ces algorithmes.

Table of Contents

List of Figures

List of Abbreviations

Abbreviation	Definition
ABC	Absorbing Boundary Condition
ADE	Auxiliary Differential Equation
BPM	Beam Propagation Method
CFL	Courant-Friedrichs-Lewy
CPU	Central Processing Unit
CRT	Cathode Ray Tube
CUDA	Compute Unified Device Architecture
DoF	Degrees of Freedom
FD	Finite-Difference
FDTD	Finite-Difference Time-Domain
FETD	Finite-Element Time-Domain
FGaBP	Finite-Element Gaussian Belief Propagation
FLOPS	Floating Point Operations Per Second
FWHM	Full Width at Half Maximum
GMRES	Generalized Minimum Residual
GPU	Graphics Processing Unit
GVD	Group Velocity Dispersion
JFNK	Jacobian-Free Newton Krylov
MIPML	Material Independent Perfectly Matched Layer
NLSE	Nonlinear Schrödinger Equation
PCG	Preconditioned Conjugate Gradient
PEC	Perfect Electric Conductor
PML	Perfectly Matched Layer
RAM	Random Access Memory
RC	Recursive Convolution
SIMT	Single Instruction Multiple Threads
SM	Streaming Multiprocessor
SVEA	Slowly Varying Envelope Approximation
TM	Transverse Magnetic
VWE	Vector Wave Equation
ZDW	Zero Dispersion Wavelength

List of Publications

The following peer-reviewed journal publications and conference presentations have resulted from the work presented in this thesis:

Peer-Reviewed Publications

[1] D. S. Abraham and D. D. Giannacopoulos, "A convolution-free mixed finite-element time-domain method for general nonlinear dispersive media," *IEEE Transactions on Antennas and Propagation*, vol. 67, no. 1, pp. 324 – 334, Jan. 2019.

[2] D. S. Abraham and D. D. Giannacopoulos, "A perfectly matched layer for the nonlinear dispersive finite-element time-domain method," *IEEE Transactions on Magnetics*, vol. 55, no. 6, Jun. 2019.

[3] D. S. Abraham and D. D. Giannacopoulos, "A convolution-free finite-element time-domain method for the nonlinear dispersive vector wave equation," *IEEE Transactions on Magnetics*, vol. 55, no. 12, Dec. 2019.

[4] D. S. Abraham and D. D. Giannacopoulos, "A parallel finite-element time-domain method for nonlinear dispersive media," *IEEE Transactions on Magnetics*, vol. 56, no. 2, Feb. 2020.

Conference Presentations

[1] D. S. Abraham* and D. D. Giannacopoulos, "A perfectly matched layer for the nonlinear dispersive finite-element time-domain method," Oct. 2018, poster presentation at the biennial Conference on Electromagnetic Field Computation (CEFC) hosted in Hangzhou, China.

[2] D. S. Abraham* and D. D. Giannacopoulos, "A convolution-free finite-element time-domain method for the nonlinear dispersive vector wave equation," Oct. 2018, oral presentation at the biennial Conference on Electromagnetic Field Computation (CEFC) hosted in Hangzhou, China.

[3] D. S. Abraham* and D. D. Giannacopoulos, "A parallel finite-element time-domain method for nonlinear dispersive media," Jul. 2019, poster presentation at the 22nd International Conference on the Computation of Electromagnetic Fields (Compumag) hosted in Paris, France.

Acknowledgements

This work would not have been possible without the guidance and support of many individuals and organizations over the years. In particular, I would like to express my deepest gratitude to my supervisor Prof. Dennis Giannacopoulos, whose patience, guidance, integrity, and commitment to his students have made him an instrumental paragon in my development as a researcher and teacher. Furthermore, I would like to thank McGill University and the Natural Sciences and Engineering Research Council of Canada (NSERC) for their generous financial support, without which this work would have undoubtedly suffered.

Moreover, my most heartfelt thanks go out to my friends and family for their endless affection and assistance throughout my time at McGill. In particular, I could not have dreamed of being where I am today without the unwavering support of Mrs. Lisa Piedimonte and Mr. Stephen Abraham. Last, but certainly not least, I also owe an immense debt of gratitude to Dr. Kirsten Gust, for her inspiration, unflinching patience, and boundless empathy.

Chapter 1

Introduction

1.1 Motivation

Classical electromagnetic theory as it is known today was largely developed over the course of the 19th century. A collective work by many physicists such as Faraday, Ampère, Helmholtz, and Hertz, their work began to demystify and explain many electromagnetic phenomena, some of which had been known since antiquity. However, it was not until the contributions of Maxwell that electromagnetic theory fully began to take form, providing a unified and consistent picture of electric and magnetic phenomena.

Despite these triumphs of mathematics and physics, these scientists were often pessimistic about the utility of their work. After his discovery of electromagnetic waves, Hertz is reputed to have remarked, "It is of no use whatsoever" [1]. Of course, in hindsight, this assessment proved inaccurate to say the least. In the over one hundred years since the synthesis of Maxwell's Equations, electromagnetic theory has become a dominant and ubiquitous force within society. From electronics, computers, and magnetic machines, to telecommunications, radio, fibre optics, and medical imaging, there are few aspects of modern life that do not leverage electromagnetic theory in some way.

As a natural consequence, the design and manufacture of electromagnetic devices plays a critical part in many commercial and industrial enterprises. Being able to accurately predict how a given device or system will interact with or generate electromagnetic fields and waves is often paramount to the design process. However, with the immense complexity

inherent in most problems in electromagnetics, closed-form exact solutions of Maxwell's Equations are often unrealistic or impossible to obtain. This leaves simulations and numerical methods as the only viable cost-effective alternatives to expensive physical experimentation and prototyping. Unsurprisingly, this has lead to the development of several numerical techniques for simulating electromagnetic phenomena. In particular, techniques such as the Finite-Difference and Finite-Element methods are among the most popular, having borne a wealth of scientific literature and commercial software packages over the years.

While electromagnetic problems in general can prove to be quite nuanced and complex, simplifying assumptions may often be made about the systems and materials being studied, to ease their analysis. In particular, assuming symmetry, time-independence, the presence of only a single frequency component (time-harmonic form), or that materials respond linearly and instantaneously to any applied fields all significantly reduce the complexity of the underlying equations. It is no surprise, therefore, that the first electromagnetic Finite-Difference and Finite-Element algorithms operated in these idealistic regimes.

Unfortunately, for many problems such simplifications may be invalid or result in unacceptable losses in accuracy. For instance, material dispersion, in which a medium's response to electromagnetic waves is not instantaneous but rather depends on the frequency, is a ubiquitous phenomena in nature [2]. While negligible under certain circumstances, dispersion can significantly impact the shape and structure of electromagnetic signals in others. As a result, being able to effectively model these frequency-dependent interactions can have far-reaching impacts, such as in the study of dispersive biological tissue samples in medical imaging [3, 4], dispersive environmental elements in radar applications [5, 6], and pulse broadening and distortion in fibre optic cables [7, 8].

Moreover, nonlinear effects, in which the medium's response can be a complicated function of the applied field strength, can also significantly increase complexity. In particular, the nonlinear response can be immediate, known as an instantaneous nonlinearity, or depend on the frequency of the applied field, known as a dispersive nonlinearity. This nonlinear be-

haviour is most commonly encountered when materials are subjected to very large field strengths, inducing a myriad of complex behaviours. Modeling these nonlinear interactions can thus have important implications for fields such as nonlinear optics, in which very high intensity laser light is routinely encountered. Such techniques could therefore be used, for example, in the design and investigation of photonic crystals [9, 10] and soliton formation and interaction [11, 12]. Moreover, it has been theoretically and empirically demonstrated that biological tissues, in addition to being dispersive, can also produce nonlinear effects [13, 14, 15]. Modeling such behaviour thus has the potential not only to increase accuracy in medical imaging but also to detect malignancies. For instance, it has been shown that mitochondria with metabolic disorders generally have suppressed harmonic generation as compared to their healthy counterparts when excited by a single frequency source [16].

In fact, the need for more general numerical methods to model complex material interactions such as these will only continue to grow as our ability to precisely design and engineer materials improves. Indeed, new materials and structures with properties unseen in nature can now be fabricated, such as so-called negative and zero-index metamaterials [17, 18]. Moreover, as technology advances, existing materials can be further exposed to extreme operating conditions, causing effects usually deemed negligible to manifest more often. As a consequence of these existing and emerging media and their applications, there is a clear and growing need for numerical methods that are able to accurately simulate very general and complex material interactions with electromagnetic fields, and that are free of inaccurate simplifying assumptions.

Fortunately, over the past decades many of the original electromagnetic numerical methods and techniques have evolved to address some of these new challenges in material modeling. For instance, generalizations such as the Finite-Difference Time-Domain (FDTD) [19] and Finite-Element Time-Domain (FETD) [20] methods have permitted the full broadband treatment of wave phenomena containing multiple frequency components all in one simulation. Moreover, these FDTD and FETD methods have both been successfully adapted to

treat dispersive media, using a number of different techniques. The analysis of nonlinear problems, in contrast, has proven more problematic. While FDTD methods have been devised in which linear dispersion, instantaneous nonlinearity, and dispersive nonlinearity are all included, many of them impose stability constraints, geometric constraints, or constraints on the classes of nonlinear materials which may be treated, among others. As a result, these and other considerations can significantly reduce their applicability and versatility.

Some of these shortcomings can be alleviated by the use of FETD-based methods. Naturally, as mentioned earlier, several FETD methods have therefore already successfully been implemented for the treatment of linear dispersive media. However, despite their potential advantages, to date very few if any FETD-based methods have been devised that are capable of including the treatment of electrical nonlinearity.

The goal of this thesis, therefore, is to facilitate the study of complex electromagnetic material interactions via the development and refinement of novel numerical techniques for the treatment of electric nonlinearity in materials. In particular, this thesis aims to develop a family of FETD-based methods capable of modeling arbitrary combinations of linear dispersion, instantaneous nonlinearity, and dispersive nonlinearity, free from the aforementioned drawbacks and simplifying assumptions present in current methods. Furthermore, additional tools and techniques for these methods will also be developed, such as artificial absorbing perfectly matched layers, to increase their utility and applicability. Lastly, while FETD-based methods are advantageous in many ways, they tend to be more computationally demanding. In consequence, this thesis will also consider a method by which the derived algorithms can be optimized and refined to reduce their computational burden. Most notably, the use of Graphics Processing Units (GPUs) will be investigated to develop parallel implementations of these new algorithms. In this way, this thesis intends to provide accurate, wide-reaching, and general numerical methods for electrically complex media, whose efficient implementation permits detailed and accurate analyses of complicated electromagnetic devices, without debilitating overhead.

The remainder of this chapter provides an overview of the main types of material interactions of interest, with an emphasis on how they complicate the mathematical treatment of electromagnetic phenomena. A detailed look at some of the existing literature, methods, and techniques for these materials will then be provided, with an analysis of the benefits and drawbacks of each. Lastly, an overview of the approach and contributions of this thesis will be provided, as well as a brief outline of the chapters to follow.

1.2 Background

1.2.1 Electrically Complex Materials

When an electromagnetic wave propagates through a medium it naturally interacts with and exerts forces on the charged particles that constitute the material. Generally, these forces result in each constituent atom or molecule stretching or deforming and thus acquiring an electric dipole moment. Macroscopically, this polarization of the material can be described via a polarization density vector $\vec{P}$, such that the resulting displacement field $\vec{D}$ is related to the electric field $\vec{E}$ via the constitutive equation [2]:

$$\vec{D} = \epsilon_0 \vec{E} + \vec{P} \tag{1.1}$$

where ϵ_0 is the permittivity of free space.

For many materials, the polarization density is approximately linearly proportional to the applied electric field, such that:

$$\vec{P} = \epsilon_0 \chi_e \vec{E} \tag{1.2}$$

where the constant χ_e is the electric susceptibility. With this assumption, the relation between $\vec{D}$ and $\vec{E}$ can also be written as:

$$\vec{D} = \epsilon_0 (1 + \chi_e) \vec{E} = \epsilon_0 \epsilon_r \vec{E} = \epsilon \vec{E} \tag{1.3}$$

where $\epsilon_r = 1+\chi_e$ is the relative permittivity, and $\epsilon = \epsilon_r\epsilon_0$ is simply known as the permittivity.

One important consequence of equation (1.2) is that a change in the applied electric field results in an instantaneous change in the polarization state. While approximately valid for many materials, such a relationship is usually non-physical. In reality, a given material is unable to react instantaneously to changes in the applied field [21], with the current polarization state thus depending on not only the material's characteristics but also past history of interaction with the field [2]. The material's ability to respond to an applied field will thus depend on how rapidly the field is changing, and as a consequence, the permittivity can effectively be considered a function of frequency, such that in the Fourier domain:

$$\vec{D}(\omega) = \epsilon(\omega)\vec{E}(\omega) \tag{1.4}$$

where ω is the angular frequency and $\epsilon(\omega)$ now describes a *dispersive* medium. As a result, should a signal of interest contain multiple spectral components, each will experience a different propagation velocity according to their respective frequencies. An initial signal can thus become distorted or broadened as it propagates, a phenomena most often characterized by a quantity known as the Group Velocity Dispersion (GVD) [21].

It is worth noting that the permittivity so defined cannot be an arbitrary function, but must instead satisfy a set of equations known as the Kramers-Kronig relations [2]. These equations fundamentally arise from the requirement that the material's response be causal, providing a link between the frequency variation and energy losses within the dielectric medium. Transforming equation (1.4) from the frequency-domain to the time-domain (via an inverse Fourier transform), it is evident that dispersion results in a time-dependent permittivity, as well as a convolution in time between the permittivity and the applied field:

$$\vec{D}(t) = \epsilon(t) * \vec{E}(t) \tag{1.5}$$

where $*$ denotes the convolution. While equation (1.5) is manifestly more complex than

(1.3) due to the dispersion, it is important to note that it remains linear in $\vec{E}$, due to the linearity property of the convolution operator [22].

In contrast, when exposed to high field strengths, a given material may no longer respond linearly to the applied field. In the most general case, rather than being directly proportional, the polarization density can instead be an arbitrary function of the electric field, such that:

$$\vec{D} = \epsilon_0 \vec{E} + \vec{P}(\vec{E}). \tag{1.6}$$

To simplify the analysis of such nonlinear interactions, it is usually customary to expand the polarization density as a power series in the electric field. While such an expansion would in the most general case involve susceptibility tensors, for many isotropic material models an adequate expansion is instead given by [23]:

$$\vec{P} = \epsilon_0 \chi^{(1)} \vec{E} + \epsilon_0 \chi^{(2)} E \vec{E} + \epsilon_0 \chi^{(3)} E^2 \vec{E} \tag{1.7}$$

where $\chi^{(n)}$ is the nth-order susceptibility, and E is the magnitude of the electric field. While a third-order approximation is often sufficient, in principle higher order terms could equally be included in the above expansion. Moreover, for many materials, symmetry within the crystal lattice structure of the medium precludes the existence of the $\chi^{(2)}$ term, a phenomena known as inversion symmetry [23]. As mentioned earlier, this nonlinear polarization can result in numerous additional and unique phenomena, such as the Kerr effect, frequency doubling, self-phase modulation, solitons, four-wave mixing, and supercontinuum generation [23, 24, 25].

The nonlinear polarization in (1.7) represents an instantaneous nonlinearity, as changes in the applied field are reflected immediately in the polarization. However, just as in the linear case, this is not necessarily a physical result and the nonlinear response can also be a function of frequency, thereby representing a dispersive nonlinearity. Much as in the linear case, this nonlinear dispersion corresponds to the introduction of additional convolutions into the

nonlinear model of (1.7) and gives rise to additional phenomena such as stimulated Raman scattering [23, 26]. As a result, incorporating linear dispersion, instantaneous nonlinearity, as well as dispersive nonlinearity into a single model for the polarization density can be accomplished quite accurately via the following expression for $\vec{P}$ [27, 28]:

$$\vec{P} = \epsilon_0 \chi^{(1)}(t) * \vec{E} + \epsilon_0 \chi^{(3)} \Big(\alpha E^2 + (1-\alpha) g(t) * E^2 \Big) \vec{E} \tag{1.8}$$

in which the susceptibility $\chi^{(1)}$ models the linear dispersion, the susceptibility $\chi^{(3)}$ an instantaneous (Kerr) and/or dispersive (Raman) nonlinearity (with the α term controlling their relative strengths), and $g(t)$ being a causal response function characterizing the nonlinear dispersion. The polarization in (1.8) represents a very general model capable of including a wide variety of material interactions and phenomena, and will form the basis for the inclusion of these effects within Maxwell's Equations in this thesis.

It is worth noting, however, that while very general, there are still materials for which equation (1.8) may not be adequate. Effects such as anisotropy, ferroelectricity, and electric hysteresis can all add additional complexity to the polarization models described above. Moreover, the discussion presented thus far has neglected the magnetic properties of the materials in question. In like manner to the electric case, the magnetization state of the medium can be characterized by a magnetization density $\vec{M}$ and permeability μ, and can thus equally exhibit phenomena such as magnetic dispersion, nonlinearity, and ferromagnetism.

However, since one of the main intended applications of this thesis is within the realm of nonlinear optics, where the effects mentioned above are often negligible or absent entirely, this thesis will focus principally on those materials whose polarization state is adequately described by equation (1.8). In addition, given their immense practical use, many numerical methods have already been devised for the treatment of dispersive and nonlinear magnetic materials within both Finite-Difference and Finite-Element-based methods (see Section 1.2.3). Thus, to simplify the following analysis, it will be assumed that the materi-

als in question have a linear and instantaneous response to an applied magnetic field, such that the permeability μ is a simple scalar constant. Any future references to nonlinearity in this thesis will thus refer implicitly to *electrical* nonlinearity. Despite these simplifying assumptions, the methods proposed in this thesis nonetheless represent a powerful tool for modeling very general material interactions and will help lay the framework for even more general methods which combine these techniques, thus allowing all of the above phenomena to be treated simultaneously.

1.2.2 The Nonlinear Wave Equation

For an electrically complex material whose polarization state is adequately described by equation (1.8), the underlying electromagnetic theory remains Maxwell's Equations:

$$\nabla \cdot \vec{D} = \rho_v \tag{1.9}$$

$$\nabla \cdot \vec{B} = 0 \tag{1.10}$$

$$\nabla \times \vec{E} = -\frac{\partial \vec{B}}{\partial t} \tag{1.11}$$

$$\nabla \times \vec{H} = \vec{J}_v + \frac{\partial \vec{D}}{\partial t} \tag{1.12}$$

where ρ_v is a volume charge density, $\vec{J}_v$ is a volume current density, $\vec{B} = \mu\vec{H}$ is the magnetic flux density, and as mentioned previously, $\vec{D} = \epsilon_0\vec{E} + \vec{P}$ [21].

In a source-free region, the above Maxwell's Equations, along with the expression for the polarization in equation (1.8), can be combined to yield a governing nonlinear vector wave equation for the electric field [29]:

$$\nabla^2\vec{E} - \mu\epsilon_L * \frac{\partial^2\vec{E}}{\partial t^2} = \mu\frac{\partial^2\vec{P}_{NL}}{\partial t^2} \tag{1.13}$$

in which ϵ_L is the linear dispersive term (related to $\chi^{(1)}$) and $\vec{P}_{NL}$ is the nonlinear part of equation (1.8), related to $\chi^{(3)}$. In general, solving this nonlinear wave equation can be

immensely difficult, if not impossible, in closed form. In consequence, most techniques tend to apply simplifying assumptions to equation (1.13) in order to ease the analysis and obtain approximate solutions.

One of the most popular approximative techniques is the Slowly Varying Envelope Approximation (SVEA). As evident in the name, SVEA assumes that the amplitude or envelope of the electromagnetic wave varies slowly in time and space as compared to the period or wavelength. As a consequence, when expanding the wave equation in (1.13), higher order derivatives can be neglected, such that [29, 30]:

$$\left|\frac{\partial^2 E}{\partial z^2}\right| \ll \left|\beta \frac{\partial E}{\partial z}\right| \tag{1.14}$$

$$\left|\frac{\partial^2 E}{\partial t^2}\right| \ll \left|\omega \frac{\partial E}{\partial t}\right| \tag{1.15}$$

with z denoting the direction of propagation, β the wavenumber, and ω the angular frequency. This results in a significantly simplified analysis.

The SVEA also forms the basis for a number of additional methods, each of which builds upon it by adding additional constraints, assumptions, or approximations. For instance, the Beam Propagation Method (BPM) [31, 32] not only assumes the signal envelope to be slowly varying, but also treats the problem in time-harmonic form, with the material parameters varying slowly in the direction of propagation, and with the wave assumed to be paraxial, or confined to a relatively small angular area.

Alternatively, by assuming the time-harmonic form of a single mode waveguide whose transverse solution is known, a formulation known as the Nonlinear Schrödinger Equation (NLSE) can be derived [29]. The NLSE governs how a wave's amplitude $A(z,t)$ changes over time and distance when propagating in a nonlinear medium:

$$j\frac{\partial A}{\partial z} - \frac{1}{2}\beta_2 \frac{\partial^2 A}{\partial \tau^2} + \gamma |A|^2 A = 0 \tag{1.16}$$

in which β_2 is the Group Velocity Dispersion (GVD) (related to the linear dispersive term), τ represents a shifted time, γ is the nonlinear coefficient (related to the nonlinear term), and j is the imaginary unit. The NLSE represents one of the most important, popular, and widespread mathematical models for the inclusion of dispersion and nonlinearity in electromagnetic wave propagation and is ubiquitous in the field of nonlinear optics.

While simpler than solving the full nonlinear wave equation (where the full field solution must be found, as opposed to just its amplitude), techniques such as the NLSE nonetheless remain mathematically challenging, with exact solutions known only for a few special cases. As a result, a panoply of numerical methods for solving the NLSE have also been devised. In particular, the Split-Step Fourier method [33, 34] is among the most prominent, in which the linear and nonlinear components in (1.16) are independently and alternately advanced in time.

However, despite the wide-ranging success of these methods, they remain fundamentally built upon approximations to the underlying physical systems. It is thus imperative that prior to using a method such as BPM or NLSE that the assumptions underpinning these methods are verified to be true for the system being studied. Of course, it is nevertheless possible for a given system to deviate from the basic assumptions adopted in these methods. For example, multi-mode propagation, higher order nonlinearities, high order dispersion, quickly varying structures, quickly varying material parameters, and unguided or free propagation can all significantly hamper the accuracy of these methods, if not render them completely ineffective [27, 30, 35, 36].

For these reasons, often only a full solution of the non-approximative nonlinear wave equation in (1.13) or Maxwell's Equations in (1.12) can yield the complete and accurate behaviour of a given physical system. Further emphasizing this point, Joseph & Taflove notably found several instances in which the NLSE disagreed with a method which solved (1.12) and (1.13) directly [11]. It is thus essential that numerical methods be devised which are capable of solving (1.12) and (1.13) in full, directly in the time-domain, such as those

presented in the next section.

1.2.3 Numerical Methods for the Nonlinear Maxwell's Equations

Given some of the potential shortcomings identified with the approximative methods of the previous section, the best way to ensure that the full dispersive and nonlinear response of an electrically complex material is captured is via a method which solves Maxwell's Equations (or the vector wave equation) directly. With this in mind, a summary of current methods which have been adapted for dispersive and nonlinear materials will be presented. However, it is worth noting that many more techniques beyond those mentioned here have equally been devised for phenomena not considered in this thesis, such as anisotropy. For a more comprehensive review of such techniques, the reader is encouraged to consult the expansive review paper in [37].

One of the first algorithms adapted to solve equations (1.9) - (1.12) and (1.13) belonged to the Finite-Difference (FD) family of methods. Popular due to their accuracy and relative simplicity, Finite-Difference methods operate by discretizing the relevant differential operators via appropriate Taylor series expansions [38]. However, given the nonlinear nature of equation (1.8) linear superposition is no longer valid, precluding the use of time-harmonic analysis when multiple frequencies are present. Thus, in order to capture the full interaction of multiple frequency components, a method must be used which operates directly in the time-domain, meaning *both* spatial and temporal derivatives are discretized. As mentioned earlier, such a family of techniques known as Finite-Difference Time-Domain (FDTD) methods [19] had already been developed for broadband linear problems, with the most popular variant being the so-called "leap-frog" method, in which the coupled first-order Maxwell's Equations are solved by alternately advancing the electric and magnetic fields in time [39]. The FDTD method would thus form one of the starting points for treating complex media.

The first challenge in adapting FDTD methods to complex media came in including the effects of dispersion. In particular, over the years three distinct techniques have emerged

for the incorporation of frequency-dependent permittivities within time-domain methods: Recursive Convolution (RC), Auxiliary Differential Equation (ADE), and z-transform. Given that many polarization models can be described by constant-coefficient differential equations, it is often found that the permittivity is expressible as a rational function in the frequency-domain, a fact exploited by all three methods. Specifically, the Recursive Convolution (RC) method uses this observation to derive a recursion relation which permits the convolution between the permittivity and the electric field to be marched forward in time [40, 41]. Similarly, the Auxiliary Differential Equation (ADE) method uses the Fourier transform to convert the frequency-dependent rational permittivity back into an equivalent differential equation for the convolution [37, 42]. This auxiliary equation is then also discretized and solved along with the original, yielding the convolution value at each time step. Lastly, the z-transform method exploits the rational nature of the permittivity's frequency dependence, as well as the properties of the discrete z-transform, to yield a series of update equations for the convolution [43, 44]. While each of these methods has known notable success, the RC and ADE methods tend to produce formulations which rapidly increase in complexity and become intractable as the dispersive order of the materials is increased. In contrast, z-transform-based methods tend to produce relatively simpler update schemes, which have a tendency to remain so for higher dispersive orders [45].

As for the treatment of nonlinearity, several extensions to the base FDTD method were derived in which, generally speaking, nonlinear update equations were solved via iterative root finding methods [46, 47]. By converting back and forth between the $\vec{D}$ and $\vec{E}$ fields at each grid point, these methods were able to successfully capture the nonlinear polarization of the materials being studied, permitting the investigation of uniquely nonlinear phenomena. When combined with the dispersive theory described above, this led to exceedingly general FDTD methods capable of including each of the linearly dispersive, instantaneous nonlinear, and dispersive nonlinear terms present in (1.8) [27, 48, 49]. The generality and accuracy of these methods thus permitted additional insight to be gained into the behaviour

of wave propagation within nonlinear media, as well as the interactions between nonlinear and dispersive phenomena in general [11].

Despite the numerous successes of FDTD-based methods in capturing the full nonlinear, dispersive, and wide-band response of electrically complex materials within a single computation, there are nevertheless certain drawbacks associated with these methods. Difficulties in modeling curved interfaces, boundaries, and complex geometries have traditionally caused issues for FDTD methods, which generally operate on fixed regular grids, resulting in staircasing errors [50, 51]. Moreover, sharp material discontinuities or abrupt changes in material parameters can significantly weaken the accuracy of the underlying Taylor approximations, if not render them useless. Lastly, the most popular FDTD algorithms tend to be explicit in time and as a result may incur reduced stability [52, 53]. While mitigating strategies are available for each of these issues (such as parameter smoothing [54], Correction Function Methods [55], and implicit temporal discretizations [56]), they inevitably increase the complexity of the underlying method and may not be able to fully counterbalance these weaknesses.

An alternative to FDTD, which also operates in the time-domain and alleviates many of these concerns (at the cost of some added computational complexity), is the Finite-Element Time-Domain (FETD) method [20]. In many respects, FETD is very similar to FDTD (with some hybrid methods even having been developed [57]); however rather than discretizing the differential operators themselves, FETD discretizes the problem domain into sub-regions known as elements. The unknown field is then approximated as a linear combination of a set of known basis functions within each element, which are then stitched together via a global minimization to form the solution. Due to the nature of this spatial discretization or meshing of elements, FETD methods are more aptly suited for complex domain geometries, as well as material discontinuities [20].

Given the similarity between the two methods, it is unsurprising that attempts to generalize the original FETD schemes to complex media followed roughly the same process and

utilized many of the same techniques as in FDTD. For instance, the inclusion of dispersion within FETD has similarly been accomplished via Recursive Convolution [20, 58], Auxiliary Differential Equation [59, 60], and z-transform techniques [45, 61]. Unfortunately, however, progress in the development of nonlinear FETD methods for electrically complex media has not seen as widespread development, implementation, or documentation.

As mentioned previously, FETD methods have already been devised for problems containing complex magnetic materials exhibiting nonlinear permeability and hysteresis [62, 63]. These methods have been used extensively in the study of magnetic machines, such as electric motors; however the permittivity is generally assumed linear and constant. In addition, problems involving nonlinear conductivity and dielectric breakdown have also received attention, however here the nonlinearity is introduced via the conductivity σ and thus the permittivity and permeability are again both assumed linear and constant [64, 65].

In regards to an *instantaneous nonlinear* permittivity, the parametric quadratic programming method presented in [66] has been developed for the treatment of piecewise linear permittivity models. By approximating the permittivity in a piecewise linear fashion, the method can simulate instantaneous nonlinear phenomena, such as the Kerr effect, by transforming the equations into a complementarity problem, solvable by a number of optimization algorithms. At present, this is one of few FETD methods capable of including an instantaneous nonlinearity; however it remains limited due to the piecewise approximate nature of the permittivity.

In like manner, to date the treatment of *dispersive nonlinearities* within FETD has also received very little attention. In fact, the only method which currently addresses both instantaneous and dispersive nonlinearities within FETD is that presented by Fisher, White, and Rodrigue in [67]. Their method makes use of explicit time-stepping, the ADE formulation for dispersion, and hexahedral elements, resulting in an efficient, high-order, and general algorithm. Nevertheless, there remains some room for improvement. For instance, the use of the ADE technique leads to increased complexity as the dispersive order of the materials

is increased. Moreover, since explicit time-stepping is used, the scheme may prove to be less stable. Indeed, the authors reported that in many cases, time steps of at most half the Courant-Friedrichs-Lewy (CFL) condition were required for numerical stability. Lastly, the method's efficiency relies in part on the use of hexahedral elements to increase sparsity, rather than the more popular triangular and tetrahedral elements. It is some of these shortcomings, as well as the notable lack of general dispersive nonlinear methods in the literature, that this thesis aims to address.

1.3 Contributions and Overview

As detailed in the previous sections, there is a clear need for numerical methods capable of modeling complex material interactions. However, many of the existing methods either rely on approximations to the underlying governing equations, or impose geometric, stability, or accuracy constraints, limiting their use in many cases. The methods proposed in this thesis, in contrast, deal with electrically complex materials within the framework of FETD simulations in a versatile, accurate, and easily generalizable way. Both instantaneous and dispersive nonlinear algorithms are derived for various forms of electromagnetic FETD, offering hitherto unseen capabilities such as increased geometric freedom, triangular elements, increased scalability, ease of implementation, and significantly improved stability. The resulting family of methods is among the first to comprehensively model linear dispersion, instantaneous nonlinearity, and dispersive nonlinearities, all within a single FETD simulation. Moreover, this thesis not only derives these algorithms, but also proposes additional currently nonexistent tools and techniques to make their application more efficient and useful. For instance, for the first time, a perfectly matched layer for dispersive nonlinear FETD is presented, permitting the emulation of a wider class of problems, including those with infinite or semi-infinite domains.

The resulting family of methods are also analyzed in terms of efficiency and imple-

mentability. Bottlenecks in the execution of the derived algorithms are identified, and attempts made to minimize their impact. Specifically, given the computational burden generally imposed by these materials, this thesis also analyzes the use of massively parallel architectures, such as Graphics Processing Units (GPUs), to accelerate their execution. Thus, not only do the proposed algorithms offer new capabilities, but do so at a substantially reduced computational cost.

The following presents an overview of the various chapters and topics covered in this thesis:

Chapter 1 - Provided an overview of the motivations, applications, theory, and literature associated with electrically complex materials and their simulation.

Chapter 2 - Derives a family of dispersive nonlinear FETD methods for the treatment of electrically complex materials.

Chapter 3 - Discusses details regarding the implementation and characteristics of the algorithms presented in the previous chapter, including numerical characteristics, optimizations, and stability.

Chapter 4 - Reviews the theory and practice of Perfectly Matched Layers (PML) and derives an extension of the PML to nonlinear dispersive FETD.

Chapter 5 - Analyzes how parallelism can be used to minimize the computational burden imposed by electrically complex materials, with emphasis on implementation on GPUs.

Chapter 6 - Provides a concluding overview of the salient details and contributions of this thesis, as well as some suggestions for future areas of study.

Lastly, the author of this thesis is the principle contributor of all its contents. This work therefore reflects the author's novel contributions to the field, with any co-authors reported in the List of Publications acting in a supervisory capacity.

Chapter 2

FETD Methods for Nonlinear Dispersive Media

2.1 Introduction

In this chapter, a novel family of FETD algorithms will be derived for the treatment of linear dispersion, instantaneous nonlinearity, and dispersive nonlinearity in electromagnetic wave simulations. In particular, these algorithms will belong to one of two main variants of the FETD method commonly in use: the mixed formulation and the Vector Wave Equation (VWE) formulation.

Mixed FETD formulations solve the coupled first-order Maxwell's Equations, obtaining both the electric and magnetic field at each time step [37, 68]. In contrast, VWE-based methods solve the second-order vector wave equation, obtaining just the electric field at each time step [20]. Even though VWE formulations are more popular due to the simplicity of having a single unknown working variable of interest, mixed approaches do have some advantages. For instance, it can be shown that the discretized VWE supports non-trivial

and potentially non-physical solutions of the form [69]:

$$\vec{E}(\vec{x},t) = -(at+b)\nabla\phi \tag{2.1}$$

where a and b are constants and ϕ is a time-independent scalar function. In theory, a and b can be set to zero via appropriate boundary conditions, however due to finite machine precision this cannot be guaranteed in practice [45]. As a result, the term proportional to a can cause the solution to drift linearly in time, resulting in so-called late-time growth [70]. Thus, if very long simulation times are required, mitigating strategies need to be adopted for VWE-based methods, whereas mixed methods do not suffer this issue.

Due to the relative strengths and weaknesses of each method, both will be developed in the following sections, allowing the freedom to choose the technique which best matches the requirements of the problem under study. While the two methods do have some notable differences, it is equally important to note that they also share a very similar underlying structure. In fact, when appropriate temporal discretizations are applied to each method, they can be shown to be homologous, meaning many of the fundamental matrices and update equations are identical. In consequence, by starting with the derivation of the mixed method, the VWE version will be shown to follow logically, with many of the derived expressions carrying over unchanged.

In the following section the traditional mixed FETD method will first be reviewed for regular linear non-dispersive media. This will then be followed by a review of the inclusion of linear dispersion via the z-transform method. With this underlying theory in hand, the novel instantaneous and dispersive nonlinear methods will then be derived. Lastly, knowledge of the mixed algorithms will then be leveraged to derive the equivalent new VWE formulations.

2.2 Mixed FETD

2.2.1 Linear Non-Dispersive Media

In this section, the fundamentals of the mixed FETD method will be presented for linear non-dispersive media, to serve as a base for the derivations to follow. To begin, Faraday and Ampère's Laws are expressed for *linear non-dispersive* media as follows [2]:

$$\nabla \times \vec{E} = -\frac{\partial \vec{B}}{\partial t} \tag{2.2}$$

$$\nabla \times \frac{\vec{B}}{\mu} = \vec{J}_v + \epsilon \frac{\partial \vec{E}}{\partial t}. \tag{2.3}$$

The Finite-Element method proceeds by discretizing or meshing the problem domain into sub-domains known as elements. These elements are most commonly simplicial, resulting in a triangular meshing in two dimensions and a tetrahedral meshing in three. The key idea is to then choose a set of interpolary vector basis functions within each element, such that the unknown solution of interest may be represented as linear combinations of these functions within each triangle or tetrahedron [20]. In the case of a mixed method, such expressions are required for both the electric and magnetic fields:

$$\vec{E} = \sum_{i=1}^{l_e} e_i^{(e)} \vec{W}_i^{(1)(e)} \tag{2.4}$$

$$\vec{B} = \sum_{j=1}^{m_e} b_j^{(e)} \vec{W}_j^{(2)(e)} \tag{2.5}$$

where $e_i^{(e)}$ and $b_j^{(e)}$ are the time-dependent interpolation weights and $\vec{W}_i^{(1)(e)}$ and $\vec{W}_j^{(2)(e)}$ are, respectively, the time-independent electric and magnetic basis functions within the element (e).

The choice of basis functions in (2.4) and (2.5) is an important one, as many of the properties of the resulting method will be dictated by the class of functions used. For

instance, it is crucial that the basis functions be constructed in such a way as to guarantee inter-element continuity of the fields, as well as satisfy the requisite interface conditions. More explicitly, this means ensuring the continuity of the tangential component of the electric field and the normal component of the magnetic flux density across element boundaries [20]. Equally important, however, are the divergence and curl characteristics of the basis functions. In particular, failing to enforce the divergenceless property of the fields in charge-free regions can lead to corrupting artifacts known as spurious solutions [71].

The most common choice of basis functions, which both enforce continuity and are divergenceless, are the Whitney forms [72, 73, 74]. Whitney 1-forms, also known as edge elements, are used to represent the electric field. These edge elements are so named since the degrees of freedom $e_i^{(e)}$ are associated with the edges of the simplices. More specifically, in the case of mixed FETD, the basis functions are unnormalized such that the weights $e_i^{(e)}$ represent the tangential component of the field on that edge, scaled by the edge length, whereas in the VWE FETD to follow, they are normalized such that the weights represent the tangential field values themselves. Similarly, Whitney 2-forms, also known as face elements, are used to represent the magnetic flux density. The face elements are so named since the degrees of freedom $b_j^{(e)}$ are this time associated with the faces of the simplices, representing the normal field component scaled by the face area.

Having selected appropriate basis functions, attention is now turned toward discretizing equations (2.2) and (2.3). For Faraday's Law, the expansions in (2.4) and (2.5) can be substituted in directly, yielding:

$$\nabla \times \sum_{i=1}^{l_e} e_i^{(e)} \vec{W}_i^{(1)(e)} = -\frac{\partial}{\partial t} \sum_{j=1}^{m_e} b_j^{(e)} \vec{W}_j^{(2)(e)}. \tag{2.6}$$

However, one additional benefit of these particular basis functions is that the curl of the Whitney 1-forms is a subset of the 2-forms. In other words, the curl of any given 1-form within an element can be written as a linear combination of that element's 2-forms, in which

the weights are either $+1$, -1, or 0 [75]. Denoting the coefficient of the j^{th} 2-form associated with the curl of the i^{th} 1-form within an element (e) as $C_{ij}^{(e)}$, this can be written as:

$$\nabla \times \vec{W}_i^{(1)(e)} = \sum_{j=1}^{m_e} C_{ij}^{(e)} \vec{W}_j^{(2)(e)} \tag{2.7}$$

which when substituted into equation (2.6) yields:

$$\sum_{i=1}^{l_e} \sum_{j=1}^{m_e} e_i^{(e)} C_{ij}^{(e)} \vec{W}_j^{(2)(e)} = -\frac{\partial}{\partial t} \sum_{j=1}^{m_e} b_j^{(e)} \vec{W}_j^{(2)(e)}. \tag{2.8}$$

By performing a term-by-term comparison of the 2-form coefficients in (2.8), a relationship between the electric and magnetic weights can be established:

$$\sum_{i=1}^{l_e} C_{ij}^{(e)} e_i^{(e)} = -\frac{\partial b_j^{(e)}}{\partial t} \tag{2.9}$$

which can be more succinctly written by making use of matrix-vector notation:

$$[C^{(e)}]^T \{e^{(e)}\} = -\frac{\partial \{b^{(e)}\}}{\partial t} \tag{2.10}$$

where in this thesis, braces $\{\cdot\}$ will denote a column vector, square brackets $[\cdot]$ will denote a matrix, and a superscript T denotes the transpose. Equation (2.10) now represents a semi-discrete version of Faraday's Law within each element.

As for the discretization of Ampère's Law, the procedure is unfortunately not as straightforward. Here, a Galerkin procedure [20] is adopted in which the dot product of equation (2.3) is taken with the 1-form basis functions, and the result integrated over each element:

$$\int_{\Omega^e} \left(\nabla \times \frac{\vec{B}}{\mu} \right) \cdot \vec{W}_j^{(1)(e)} d\Omega = \int_{\Omega^e} \vec{J}_v \cdot \vec{W}_j^{(1)(e)} d\Omega + \int_{\Omega^e} \epsilon \frac{\partial \vec{E}}{\partial t} \cdot \vec{W}_j^{(1)(e)} d\Omega. \tag{2.11}$$

Applying the vector identity

$$\vec{B} \cdot \nabla \times \vec{A} = \nabla \cdot (\vec{A} \times \vec{B}) + \vec{A} \cdot \nabla \times \vec{B} \tag{2.12}$$

to the term on the left-hand side, equation (2.11) can also be expressed as:

$$\int_{\Omega^e} \nabla \cdot \left(\frac{\vec{B}}{\mu} \times \vec{W}_j^{(1)(e)} \right) d\Omega + \int_{\Omega^e} \frac{\vec{B}}{\mu} \cdot \left(\nabla \times \vec{W}_j^{(1)(e)} \right) d\Omega = \int_{\Omega^e} \vec{J}_v \cdot \vec{W}_j^{(1)(e)} d\Omega + \int_{\Omega^e} \epsilon \frac{\partial \vec{E}}{\partial t} \cdot \vec{W}_j^{(1)(e)} d\Omega. \tag{2.13}$$

Lastly, this can then be further refined by an application of the divergence theorem to the first term on the left-hand side, from which the final expression is obtained:

$$\int_{\Omega^e} \frac{\vec{B}}{\mu} \cdot \left(\nabla \times \vec{W}_j^{(1)(e)} \right) d\Omega = \int_{\Omega^e} \vec{J}_v \cdot \vec{W}_j^{(1)(e)} d\Omega + \int_{\Omega^e} \epsilon \frac{\partial \vec{E}}{\partial t} \cdot \vec{W}_j^{(1)(e)} d\Omega - \int_{\partial\Omega^e} \frac{1}{\mu} \left(\vec{B} \times \vec{W}_j^{(1)(e)} \right) \cdot d\vec{S}. \tag{2.14}$$

The first and third integrals on the right-hand side of equation (2.14) essentially act as source terms, with the first representing a volume current source and the third being useful for the imposition of special boundary conditions on the element boundary $\partial\Omega$. Moreover, both sources may be combined and written more compactly by defining an elemental source vector $\{f^{(e)}\}$, whose j^{th} component is given by:

$$\{f^{(e)}\}_j = \int_{\Omega^e} \vec{J}_v \cdot \vec{W}_j^{(1)(e)} d\Omega - \int_{\partial\Omega^e} \frac{1}{\mu} \left(\vec{B} \times \vec{W}_j^{(1)(e)} \right) \cdot d\vec{S} \tag{2.15}$$

such that equation (2.14) can be more simply expressed as:

$$\int_{\Omega^e} \frac{\vec{B}}{\mu} \cdot \left(\nabla \times \vec{W}_j^{(1)(e)} \right) d\Omega = \int_{\Omega^e} \epsilon \frac{\partial \vec{E}}{\partial t} \cdot \vec{W}_j^{(1)(e)} d\Omega + \{f^{(e)}\}_j. \tag{2.16}$$

As for the remaining two terms in (2.16), the expansions in (2.4) and (2.5) can be substituted in place of the electric and magnetic fields. For the magnetic term on the left-hand side of (2.16), this results in:

$$\int_{\Omega^e} \frac{\vec{B}}{\mu} \cdot \left(\nabla \times \vec{W}_j^{(1)(e)} \right) d\Omega = \int_{\Omega^e} \frac{1}{\mu} \sum_{i=1}^{m_e} b_i^{(e)} \vec{W}_i^{(2)(e)} \cdot \left(\nabla \times \vec{W}_j^{(1)(e)} \right) d\Omega \tag{2.17}$$

$$= \sum_{i=1}^{m_e} \sum_{k=1}^{m_e} \int_{\Omega^e} \frac{1}{\mu} b_i^{(e)} \vec{W}_i^{(2)(e)} \cdot C_{jk}^{(e)} \vec{W}_k^{(2)(e)} d\Omega \tag{2.18}$$

where the curl property of equation (2.7) has been used in (2.18). This equation may also be simplified by defining a new $[M_f^{(e)}]$ matrix, whose entry at the i^{th} row and j^{th} column is given by:

$$[M_f^{(e)}]_{ij} = \int_{\Omega^e} \frac{1}{\mu} \vec{W}_i^{(2)(e)} \cdot \vec{W}_j^{(2)(e)} d\Omega \tag{2.19}$$

from which (2.18) becomes:

$$\int_{\Omega^e} \frac{\vec{B}}{\mu} \cdot \left(\nabla \times \vec{W}_j^{(1)(e)} \right) d\Omega = \left([C^{(e)}][M_f^{(e)}]\{b^{(e)}\} \right)_j . \tag{2.20}$$

Following a similar procedure for the electric field term results in:

$$\int_{\Omega^e} \epsilon \frac{\partial \vec{E}}{\partial t} \cdot \vec{W}_j^{(1)(e)} d\Omega = \sum_{i=1}^{l_e} \int_{\Omega^e} \epsilon \frac{\partial e_i^{(e)}}{\partial t} \vec{W}_i^{(1)(e)} \cdot \vec{W}_j^{(1)(e)} d\Omega \tag{2.21}$$

for which if a new $[M^{(e)}]$ matrix is defined:

$$[M^{(e)}]_{ij} = \int_{\Omega^e} \epsilon \, \vec{W}_i^{(1)(e)} \cdot \vec{W}_j^{(1)(e)} d\Omega \tag{2.22}$$

can instead be written as:

$$\int_{\Omega^e} \epsilon \frac{\partial \vec{E}}{\partial t} \cdot \vec{W}_j^{(1)(e)} d\Omega = \left([M^{(e)}] \frac{\partial \{e^{(e)}\}}{\partial t} \right)_j . \tag{2.23}$$

Combining all of these terms, as well as the previously derived semi-discrete Faraday's Law, the spatially discretized Maxwell's Equations are at last obtained in matrix form:

$$[C^{(e)}]^T\{e^{(e)}\} = -\frac{\partial\{b^{(e)}\}}{\partial t} \tag{2.24}$$

$$[C^{(e)}][M_f^{(e)}]\{b^{(e)}\} = [M^{(e)}]\frac{\partial\{e^{(e)}\}}{\partial t} + \{f^{(e)}\}. \tag{2.25}$$

Recall, however, that so far the above equations have only been defined within each individual element, as indicated by the superscript (e). In order to obtain the global solution, these elemental matrices and equations must be added and combined into equivalent global matrices according to their connectivity:

$$[M] = \sum_{(e)}[M'^{(e)}] \qquad [M_f] = \sum_{(e)}[M_f'^{(e)}] \qquad [C] = \sum_{(e)}[C'^{(e)}] \qquad \{f\} = \sum_{(e)}\{f'^{(e)}\} \tag{2.26}$$

where terms without a superscript (e) represent global quantities and a prime denotes the item has been expanded according to the global variable numbering (for a more detailed description of the matrix assembly process, see Appendix A). The resulting *global* matrix equations are identical to those in (2.24) and (2.25):

$$[C]^T\{e\} = -\frac{\partial\{b\}}{\partial t} \tag{2.27}$$

$$[C][M_f]\{b\} = [M]\frac{\partial\{e\}}{\partial t} + \{f\} \tag{2.28}$$

except the global matrices themselves are now large and sparse. In particular, the dimensions of these global matrices and vectors are given in Table 2.1.

At this point the equations have been spatially discretized, however they are still continuous in time. As a result, a temporal discretization must now be applied such that the solution can be found at discrete time steps. While many such temporal discretizations are possible, one popular variant uses central differences to approximate the temporal deriva-

Table 2.1: Semi-discrete Maxwell's Equations matrices and vectors.

Component	Dimension	Sparsity
$[M]$	$l \times l$	Sparse
$[M_f]$	$m \times m$	Sparse
$[C]$	$l \times m$	Sparse
$\{e\}$	$l \times 1$	Dense
$\{b\}$	$m \times 1$	Dense

[1] l is the number of distinct edges in the global mesh
[2] m is the number of distinct faces in the global mesh

tives, yielding an explicit algorithm very similar to the FDTD method. While such a method could easily be used here, the resulting algorithm is unfortunately only conditionally stable [68]. Alternatively, another possibility is the use of the Crank-Nicolson method [76], which is not only implicit but has recently been posited to be unconditionally stable when used in linear mixed FETD [45, 77].

Given a first-order ordinary differential equation such as:

$$\frac{du}{dt} = f(u,t) \tag{2.29}$$

the Crank-Nicolson method approximates the temporal derivative via a finite-difference formula:

$$\frac{du}{dt} \approx \frac{u^{n+1} - u^n}{\Delta t} \tag{2.30}$$

where superscripts denote the discrete temporal step number and Δt is the size of the discretized time step, such that $t = n\Delta t$. Moreover, rather than equating this approximation to the source term at time n (which would result in forward Euler) or time $n+1$ (which would result in backward Euler), Crank-Nicolson uses a combination of the two, such that

the discretized form of equation (2.29) is [20]:

$$\frac{u^{n+1}-u^n}{\Delta t}=\frac{1}{2}\left(f^n+f^{n+1}\right). \tag{2.31}$$

Applying this to the spatially discretized form of Faraday and Ampère's Laws in (2.27) and (2.28) results in:

$$\frac{\{b\}^{n+1}-\{b\}^n}{\Delta t}=-\frac{1}{2}[C]^T\Big(\{e\}^n+\{e\}^{n+1}\Big) \tag{2.32}$$

$$[M]\frac{\{e\}^{n+1}-\{e\}^n}{\Delta t}=\frac{1}{2}[C][M_f]\Big(\{b\}^n+\{b\}^{n+1}\Big)+\frac{1}{2}\Big(f^n+f^{n+1}\Big). \tag{2.33}$$

By using (2.32) to eliminate $\{b\}^{n+1}$ from (2.33), two fully discretized update equations can be obtained for the electric and magnetic fields:

$$\left([M]+\frac{\Delta t^2}{4}[C][M_f][C]^T\right)\{e\}^{n+1}=\left([M]-\frac{\Delta t^2}{4}[C][M_f][C]^T\right)\{e\}^n + \Delta t[C][M_f]\{b\}^n+\frac{\Delta t}{2}\Big(f^n+f^{n+1}\Big) \tag{2.34}$$

$$\{b\}^{n+1}=\{b\}^n-\frac{\Delta t}{2}[C]^T\Big(\{e\}^n+\{e\}^{n+1}\Big). \tag{2.35}$$

With these two update equations in hand, the solution procedure in traditional linear non-dispersive media is thus as follows:

1. Using (2.34) and known quantities from the current and previous time steps, solve for $\{e\}^{n+1}$.

2. With $\{e\}^{n+1}$ known, use equation (2.35) to solve for $\{b\}^{n+1}$.

3. Repeat until the desired end time.

2.2.2 Dispersive Media

As detailed earlier in Subsection 1.2.3 a well-developed literature exists surrounding the treatment of *linear* dispersion within computational electromagnetics. Recursive Convolution, Auxiliary Differential Equation, and z-transform techniques have all been developed for FETD-based methods with much success. However, as has been noted, the z-transform approach in particular tends to have a simpler overall implementation [45]. In particular, when dealing with higher-order dispersion, the convolution expressions in RC and the differential equations in ADE can become very large intractable expressions, hampering their effectiveness and ease of use. The z-transform update equations, in contrast, remain relatively straightforward to implement. Moreover, given the importance of the z-transform in signal processing, implementing dispersion in this way can be facilitated by drawing on the immense existing signal processing literature. As a result, this thesis will exclusively employ the z-transform method to model dispersion in all of the algorithms to come. In view of this, an overview of the dispersive z-transform theory originally derived in [45] and [61] will now be presented.

As was outlined in Subsection 1.2.1, having an electrically dispersive medium implies that the permittivity is now a function of time, requiring the computation of a convolution:

$$\vec{D}(t) = \epsilon_0 \vec{E}(t) + \epsilon_0 \chi^{(1)}(t) * \vec{E}(t) \tag{2.36}$$

$$= \epsilon_0 \Big(\delta(t) + \chi^{(1)}(t) \Big) * \vec{E}(t) \tag{2.37}$$

$$= \epsilon_L(t) * \vec{E}(t) \tag{2.38}$$

where $\delta(t)$ is the Dirac delta function. Under these conditions, Faraday's Law is unchanged, however Ampère's Law becomes:

$$\nabla \times \frac{\vec{B}}{\mu} = \vec{J}_v + \frac{\partial}{\partial t} \left(\epsilon_L * \vec{E} \right). \tag{2.39}$$

Assuming that the permittivity is constant in space within each element, the exact same spatial discretization procedure as in the last section can be applied and yields a similar semi-discrete system of equations:

$$[C]^T\{e\} = -\frac{\partial\{b\}}{\partial t} \tag{2.40}$$

$$[C][M_f]\{b\} = [\tilde{M}]\frac{\partial}{\partial t}\Big(\epsilon_L * \{e\}\Big) + \{f\} \tag{2.41}$$

where the $[\tilde{M}]$ matrix differs from the previous $[M]$ matrix in that the permittivity is absent:

$$[\tilde{M}^{(e)}]_{ij} = \int_{\Omega^e} \vec{W}_i^{(1)(e)} \cdot \vec{W}_j^{(1)(e)} d\Omega. \tag{2.42}$$

In applying the Crank-Nicolson method to equations (2.40) and (2.41) it is evident that the value of the convolution:

$$\{\mathcal{L}\}(t) \triangleq \epsilon_L(t) * [\tilde{M}]\{e\}(t) \tag{2.43}$$

will be required at times n and $n+1$. It will thus be necessary to derive an update equation for the convolution, such that it can be advanced in time along with the solution vectors.

To alleviate this burden, the z-transform approach will be adopted in which the convolution is instead converted into a series of multiplications. To that end, the Laplace transform of equation (2.43) can be taken, which due to the convolution property, is converted into a multiplication in the s-domain [22]:

$$\mathcal{L}(s) = \epsilon_L(s)[\tilde{M}]\{e\}(s). \tag{2.44}$$

However, as mentioned previously, due to the constraints imposed by causality and the Kramers-Kronig relations, the permittivity cannot be an arbitrary function of frequency. Moreover, for many materials the polarization can be modeled by a constant coefficient

differential equation, and so it is commonly found that the permittivity is expressible as the quotient of two s-dependent polynomials in the Laplace domain:

$$\epsilon_L(s) = \frac{c_p s^p + \ldots + c_0}{d_p s^p + \ldots + d_0} \tag{2.45}$$

where p represents the dispersive order of the medium. This rational permittivity is thus either the result of a theoretical model, such as in Lorentz and Debye dispersion [19], or else of curve fitting of empirical permittivity data by rational functions, such as in the well-known Sellmeier equation [78]. As a result, for most dispersive materials of practical interest, the permittivity may be expressed as in equation (2.45), such that (2.44) becomes:

$$\{\mathcal{L}\}(s) = \frac{c_p s^p + \ldots + c_0}{d_p s^p + \ldots + d_0}[\tilde{M}]\{e\}(s). \tag{2.46}$$

The goal now is to move from the continuous frequency variable s to the discretized frequency space of the z-transform. If the Laplace transform of a discrete-time signal sampled at intervals $t = n\Delta t$ is taken, the result can be shown to be equivalent to the evaluation of the z-transform of the signal if [22]:

$$z = e^{s\Delta t} \tag{2.47}$$

or alternatively:

$$s = \frac{1}{\Delta t}\ln(z). \tag{2.48}$$

By expanding the natural logarithm in (2.48) in terms of a bilinear series, an approximation to this mapping can be obtained in terms of a Möbius transform [79]:

$$s = \frac{1}{\Delta t}\ln(z) \tag{2.49}$$

$$= \frac{2}{\Delta t}\left(\frac{z-1}{z+1} + \frac{1}{3}\left(\frac{z-1}{z+1}\right)^3 + \frac{1}{5}\left(\frac{z-1}{z+1}\right)^5 + \cdots\right) \tag{2.50}$$

$$\approx \frac{2}{\Delta t}\frac{z-1}{z+1} \tag{2.51}$$

$$= \frac{2}{\Delta t}\frac{1-z^{-1}}{1+z^{-1}}. \tag{2.52}$$

This approximation thus permits the mapping from the continuous s-domain to the discrete z-domain, while importantly preserving the rational nature of the permittivity.

Applying this transformation to (2.46), normalizing the coefficients by the first term in the denominator of the permittivity, and cross multiplying yields:

$$(1 + \ldots + b_p z^{-p})\{\mathcal{L}\}(z) = (a_0 + \ldots + a_p z^{-p})[\tilde{M}]\{e\}(z). \tag{2.53}$$

Lastly, by leveraging the time shifting property of the inverse z-transform [22], (2.53) can be transformed back to the time-domain, yielding an update equation for the convolution:

$$\{\mathcal{L}\}^n = a_0[\tilde{M}]\{e\}^n + \ldots + a_p[\tilde{M}]\{e\}^{n-p} - b_1\{\mathcal{L}\}^{n-1} - \ldots - b_p\{\mathcal{L}\}^{n-p}. \tag{2.54}$$

While equation (2.54) can be straightforwardly implemented as shown, more efficient update strategies are possible. Indeed, as mentioned earlier, one of the key advantages of the z-transform method is the ability to draw upon the signal processing literature. In particular, rather than using the explicit update equation in (2.54), which requires $2p + 1$ past field and convolution values to be stored, these values can instead be accumulated into auxiliary variables as time stepping progresses. More specifically, by using the Transposed Direct Form II implementation, as originally suggested in [45, 61], the update equation for

the convolution can alternatively be expressed as:

$$\{\mathcal{W}_\alpha\}^n = a_\alpha[\tilde{M}]\{e\}^n - b_\alpha\{\mathcal{L}\}^n + \{\mathcal{W}_{\alpha+1}\}^{n-1} \qquad \alpha < p \tag{2.55}$$

$$\{\mathcal{W}_\alpha\}^n = a_\alpha[\tilde{M}]\{e\}^n - b_\alpha\{\mathcal{L}\}^n \qquad \alpha = p \tag{2.56}$$

$$\{\mathcal{L}\}^n = a_0[\tilde{M}]\{e\}^n + \{\mathcal{W}_1\}^{n-1} \tag{2.57}$$

where $\{\mathcal{W}_\alpha\}$ are the auxiliary variables in question, of which only p are needed.

The expression for the convolution in (2.57) can now be inserted into the temporal discretization obtained via Crank-Nicolson, resulting in a modified form of the update equation for the electric field in (2.34):

$$\begin{aligned}\left(a_0[\tilde{M}] + \frac{\Delta t^2}{4}[C][M_f][C]^T\right)\{e\}^{n+1} = \\ \left(a_0[\tilde{M}] - \frac{\Delta t^2}{4}[C][M_f][C]^T\right)\{e\}^n + \Delta t[C][M_f]\{b\}^n \\ - \{\mathcal{W}_1\}^n + \{\mathcal{W}_1\}^{n-1} - \frac{\Delta t}{2}\Big(\{f\}^n + \{f\}^{n+1}\Big).\end{aligned} \tag{2.58}$$

The solution procedure for linear dispersive media is now quite similar to before, with the exception of the need to update the auxiliary variables at each time step:

1. Using (2.58) and all known values from previous time steps, obtain $\{e\}^{n+1}$.
2. Using this result, obtain $\{b\}^{n+1}$ using equation (2.35), which remains unchanged from the non-dispersive case.
3. Advance each of the auxiliary variables and the convolution to $n+1$ in order, using equations (2.55) - (2.57).
4. Repeat until the desired end time.

2.2.3 Instantaneous Nonlinearity

With a review of linear dispersive theory complete, attention is now turned toward one of the first key contributions of this thesis, the development of implicit mixed nonlinear FETD methods for electrically complex media. Indeed, as mentioned earlier, to date the only existing mixed nonlinear FETD algorithm [67] is *explicit* in time, and as such the algorithms derived from this point forward represent the first time instantaneous and dispersive nonlinearity has been incorporated within *implicit* mixed FETD methods. As detailed further in Chapter 3, this will have important consequences for the stability of the method. Here, the linear mixed FETD theory of Subsection 2.2.1 will be modified to accommodate an instantaneous nonlinearity, with the further generalization to dispersive nonlinearities provided in the following subsection.

In the case of an instantaneous nonlinearity convolutions are absent, however the permittivity now depends on the electric field strength ($\alpha = 1$):

$$\vec{D} = \underbrace{\epsilon_0(1 + \chi^{(1)} + \chi^{(3)}E^2)}_{\epsilon}\vec{E}. \tag{2.59}$$

Applying a spatial discretization procedure to the now nonlinear Ampère's Law proceeds in much the same way as it did in Subsection 2.2.1. The main exception of course is that the corresponding $[M]$ matrix is no longer constant due to its dependence on the electric field magnitude E:

$$[M^{(e)}]_{ij} = \int_{\Omega^e} \epsilon(E)\vec{W}_i^{(1)(e)} \cdot \vec{W}_j^{(1)(e)} d\Omega \tag{2.60}$$

$$= \int_{\Omega^e} \epsilon_0(1 + \chi^{(1)} + \chi^{(3)}E^2)\vec{W}_i^{(1)(e)} \cdot \vec{W}_j^{(1)(e)} d\Omega \tag{2.61}$$

and thus must remain within the temporal derivative:

$$[C]^T\{e\} = -\frac{\partial\{b\}}{\partial t} \tag{2.62}$$

$$[C][M_f]\{b\} = \frac{\partial}{\partial t}\Big([M]\{e\}\Big) + \{f\}. \tag{2.63}$$

As a result, the Crank-Nicolson temporal discretization must now be applied to the entire matrix-vector product, such that the temporal derivative in (2.63) is instead approximated as:

$$\frac{\partial}{\partial t}\Big([M]\{e\}\Big) \approx \frac{[M]^{n+1}\{e\}^{n+1} - [M]^n\{e\}^n}{\Delta t}. \tag{2.64}$$

Keeping this in mind, the derivation of the required update equations for the $\vec{E}$ and $\vec{B}$ fields follows a similar procedure to that presented previously, yielding the following:

$$\left([M]^{n+1} + \frac{\Delta t^2}{4}[C][M_f][C]^T\right)\{e\}^{n+1} = \left([M]^n - \frac{\Delta t^2}{4}[C][M_f][C]^T\right)\{e\}^n + \Delta t[C][M_f]\{b\}^n + \frac{\Delta t}{2}\Big(\{f\}^n + \{f\}^{n+1}\Big) \tag{2.65}$$

$$\{b\}^{n+1} = \{b\}^n - \frac{\Delta t}{2}[C]^T\Big(\{e\}^n + \{e\}^{n+1}\Big). \tag{2.66}$$

While deceptively similar to the update equations derived for linear media, the solution of the above system of equations is manifestly more complex due to the dependence of the $[M]$ matrix on the solution vector $\{e\}$. As a result, the above system must be solved using an iterative root finding method, rather than a straightforward linear solver. While many techniques exist for solving such systems, here the Newton-Raphson technique will be adopted due to its ubiquity, seminal nature, and fast rate of convergence [80].

In the Newton-Raphson method, a system of nonlinear equations of the form $\{F\} = 0$, dependent on a set of unknown variables $\{x\}$, may be solved by iterating in the following manner:

$$\{x\}_{(k+1)} = \{x\}_{(k)} - [J]^{-1}_{(k)}\{F\}_{(k)} \tag{2.67}$$

in which the bracketed subscript (k) is the iteration number, and $[J]$ is the Jacobian matrix defined as:

$$[J]_{ij} = \frac{\partial \{F\}_i}{\partial \{x\}_j}. \tag{2.68}$$

By moving all terms of equation (2.65) to one side, the Newton-Raphson method can easily be applied, assuming the Jacobian can be found. Thus, attention is now turned toward determining an appropriate expression for the Jacobian associated with (2.65). Notably, the derivation of the Jacobian to follow is similar to that derived in [62] for nonlinear magnetic materials in the VWE formulation, and produces comparable expressions.

In equation (2.65) the unknown variable of interest is the vector $\{e\}^{n+1}$, and so the Jacobian will only take derivatives with respect to this quantity. As a result, only the left-hand side of (2.65) will have a non-zero contribution to the Jacobian. To facilitate the analysis further, the matrix-vector product on the left-hand side of equation (2.65) is explicitly expressed as a sum, whose derivative with respect to $\{e\}_j^{n+1}$ is subsequently taken:

$$[J]_{ij}^{n+1} = \frac{\partial \{F\}_i}{\partial \{e\}_j^{n+1}} = \frac{\partial}{\partial \{e\}_j^{n+1}} \sum_{k=1}^{l} \left([M]_{ik}^{n+1} + \frac{\Delta t^2}{4} [C][M_f][C]_{ik}^T \right) \{e\}_k^{n+1}. \tag{2.69}$$

Distributing the derivative into the sum and applying the product rule results in:

$$[J]_{ij}^{n+1} = \sum_{k=1}^{l} \left([M]_{ik}^{n+1} + \frac{\Delta t^2}{4} [C][M_f][C]_{ik}^T \right) \frac{\partial \{e\}_k^{n+1}}{\partial \{e\}_j^{n+1}} + \sum_{k=1}^{l} \frac{\partial}{\partial \{e\}_j^{n+1}} \left([M]_{ik}^{n+1} + \frac{\Delta t^2}{4} [C][M_f][C]_{ik}^T \right) \{e\}_k^{n+1}. \tag{2.70}$$

Due to the independence of each of the unknown variables, it can be shown that:

$$\frac{\partial \{e\}_k^{n+1}}{\partial \{e\}_j^{n+1}} = \begin{cases} 1 & \text{for } j = k \\ 0 & \text{for } j \neq k \end{cases} \tag{2.71}$$

which indicates that the only contribution from the first term will occur when $j = k$. More-

over, since the $[C]$ and $[M_f]$ matrices are constant, their derivatives vanish. With these two results, (2.70) simplifies considerably:

$$[J]_{ij}^{n+1} = [M]_{ij}^{n+1} + \frac{\Delta t^2}{4}[C][M_f][C]_{ij}^{T} + \sum_{k=1}^{l} \frac{\partial [M]_{ik}^{n+1}}{\partial \{e\}_j^{n+1}} \{e\}_k^{n+1}. \tag{2.72}$$

The last term in (2.72) can be further refined by returning to the definition of the elemental $[M^{(e)}]$ matrices from which the global $[M]$ matrix is built, and bringing the derivative, sum, and unknown vector inside the integral:

$$\sum_{k=1}^{l_e} \frac{\partial [M^{(e)}]_{ik}^{n+1}}{\partial \{e^{(e)}\}_j^{n+1}} \{e^{(e)}\}_k^{n+1} = \int_{\Omega^e} \frac{\partial \epsilon^{n+1}}{\partial \{e^{(e)}\}_j^{n+1}} \vec{W}_i^{(1)(e)} \cdot \sum_{k=1}^{l_e} \vec{W}_k^{(1)(e)} \{e^{(e)}\}_k^{n+1} d\Omega \tag{2.73}$$

where it is noticed that the sum in the above is exactly the interpolated representation of the electric field within each element from (2.4):

$$\sum_{k=1}^{l_e} \frac{\partial [M^{(e)}]_{ik}^{n+1}}{\partial \{e^{(e)}\}_j^{n+1}} \{e^{(e)}\}_k^{n+1} = \int_{\Omega^e} \frac{\partial \epsilon^{n+1}}{\partial \{e^{(e)}\}_j^{n+1}} \vec{W}_i^{(1)(e)} \cdot \vec{E}^{n+1} d\Omega. \tag{2.74}$$

Furthermore, using the chain rule, the derivative of the permittivity may be recast in a form dependent on the field strength E:

$$\int_{\Omega^e} \frac{\partial \epsilon^{n+1}}{\partial \{e^{(e)}\}_j^{n+1}} \vec{W}_i^{(1)(e)} \cdot \vec{E}^{n+1} d\Omega = \int_{\Omega^e} \frac{\partial \epsilon^{n+1}}{\partial E^{n+1}} \frac{\partial E^{n+1}}{\partial \{e^{(e)}\}_j^{n+1}} \vec{W}_i^{(1)(e)} \cdot \vec{E}^{n+1} d\Omega. \tag{2.75}$$

Here, one last simplification is possible by analyzing the dependence of E on the unknown vector $\{e^{(e)}\}$. In particular, it can be shown through straightforward analysis that:

$$\frac{\partial E^{n+1}}{\partial \{e^{(e)}\}_j^{n+1}} = \frac{\vec{E}^{n+1}}{E^{n+1}} \cdot \vec{W}_j^{(1)(e)}. \tag{2.76}$$

Making this final substitution into equation (2.75) and returning to the full expression

in (2.72) at last produces the desired simplified elemental Jacobian:

$$[J^{(e)}]_{ij}^{n+1} = [M^{(e)}]_{ij}^{n+1} + \frac{\Delta t^2}{4}[C^{(e)}][M_f^{(e)}][C^{(e)}]_{ij}^T + \int_{\Omega^e} \frac{1}{E^{n+1}} \frac{\partial \epsilon^{n+1}}{\partial E^{n+1}} \big(\vec{W}_i^{(1)(e)} \cdot \vec{E}^{n+1}\big)\big(\vec{W}_j^{(1)(e)} \cdot \vec{E}^{n+1}\big) d\Omega \quad (2.77)$$

where in the present case, using the permittivity in (2.59), the derivative reduces to:

$$\frac{\partial \epsilon^{n+1}}{\partial E^{n+1}} = 2\epsilon_0 \chi^{(3)} E^{n+1} \quad (2.78)$$

and the global Jacobian is assembled from the elemental Jacobians as before. Lastly, it is important to note that in the limiting case of a linear problem, in which $\chi^{(3)} = 0$, the Jacobian above reduces exactly to the left-hand side matrix of Subsection 2.2.1, as required.

With this result, the solution may now be found in a straightforward, albeit computationally intensive, manner. In general, the Jacobian in (2.77) will change not only during each time step, but also each iteration of (2.67), as will the $[M]$ matrix in (2.61). They will therefore need to be locally recomputed within each nonlinear element and reassembled into their global counterparts multiple times within each time step.

The general solution procedure for an *instantaneous nonlinearity* is thus as follows:

1. Iterate (2.65) using (2.67) and (2.77), recomputing and assembling $[J]$ and $[M]$ each time, until $\{e\}^{n+1}$ converges to the desired tolerance.

2. Compute $\{b\}^{n+1}$ using (2.66) and the computed $\{e\}^{n+1}$.

3. Repeat the process until the desired end time.

2.2.4 Dispersive Nonlinearity

In this section, for the first time within an implicit mixed FETD formulation, the most general form of the permittivity will be used, including linear dispersion, instantaneous non-

linearity, and dispersive nonlinearity. The resulting scheme is thus highly versatile, capable of modeling a wide variety of material properties.

Recall that for such a general material, its permittivity may be expressed as:

$$\vec{D} = \epsilon_L * \vec{E} + \epsilon_0 \chi^{(3)} \Big(\alpha E^2 + (1-\alpha) g(t) * E^2 \Big) \vec{E}. \tag{2.79}$$

Inserting this expression into Ampère's Law and applying the same spatial discretization procedure used previously produces the following semi-discrete system:

$$[C]^T \{e\} = -\frac{\partial \{b\}}{\partial t} \tag{2.80}$$

$$[C][M_f]\{b\} = [\tilde{M}] \frac{\partial}{\partial t} \Big(\epsilon_L * \{e\} \Big) + \frac{\partial}{\partial t} \Big([\hat{M}]\{e\} \Big) + \{f\} \tag{2.81}$$

where $[\tilde{M}]$ is the same as that seen in equation (2.42), and $[\hat{M}]$ contains the nonlinear contribution defined by:

$$[\hat{M}^{(e)}]_{ij} = \int_{\Omega^e} \epsilon_0 \chi^{(3)} \big(\alpha E^2 + (1-\alpha) g(t) * E^2 \big) \vec{W}_i^{(1)(e)} \cdot \vec{W}_j^{(1)(e)} d\Omega. \tag{2.82}$$

The first term on the right-hand side of equation (2.81) is identical to that seen previously for linear dispersion. Likewise, the second term on the right-hand side is clearly of the same form as that studied for instantaneous nonlinearities in the previous section. The required update equations are thus readily derived by combining both methods. In fact, substituting the linear convolution update equation from (2.57), $\{\mathcal{L}\} = a_0[\tilde{M}] + \{\mathcal{W}_1\}$, as well as observing the time-dependence of the $[\hat{M}]$ matrix, yields the full set of update equations:

$$\begin{aligned} \left(a_0[\tilde{M}] + [\hat{M}]^{n+1} + \frac{\Delta t^2}{4} [C][M_f][C]^T \right) \{e\}^{n+1} = \\ \left(a_0[\tilde{M}] + [\hat{M}]^{n} - \frac{\Delta t^2}{4} [C][M_f][C]^T \right) \{e\}^{n} + \Delta t [C][M_f]\{b\}^n \\ - \{\mathcal{W}_1\}^n + \{\mathcal{W}_1\}^{n-1} - \frac{\Delta t}{2} (\{f\}^n + \{f\}^{n+1}) \end{aligned} \tag{2.83}$$

$$\{b\}^{n+1} = \{b\}^n - \frac{\Delta t}{2}[C]^T(\{e\}^n + \{e\}^{n+1}). \tag{2.84}$$

Equation (2.83) can be more compactly written by defining a new $[K]$ matrix:

$$[K]_{ij} = a_0[\tilde{M}]_{ij} + [\hat{M}]_{ij} \tag{2.85}$$

or equivalently:

$$[K^{(e)}]_{ij} = \int_{\Omega^e} \left(a_0 + \epsilon_0\chi^{(3)}E^2 + (1-\alpha)\epsilon_0\chi^{(3)}g(t) * E^2\right) \vec{W}_i^{(1)(e)} \cdot \vec{W}_j^{(1)(e)} d\Omega \tag{2.86}$$

such that the global update equation for the electric field becomes:

$$\begin{aligned}\left([K]^{n+1} + \frac{\Delta t^2}{4}[C][M_f][C]^T\right)\{e\}^{n+1} = \\ \left([K]^n - \frac{\Delta t^2}{4}[C][M_f][C]^T\right)\{e\}^n + \Delta t[C][M_f]\{b\}^n \\ - \{\mathcal{W}_1\}^n + \{\mathcal{W}_1\}^{n-1} - \frac{\Delta t}{2}\Big(\{f\}^n + \{f\}^{n+1}\Big).\end{aligned} \tag{2.87}$$

Despite equation (2.87) having a conspicuous similarity to that derived for instantaneous nonlinearity, there is one fundamental difference that has yet to be addressed: the convolution *within* the $[K^{(e)}]$ matrices. Thus, the main remaining obstacle is to now derive an update equation for this convolution:

$$\mathcal{B}(t) \triangleq g(t) * E^2(t). \tag{2.88}$$

Luckily, the z-transform theory developed for the linear dispersive case can also be adapted to the present convolution without much complication. Indeed, in some sense the nonlinear convolution above is actually simpler than that for linear dispersion, as the convolution is a scalar rather than vector entity.

Since the nonlinear dispersive kernel $g(t)$ is often likewise the result of a constant-coefficient differential equation model, its Laplace transform can also commonly be expressed

as a rational function in frequency space:

$$g(s) = \frac{r_p s^p + \ldots + r_0}{w_p s^p + \ldots + w_0} \tag{2.89}$$

such that, in the Laplace domain, the convolution becomes:

$$\mathcal{B}(s) = \frac{r_p s^p + \ldots + r_0}{w_p s^p + \ldots + w_0} E^2(s). \tag{2.90}$$

Applying the same bilinear Möbius transform as for linear dispersion, the above equation can similarly be mapped to the z-domain:

$$\mathcal{B}(z) = \frac{h_0 + \ldots + h_p z^{-p}}{1 + \ldots + q_p z^{-p}} E^2(z) \tag{2.91}$$

and thus, upon applying the inverse z-transform, yields an update equation for the nonlinear convolution:

$$\mathcal{B}^n = h_0 (E^2)^n + \ldots + h_p (E^2)^{n-p} - q_1 \mathcal{B}^{n-1} - \ldots - q_p \mathcal{B}^{n-p}. \tag{2.92}$$

Furthermore, just as in the linear case, this update process can be made more efficient by the introduction of auxiliary variables. In view of this, the update equation above can be recast as:

$$\mathcal{G}_\alpha^n = h_\alpha (E^2)^n - q_\alpha \mathcal{B}^n + \mathcal{G}_{\alpha+1}^{n-1} \qquad \alpha < p \tag{2.93}$$

$$\mathcal{G}_\alpha^n = h_\alpha (E^2)^n - q_\alpha \mathcal{B}^n \qquad \alpha = p \tag{2.94}$$

$$\mathcal{B} = h_0 (E^2)^n + \mathcal{G}_1^{n-1} \tag{2.95}$$

where $\mathcal{G}_\alpha$ are the associated new auxiliary variables.

Making the substitution of equation (2.95) for the convolution $\mathcal{B}$ into the definition of

the $[K^{(e)}]$ matrix yields its final required form:

$$[K^{(e)}]_{ij}^{n+1} = \int_{\Omega^e} \Big(a_0 + \epsilon_0 \alpha \chi^{(3)} (E^2)^{n+1} + (1-\alpha)\epsilon_0 \chi^{(3)} \big(h_0 (E^2)^{n+1} + \mathcal{G}_1^n \big) \Big) \vec{W}_i^{(1)(e)} \cdot \vec{W}_j^{(1)(e)} d\Omega. \tag{2.96}$$

With this result, the final requirement to be able to solve the nonlinear update equation for the electric field in (2.87) is to derive the associated Jacobian matrix. Luckily, due to the similarity between the update equation in (2.87) and that for instantaneous nonlinear media, the derivation of the Jacobian proceeds in much the same way, producing:

$$[J^{(e)}]_{ij}^{n+1} = [K^{(e)}]_{ij}^{n+1} + \frac{\Delta t^2}{4} [C^{(e)}][M_f^{(e)}][C^{(e)}]_{ij}^T + \int_{\Omega^e} \frac{1}{E^{n+1}} \frac{\partial \epsilon^{n+1}}{\partial E^{n+1}} \big(\vec{W}_i^{(1)(e)} \cdot \vec{E}^{n+1} \big) \big(\vec{W}_j^{(1)(e)} \cdot \vec{E}^{n+1} \big) d\Omega \tag{2.97}$$

with the exception that the derivative of the permittivity is now given by:

$$\frac{\partial \epsilon^{n+1}}{\partial E^{n+1}} = 2\epsilon_0 \chi^{(3)} E^{n+1} \Big(\alpha + (1-\alpha) h_0 \Big). \tag{2.98}$$

The general solution procedure for a medium exhibiting linear dispersion, instantaneous nonlinearity, and dispersive nonlinearity is now as follows:

1. Iterate equation (2.87), using (2.67) and (2.97), re-computing and assembling the $[J]$ and $[K]$ matrices each time, until $\{e\}^{n+1}$ converges to the desired tolerance.
2. Compute $\{b\}^{n+1}$ using (2.84).
3. Update each of the linear auxiliary variables and the linear convolution, using (2.55) to (2.57), to time $n+1$.
4. Update the nonlinear auxiliary variables and the nonlinear convolution, using (2.93) to (2.95), to time $n+1$.
5. Repeat the process until the desired end time.

2.3 VWE FETD

As mentioned earlier, an alternative to the mixed FETD method is the vector wave equation (VWE) method, which combines Maxwell's Equations into a second-order differential equation rather than treating them as a coupled first-order system. The VWE formulation of the FETD method is generally more popular than mixed methods due to its relative simplicity, however despite this to date there are no reported VWE FETD implementations of dispersive dielectric nonlinearity in the literature. As such, the VWE FETD methods presented in the second half of this section are novel in the generality of materials they can model.

As with the mixed FETD method, this section will begin with a brief overview of linear VWE FETD as a foundation for the nonlinear and dispersive algorithms to follow. These generalizations will then be derived, where the remarkable similarities alluded to earlier between the mixed and VWE implementations will be exploited.

2.3.1 Linear Non-Dispersive Media

In many numerical studies, the main variable of interest is the electric field, for which the VWE can be obtained by eliminating the magnetic field. To do so, Faraday's law can be multiplied by the reciprocal of the permeability and the curl of both sides taken:

$$\nabla \times \frac{1}{\mu} \nabla \times \vec{E} = -\frac{\partial}{\partial t} \left(\nabla \times \frac{\vec{B}}{\mu} \right). \tag{2.99}$$

Substituting the curl on the right-hand side with the electric field expression from Ampère's Law then results in the second-order vector wave equation:

$$\nabla \times \frac{1}{\mu} \nabla \times \vec{E} + \epsilon \frac{\partial^2 \vec{E}}{\partial t^2} = -\frac{\partial \vec{J}}{\partial t}. \tag{2.100}$$

Applying the FETD formulation to this equation proceeds in much the same way as it did for the coupled mixed problem. However, since this equation is only in terms of

the electric field $\vec{E}$, only the Whitney 1-forms or edge elements are required. Applying a Galerkin procedure to the above VWE, the dot product is taken with the 1-forms and the result integrated over each element [20]:

$$\int_{\Omega^e} \left(\nabla \times \frac{1}{\mu} \nabla \times \vec{E} \right) \cdot \vec{W}_j^{(1)(e)} d\Omega + \frac{\partial^2}{\partial t^2} \int_{\Omega^e} \epsilon \, \vec{E} \cdot \vec{W}_j^{(1)(e)} d\Omega = - \int_{\Omega^e} \frac{\partial \vec{J}}{\partial t} \cdot \vec{W}_j^{(1)(e)} d\Omega. \quad (2.101)$$

Using the same vector identity (2.12) as before, in addition to the divergence theorem, the final weak-form of the VWE above is obtained:

$$\int_{\Omega^e} \frac{1}{\mu} \left(\nabla \times \vec{E} \right) \cdot \left(\nabla \times \vec{W}_j^{(1)(e)} \right) d\Omega + \frac{\partial^2}{\partial t^2} \int_{\Omega^e} \epsilon \, \vec{E} \cdot \vec{W}_j^{(1)(e)} d\Omega$$
$$= - \int_{\Omega^e} \frac{\partial \vec{J}}{\partial t} \cdot \vec{W}_j^{(1)(e)} d\Omega - \int_{\partial\Omega^e} \frac{1}{\mu} \left(\nabla \times \vec{E} \right) \times \vec{W}_j^{(1)(e)} \cdot d\vec{S}. \quad (2.102)$$

As with the mixed method, the first term on the right-hand side represents a volume current source with the second term being useful for the imposition of special boundary conditions. Moreover, both of these terms can again be written more compactly by the use of vector notation, such that if the source vector $\{f^{(e)}\}$ is defined as:

$$\{f^{(e)}\}_j = \int_{\Omega^e} \frac{\partial \vec{J}}{\partial t} \cdot \vec{W}_j^{(1)(e)} d\Omega + \int_{\partial\Omega^e} \frac{1}{\mu} \left(\nabla \times \vec{E} \right) \times \vec{W}_j^{(1)(e)} \cdot d\vec{S} \quad (2.103)$$

then equation (2.102) can also be written as:

$$\int_{\Omega^e} \frac{1}{\mu} \left(\nabla \times \vec{E} \right) \cdot \left(\nabla \times \vec{W}_j^{(1)(e)} \right) d\Omega + \frac{\partial^2}{\partial t^2} \int_{\Omega^e} \epsilon \vec{E} \cdot \vec{W}_j^{(1)(e)} d\Omega = -\{f^{(e)}\}_j. \quad (2.104)$$

The electric field can then be replaced in the left-hand side terms with its basis function representation of (2.4):

$$\int_{\Omega^e} \frac{1}{\mu} \left(\nabla \times \sum_{i=1}^{l_e} e_i \vec{W}_i^{(1)(e)} \right) \cdot \left(\nabla \times \vec{W}_j^{(1)(e)} \right) d\Omega + \frac{\partial^2}{\partial t^2} \int_{\Omega^e} \epsilon \sum_{i=1}^{l_e} e_i \vec{W}_i^{(1)(e)} \cdot \vec{W}_j^{(1)(e)} d\Omega = -\{f^{(e)}\}_j. \quad (2.105)$$

This can be more succinctly written by defining the $[T^{(e)}]$ and $[S^{(e)}]$ matrices as follows:

$$[T^{(e)}]_{ij} = \int_{\Omega^e} \epsilon \, \vec{W}_i^{(1)(e)} \cdot \vec{W}_j^{(1)(e)} d\Omega \tag{2.106}$$

$$[S^{(e)}]_{ij} = \int_{\Omega^e} \frac{1}{\mu} \left(\nabla \times \vec{W}_i^{(1)(e)}\right) \cdot \left(\nabla \times \vec{W}_j^{(1)(e)}\right) d\Omega \tag{2.107}$$

from which the global spatially discretized VWE becomes:

$$[T]\frac{d^2\{e\}}{dt^2} + [S]\{e\} + \{f\} = 0 \tag{2.108}$$

where the quantities $[T]$, $[S]$, and $\{f\}$ have been assembled from their local counterparts.

The last step is to now apply a temporal discretization to the above system to obtain a complete update equation for the electric field. In contrast to the mixed method of the previous section in which Crank-Nicolson was used, here a method must be applied which can discretize a second-order derivative in time. One of the most popular choices for this task is the Newmark method [81], in which a function of interest $u(t)$ and its first temporal derivative du/dt are approximated as:

$$\left(\frac{du}{dt}\right)^{n+1} = \left(\frac{du}{dt}\right)^{n} + \Delta t(1-\gamma)\left(\frac{d^2u}{dt^2}\right)^{n} + \Delta t\gamma\left(\frac{d^2u}{dt^2}\right)^{n+1} \tag{2.109}$$

$$u^{n+1} = u^n + \Delta t\left(\frac{du}{dt}\right)^{n} + \frac{\Delta t^2}{2}(1-2\beta)\left(\frac{d^2y}{dt^2}\right)^{n} + \Delta t^2\beta\left(\frac{d^2u}{dt^2}\right)^{n+1} \tag{2.110}$$

where the parameters γ and β can in general be selected to yield different update schemes. For instance, selecting $\gamma = \frac{1}{2}$ yields the most common family of methods known as Newmark-β. As for the parameter β, it can be shown that setting $\beta = 0$ amounts to the central difference method, whereas $\beta = \frac{1}{4}$ amounts to the trapezoidal rule. Of particular importance, however, is the result that for $\beta \geq \frac{1}{4}$ Newmark-β can be shown to be unconditionally stable for linear FETD [20]. As a result, in the following analyses the Newmark-β method will be used with $\beta = \frac{1}{4}$.

By using the semi-discrete equation (2.108) to obtain an expression for the second temporal derivative in (2.110), and using (2.109) to eliminate the first temporal derivative, the final fully discretized update equation for the electric field can be obtained:

$$\left([T]+\frac{\Delta t^2}{4}[S]\right)\{e\}^{n+1}=2\left([T]-\frac{\Delta t^2}{4}[S]\right)\{e\}^{n}-\left([T]+\frac{\Delta t^2}{4}[S]\right)\{e\}^{n-1} - \frac{\Delta t^2}{4}\Big(\{f\}^{n+1}+2\{f\}^{n}+\{f\}^{n-1}\Big). \quad (2.111)$$

Interestingly, comparing equation (2.111) to that obtained for the mixed Crank-Nicolson method in (2.34) reveals a wealth of similarity. Most notably, owing to the relationship between the curl of the 1-forms and the 2-forms in equation (2.7), it can be shown that:

$$[T]=[M] \quad (2.112)$$

$$[S]=[C][M_f][C]^T \quad (2.113)$$

such that the matrices multiplying the $\{e\}$ terms are identical between the two methods. This remarkable homology between the two implementations has been previously noted in [45] and may help explain the posited unconditional stability of the mixed Crank-Nicolson method. By the same token, this striking similarity between the two will prove useful in the derivation of the dispersive and nonlinear algorithms to follow, as much of the theory derived in the previous section for the mixed method will carry forward to the VWE formulation.

2.3.2 Dispersive Media

The inclusion of linear dispersion within the Newmark-β VWE formulation follows precisely the same steps as those outlined for the mixed method. Under the effects of linear dispersion, convolutions are introduced such that the VWE becomes:

$$\nabla\times\frac{1}{\mu}\nabla\times\vec{E}+\frac{\partial^2}{\partial t^2}\left(\epsilon_L*\vec{E}\right)=-\frac{\partial\vec{J}}{\partial t}. \quad (2.114)$$

Applying the same discretization procedure as in the previous subsection, it is found that the convolution which must now be advanced in time is of the form:

$$\{\mathcal{L}\} \triangleq \epsilon_L * [\tilde{T}]\{e\} \tag{2.115}$$

in which the $[\tilde{T}]$ matrix is identical to the $[\tilde{M}]$ matrix in equation (2.42).

Adopting the same z-transform and auxiliary variable updating schemes as in the mixed method, the update equations for the convolution become:

$$\{\mathcal{W}_\alpha\}^n = a_\alpha[\tilde{T}]\{e\}^n - b_\alpha\{\mathcal{L}\}^n + \{\mathcal{W}_{\alpha+1}\}^{n-1} \qquad \alpha < p \tag{2.116}$$

$$\{\mathcal{W}_\alpha\}^n = a_\alpha[\tilde{T}]\{e\}^n - b_\alpha\{\mathcal{L}\}^n \qquad \alpha = p \tag{2.117}$$

$$\{\mathcal{L}\}^n = a_0[\tilde{T}]\{e\}^n + \{\mathcal{W}_1\}^{n-1}. \tag{2.118}$$

The final update equation for the electric field within the VWE treatment of linear dispersive materials is then obtained by the substitution of equation (2.118) into Newmark-β anywhere the convolution (2.115) is encountered:

$$\left(a_0[\tilde{T}] + \frac{\Delta t^2}{4}\right)\{e\}^{n+1} = 2\left(a_0[\tilde{T}] - \frac{\Delta t^2}{4}[S]\right)\{e\}^n - \left(a_0[\tilde{T}] + \frac{\Delta t^2}{4}[S]\right)\{e\}^{n-1}$$
$$- \left(\{\mathcal{W}_1\}^n - 2\{\mathcal{W}_1\}^{n-1} + \{\mathcal{W}_1\}^{n-2}\right) - \frac{\Delta t^2}{4}\left(\{f\}^{n+1} + 2\{f\}^n + \{f\}^{n-1}\right) \tag{2.119}$$

with the general solution procedure being the same as that reported for the mixed method, with the exception of no longer needing to advance the magnetic flux density in time.

2.3.3 Instantaneous Nonlinearity

In the case of an instantaneous nonlinearity ($\alpha = 1$), the spatial discretization procedure is the same as in the linear case, except that it must be noted that the $[T]$ matrix is now a function of time due to the dependence of ϵ on E, requiring it be kept within the temporal

derivative:

$$\frac{\partial^2}{\partial t^2}\Big([T]\{e\}\Big) + [S]\{e\} + \{f\} = 0. \tag{2.120}$$

As a result, the Newmark-β expressions in equations (2.109) and (2.110) must be applied to the matrix-vector product $[T]\{e\}$ as a whole, resulting in the following update equation:

$$\left([T]^{n+1} + \frac{\Delta t^2}{4}[S]\right)\{e\}^{n+1} = 2\left([T]^{n} - \frac{\Delta t^2}{4}[S]\right)\{e\}^{n} - \left([T]^{n-1} + \frac{\Delta t^2}{4}[S]\right)\{e\}^{n-1} - \frac{\Delta t^2}{4}\Big(\{f\}^{n+1} + 2\{f\}^{n} + \{f\}^{n-1}\Big). \tag{2.121}$$

As before, the above equation bears a striking similarity to its linear counterpart, but is now nonlinear and must be solved iteratively. Luckily, as noted earlier, the left-hand side of the above equation is identical to that found in mixed Crank-Nicolson. Given that this is the only term which contributes to the Jacobian, it follows that the expression for the VWE Jacobian is precisely the same as that obtained earlier for the mixed method:

$$[J^{(e)}]_{ij}^{n+1} = [T^{(e)}]_{ij}^{n+1} + \frac{\Delta t^2}{4}[S^{(e)}]_{ij} + \int_{\Omega^e} \frac{1}{E^{n+1}}\frac{\partial \epsilon^{n+1}}{\partial E^{n+1}}\left(\vec{W}_i^{(1)(e)} \cdot \vec{E}^{n+1}\right)\left(\vec{W}_j^{(1)(e)} \cdot \vec{E}^{n+1}\right) d\Omega \tag{2.122}$$

where once again, for an instantaneous nonlinearity:

$$\frac{\partial \epsilon^{n+1}}{\partial E^{n+1}} = 2\epsilon_0 \chi^{(3)} E^{n+1}. \tag{2.123}$$

The solution procedure thus again follows that for the mixed method, but without the need to update the magnetic flux density at each time step.

2.3.4 Dispersive Nonlinearity

Lastly, for the most general case of linear dispersion, instantaneous nonlinearity, and dispersive nonlinearity, the procedure is, unsurprisingly, similar to that previously derived for the mixed method. Indeed, applying the spatial discretization procedure detailed earlier results

in the following semi-discrete system:

$$[\tilde{T}]\frac{\partial^2}{\partial t^2}\Big(\epsilon_L * \{e\}\Big) + \frac{\partial^2}{\partial t^2}\Big([\hat{T}]\{e\}\Big) + [S]\{e\} + \{f\} = 0 \tag{2.124}$$

where the $[\tilde{T}]$ matrix is the same as that seen earlier for linear dispersion, and $[\hat{T}]$ is identical to the $[\hat{M}]$ matrix seen previously with the mixed method in equation (2.82).

Applying the Newmark-β method to the semi-discrete equation in (2.124), borrowing the dispersive update equations from Subsection 2.2.2, and keeping in mind the time dependence of the $[\hat{T}]$ matrix, the following update equation for the electric field may be derived:

$$\begin{aligned}\left([K]^{n+1} + \frac{\Delta t^2}{4}[S]\right)\{e\}^{n+1} &= 2\left([K]^n - \frac{\Delta t^2}{4}[S]\right)\{e\}^n - \left([K]^{n-1} + \frac{\Delta t^2}{4}[S]\right)\{e\}^{n-1} \\ &- \Big(\{\mathcal{W}_1\}^n - 2\{\mathcal{W}_1\}^{n-1} + \{\mathcal{W}_1\}^{n-2}\Big) - \frac{\Delta t^2}{4}\Big(\{f\}^{n+1} + 2\{f\}^n + \{f\}^{n-1}\Big)\end{aligned} \tag{2.125}$$

where the $[K]$ matrix is precisely the same as that seen previously in (2.85):

$$[K] = a_0[\tilde{T}] + [\hat{T}] = a_0[\tilde{M}] + [\hat{M}]. \tag{2.126}$$

Since the $[K]$ matrix above is identical to that seen previously, both the nonlinear dispersive z-transform theory and the Jacobian derived for the mixed FETD method in Subsection 2.2.4 remain largely unchanged:

$$[J^{(e)}]_{ij}^{n+1} = [K^{(e)}]_{ij}^{n+1} + \frac{\Delta t^2}{4}[S^{(e)}]_{ij} + \int_{\Omega^e} \frac{1}{E^{n+1}}\frac{\partial \epsilon^{n+1}}{\partial E^{n+1}}\left(\vec{W}_i^{(1)(e)} \cdot \vec{E}^{n+1}\right)\left(\vec{W}_j^{(1)(e)} \cdot \vec{E}^{n+1}\right) d\Omega \tag{2.127}$$

in which again it is noted that, for a dispersive nonlinearity:

$$\frac{\partial \epsilon^{n+1}}{\partial E^{n+1}} = 2\epsilon_0\chi^{(3)}E^{n+1}\big(\alpha + (1-\alpha)h_0\big). \tag{2.128}$$

Lastly, once more the update procedure for the electric field mirrors that in the mixed

method, with the exception of no longer needing to advance the value of the magnetic flux density in time.

2.4 Convergence Studies

In this section the nonlinear dispersive FETD methods derived in this chapter will be benchmarked to verify their convergence and accuracy. Unfortunately, as was pointed out earlier in Chapter 1, exact solutions to the nonlinear wave equation are in general exceedingly difficult to obtain in closed form. In consequence, without a known test case, it can prove challenging to judge the accuracy and convergence of a new numerical method. For this reason, here a different approach was adopted in which an exact solution was artificially manufactured instead.

Specifically, rather than find a function that satisfies the nonlinear wave equation exactly in some domain, a function is simply selected to be the exact solution instead. Of course, since this function is essentially arbitrary, when the differential equation of interest is applied to this solution there will naturally be a residual. However, if the source term of the differential equation is selected to exactly equal this residual, then the selected function will become the exact solution. A numerical simulation can then be run with this source term and the computed solution compared with the selected, now exact, solution, yielding accuracy and convergence data.

Since *linear* dispersion has already been tested and well established in the literature, in this section the focus will be solely on testing the novel nonlinear aspects of the algorithms in both the instantaneous and dispersive nonlinear regimes. As a result, in each of the scenarios tested in this section, the linear susceptibility $\chi^{(1)}$ is a simple scalar constant. Moreover, unless otherwise stated, all of the algorithms below were implemented and tested in MATLAB R2018a [82].

2.4.1 Instantaneous Nonlinearity

To test the modeling of instantaneous nonlinear media, the following permittivity model was selected:

$$\epsilon = \epsilon_0 \left(1 + \chi^{(1)} + \chi^{(2)} E + \chi^{(3)} E^2\right) \tag{2.129}$$

with $\chi^{(1)} = 2.2$, $\chi^{(2)} = 3.1\ m/V$, and $\chi^{(3)} = 4.3\ m^2/V^2$. Note that here, in contrast to equation (1.8), the $\chi^{(2)}$ term has been kept to provide a more general test case. Moreover, these values for the susceptibilities, as well as all those used in this chapter unless otherwise noted, were chosen either to facilitate the computation of the required source term or to enhance nonlinear effects such that they are more readily observed within the selected computational domains, and do not necessarily reflect any particular existing material. Specifically, the values of $\chi^{(2)}$ and $\chi^{(3)}$ above are far, far, larger than any which would naturally occur, and are meant to test the method.

With this in mind, the simulation domain was selected to be a unit square in two spatial dimensions, $\Omega = [0,1] \times [0,1]$, in which Perfect Electric Conductor (PEC) boundary conditions were applied at each of the four domain edges. Triangular first-order elements were used to mesh the area, and the manufactured exact solutions for the $\vec{E}$ and $\vec{B}$ fields were selected as follows:

$$\begin{aligned} \vec{E} &= A \sin(2\pi x)^2 \sin(2\pi c_1 t)^2 \hat{a}_y && (V/m) \\ \vec{B} &= -\frac{A}{c_1} \sin(2\pi x)\cos(2\pi x)\Big(2\pi c_1 t - \cos(2\pi c_1 t)\sin(2\pi c_1 t)\Big)\hat{a}_z && (Wb/m^2) \end{aligned} \tag{2.130}$$

where $A = 1\ V/m$ and $c_1 = 1/\sqrt{\epsilon_0(1+\chi^{(1)})\mu_0}$. Running this solution through the nonlinear Maxwell Equations results in the following source term being required:

$$\begin{aligned} \vec{J}_v = &-4\pi c_1 \epsilon_0 \cos(2\pi c_1 t)\sin(2\pi c_1 t)^3 \sin(2\pi x)^4 \Big(2\chi^{(2)} + 3\chi^{(3)} \sin(2\pi c_1 t)^2 \sin(2\pi x)^2\Big) \\ &- 2\pi c_1 \epsilon_0 (1+\chi^{(1)})\Big(2\pi c_1 t + \cos(2\pi c_1 t)\sin(2\pi c_1 t) - 4\pi c_1 t \cos(2\pi x)^2\Big)\hat{a}_y \quad (A/m^2). \end{aligned} \tag{2.131}$$

With these parameters, the accuracy and convergence of the method were obtained by performing several computations on progressively refined spatial grids with the temporal time step size held fixed at $\Delta t = h/c$, where c is the speed of light in vacuum and h is the average element edge length in the mesh. The stop criteria for the Newton-Raphson iteration was set to a relative change in solution of 10^{-6}, and the error was measured in terms of the L_2 and L_∞ norms, defined here as:

$$L_2 = \frac{1}{\sqrt{N_t}} \left(\sum_{n=1}^{N_t} \int_\Omega \left| \vec{E}^n - \vec{E}^n_{ex} \right|^2 d\Omega \right)^{1/2} \tag{2.132}$$

$$L_\infty = \max_{t,x,y} \left| \vec{E} - \vec{E}_{ex} \right| \tag{2.133}$$

where $\vec{E}$ is the computed solution, $\vec{E}_{ex}$ is the exact solution, and N_t is the total number of time steps taken in the simulation. Figure 2.1 contains a log-log plot of the error in the L_2 and L_∞ norms for the mixed Crank-Nicolson method for both the electric and magnetic fields (note that the magnetic field has been scaled by c_1 in order to facilitate the comparison). In contrast, Figure 2.2 contains the same error data but for the vector wave equation Newmark-β method, in which only the electric field is computed. Looking at both figures it is evident that for each method all of the errors are converging to first order in the mesh size h (and thus also in the time step size Δt). Despite the Crank-Nicolson and Newmark-β methods being both *second*-order in time, since only first-order spatial basis functions were used the overall convergence rate is expected to fall to first-order, which is precisely the observed trend. Comparing the error in the electric field between the mixed and VWE formulations in both norms, meanwhile, shows that they are roughly equal, meaning both methods demonstrate a similar level of accuracy and comparable solutions.

Lastly, it is worth noting that in Figure 2.1 the error in the magnetic field is approximately one order of magnitude larger than that for the electric field in both norms. This is largely due to the fact that the reported errors are absolute and not relative, meaning the extra error is simply due to the scaled magnetic field being slightly larger than the electric field

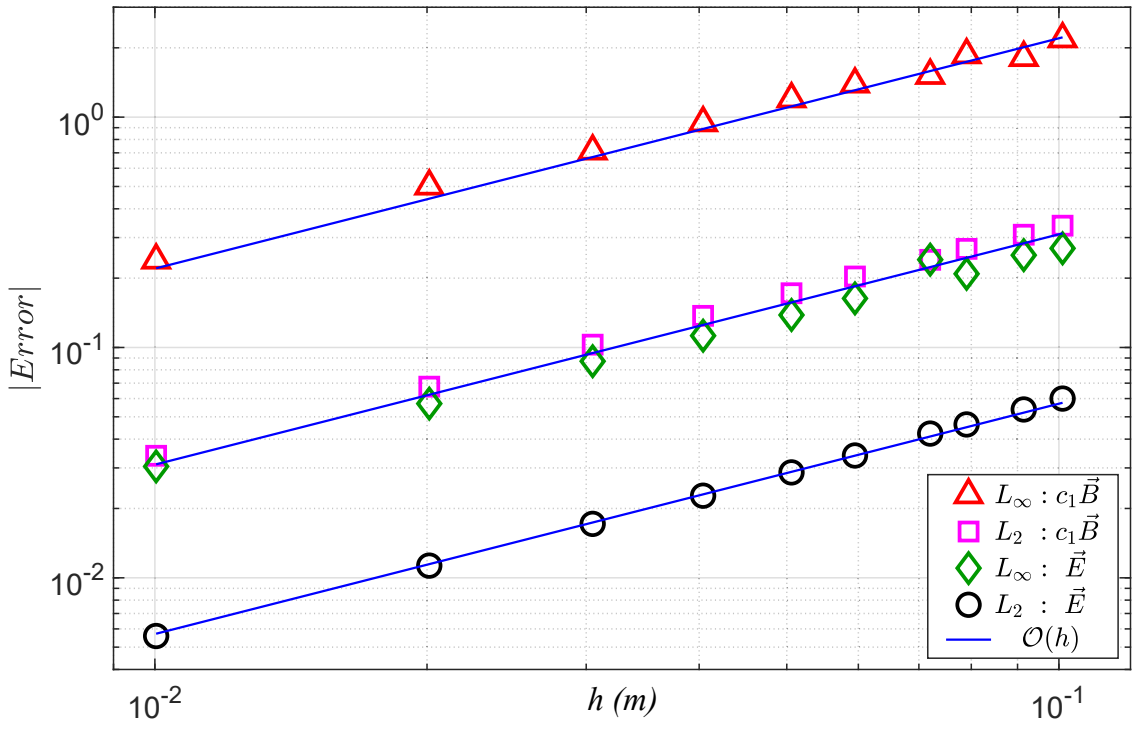

Figure 2.1: Convergence of the mixed Crank-Nicolson FETD method for an instantaneous nonlinearity.

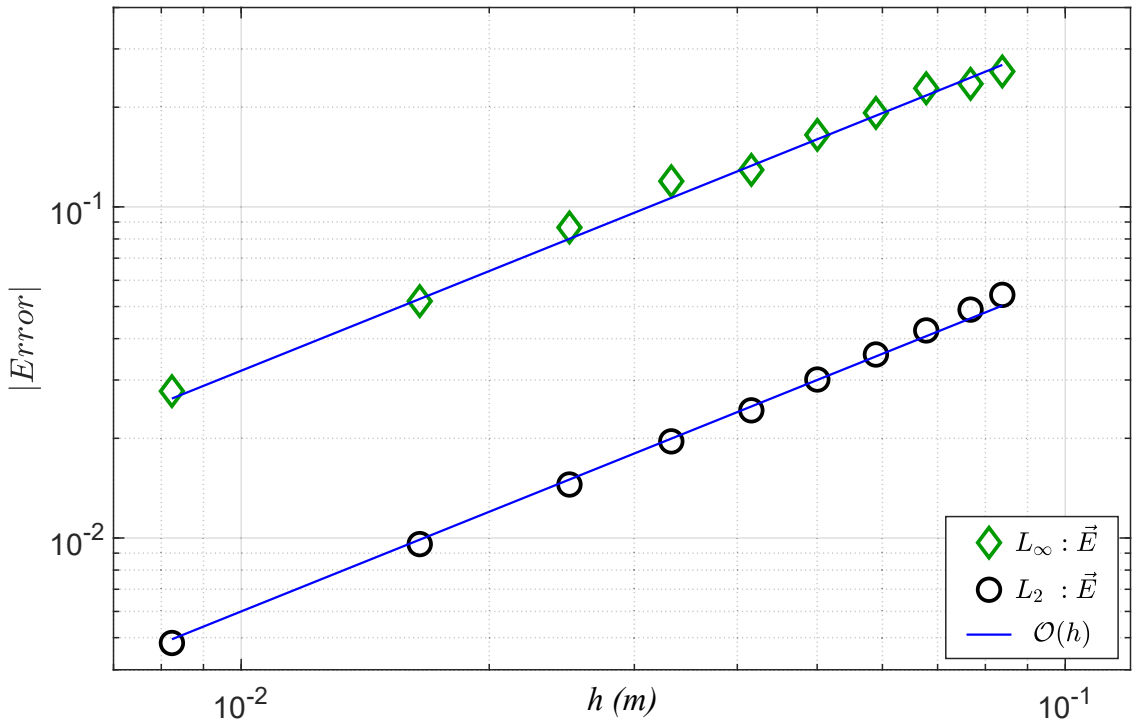

Figure 2.2: Convergence of the VWE Newmark-β FETD method for an instantaneous nonlinearity.

and does not represent an issue with the magnetic field modeling itself.

2.4.2 Dispersive Nonlinearity

With the convergence of the methods having been verified for *instantaneous* nonlinearities, here a similar procedure is now repeated for *dispersive* nonlinearities. To that end, this time the following permittivity model was used:

$$\epsilon = \epsilon_0 \left(1 + \chi^{(1)} + \chi^{(3)} g(t) * E^2\right) \tag{2.134}$$

where $\chi^{(1)} = 2.2$, $\chi^{(3)} = 4.1\ m^2/V^2$ and $g(t)$ was chosen to mimic a first-order Debye-like dispersion:

$$g(t) = \frac{1}{\tau_e} e^{-t/\tau_e} u(t) \tag{2.135}$$

with $\tau_e = 10^{-9}\ s$ and $u(t)$ representing the Heaviside step function.

The same domain, boundary, and mesh parameters were used as in the previous instantaneous nonlinear case, except this time the exact solutions were chosen to be:

$$\vec{E} = \begin{cases} -A \sin(2\pi x) \sum_{n=1}^{3} a_n n \sin(n\omega t) \hat{a}_y & 0 \le t \le T \\ 0 & \text{otherwise} \end{cases} \quad (V/m) \tag{2.136}$$

$$\vec{B} = \begin{cases} -\frac{2\pi}{\omega} A \cos(2\pi x) \sum_{n=0}^{3} a_n \cos(n\omega t) \hat{a}_z & 0 \le t \le T \\ 0 & \text{otherwise} \end{cases} \quad (Wb/m^2) \tag{2.137}$$

where:

$$a_0 = \frac{3179}{9000} \qquad a_1 = -\frac{4392}{9000} \qquad a_2 = \frac{1305}{9000} \qquad a_3 = -\frac{92}{9000}, \tag{2.138}$$

$A = 1\ V/m$, $T = 5.97 \times 10^{-9}\ s$, and $\omega = 1.053 \times 10^9\ rad/s$, such that the temporal shape of the $\vec{E}$ and $\vec{B}$ fields were the Differentiated Blackman-Harris and Blackman-Harris pulses, respectively.

As before, these solutions were run through Maxwell's Equations to obtain the required source term for the convergence study. However, while this source term was obtained in exact closed form, due to the convolution it is a large and intractable expression and has thus been omitted here for clarity.

Figure 2.3 shows the resulting error for the mixed Crank-Nicolson method over a series of refined meshes as again measured in the L_2 and L_∞ norms of equations (2.132) and (2.133), respectively. The magnetic field was once again scaled by a factor of c_1 to facilitate the analysis and comparison. Figure 2.4, meanwhile, shows the same data but this time as measured for the vector wave equation Newmark-β implementation. Looking at both figures it is once again evident that all fields across both methods are converging to the expected first-order accuracy in both norms. Furthermore, a comparison between the electric field errors of the two methods shows that, once again, the mixed and VWE methods yield similar levels of accuracy. The results of these convergence studies thus not only demonstrate the appropriate degree of convergence, but also indicate that the mixed Crank-Nicolson and VWE Newmark-β methods are very similar in terms of their overall accuracy and results. This is both expected and encouraging, considering the extensive homology of the methods discussed in Subsection 2.3.1. Overall, these convergence studies thus strongly corroborate that the methods introduced in this thesis are not only correctly modeling complex material behaviour, but doing so in an accurate and predictable way.

2.5 Numerical Examples

In the previous section the accuracy and convergence of the nonlinear dispersive FETD methods was evaluated through the use of fabricated non-physical solutions. In contrast, this section will more prominently demonstrate the utility and power of these methods via the simulation of several well-known and physically significant nonlinear phenomena. In particular, the occurrence of three uniquely nonlinear propagation modes known as spatial

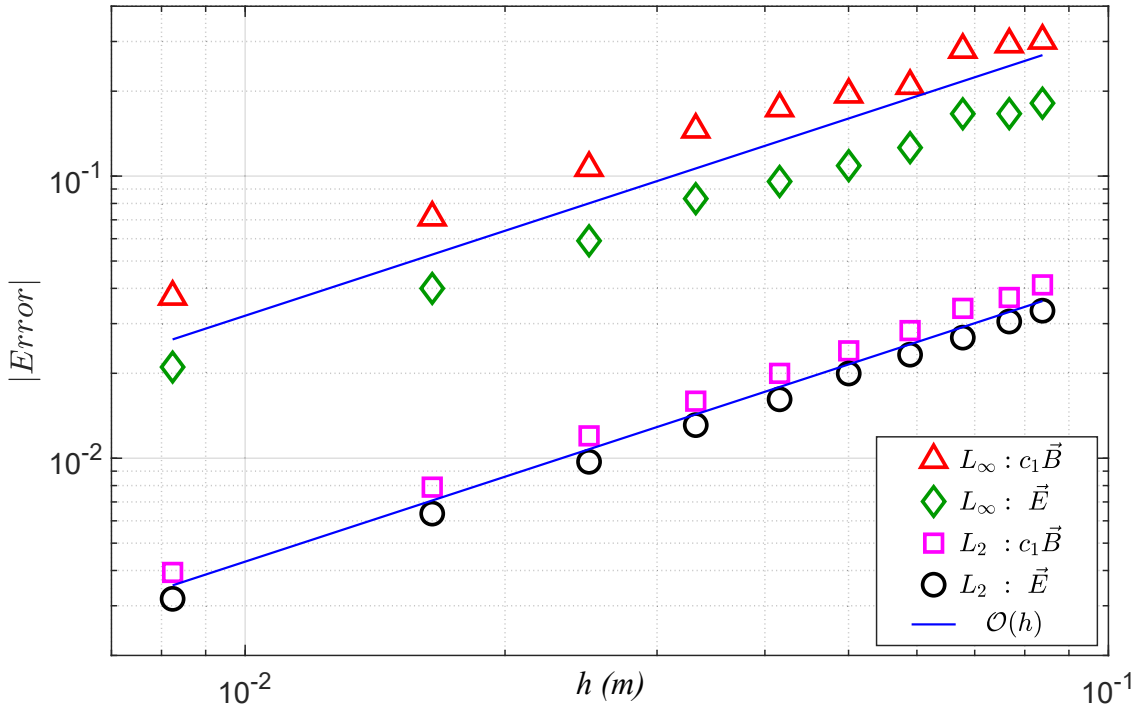

Figure 2.3: Convergence of the mixed Crank-Nicolson FETD method for a dispersive nonlinearity.

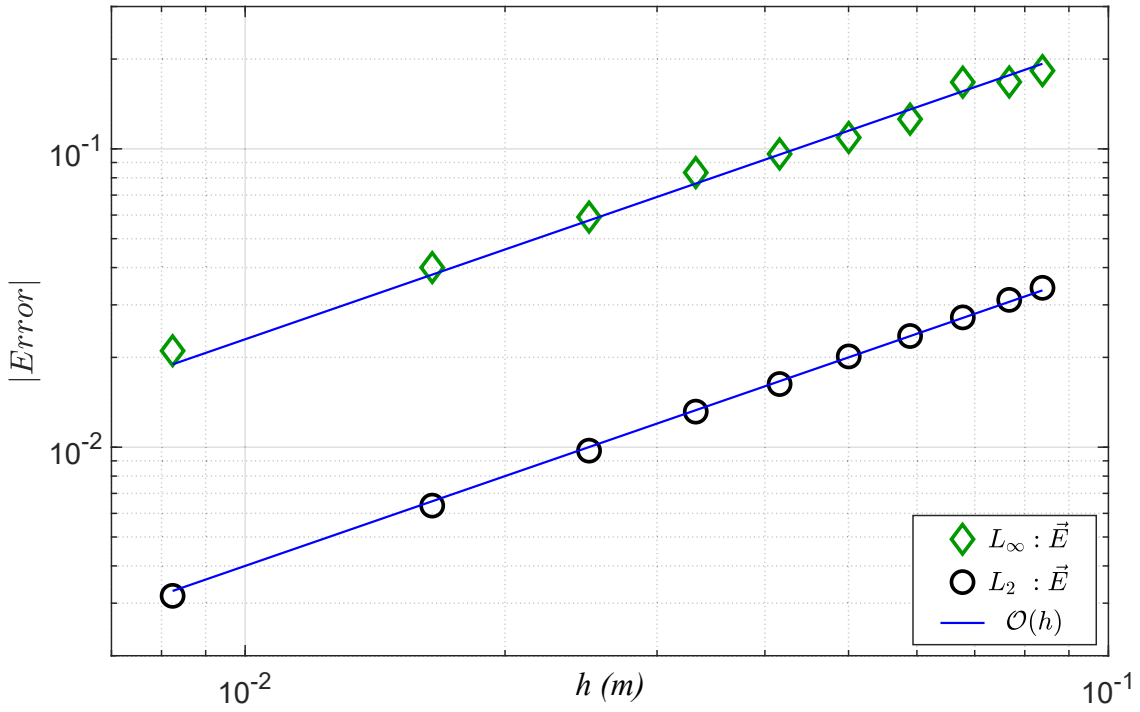

Figure 2.4: Convergence of the VWE Newmark-β FETD method for a dispersive nonlinearity.

solitons, temporal solitons, and supercontinuum generation will be investigated. While the accuracy of the methods cannot be verified for these problems due to the lack of exact closed-form solutions, the successful recreation and modeling of these phenomena within the derived nonlinear FETD methods further corroborates their versatility, applicability, and proper functioning.

2.5.1 Spatial Soliton

One fundamental aspect of non-guided wave propagation in bulk linear media is *diffraction*, in which an initially confined beam gradually spreads out in space. In consequence, any initially focused beam will tend to rapidly widen and spread out as it propagates, becoming less focused and more diffuse [2]. In a nonlinear medium, however, the dependence of the permittivity or refractive index on the field strength can give rise to a lensing or self-focusing effect. Essentially, the transverse profile of a beam's intensity can yield a gradient in the material's refractive index such that it mimics a convex lens. The result is that if a beam of the correct profile and intensity is emitted into a bulk nonlinear medium, the focusing effect of the nonlinearity can be made to exactly counterbalance the diffraction, yielding the stable propagation of a confined beam over large distances without a guiding structure, known as a spatial soliton [23].

To demonstrate the occurrence of this phenomena, a rectangular slab of bulk nonlinear media measuring 30 cm wide by 100 cm long was simulated in two dimensions with the mixed FETD formulation. Here, a two dimensional domain is important, since cylindrically symmetric three-dimensional spatial solitons are known to be unstable in a pure Kerr medium [23]. Additionally, these results are equally obtainable with the VWE formulation, but this data has been omitted for brevity. An initially confined beam was injected into the left-hand side of the slab and was permitted to propagate, with the other three boundaries having Perfect Electric Conductor (PEC) conditions applied. More specifically, the beam's electric

field was given by:

$$\vec{E} = A\,\mathrm{sech}\left(k_y(y-y_0)\right)f(t)\hat{a}_y \tag{2.139}$$

where $A = 3\times10^{10}\ V/m$, $k_y = 180\ m^{-1}$, $y_0 = 0.15\ m$ and the temporal profile $f(t)$ was that of a ramped sinusoid:

$$f(t) = \begin{cases} 0 & t \le 0 \\ \frac{1}{2}\left(1-\cos(\frac{\omega t}{2\alpha})\right)\sin(\omega t) & 0 < t \le \frac{2\pi K}{\omega} \\ \sin(\omega t) & t > \frac{2\pi K}{\omega} \end{cases} \tag{2.140}$$

with $f = 2.4\times10^{9}\ Hz$, $\omega = 2\pi f$, and $K = 3/2$, resulting in a transverse profile with a Full Width at Half Maximum (FWHM) of 1.46 cm. The average edge length in the mesh was selected to be $h = 2\times10^{-3}\ m$, the temporal time step set to $\Delta t = h/c$, and the Newton-Raphson stop criteria made equal to a relative change in solution of 10^{-6}.

Figure 2.5 depicts the result of the simulation for a purely linear medium in which $\epsilon = 4.2\epsilon_0$ (or equivalently for which $\chi^{(1)} = 3.2$), and $\chi^{(2)} = \chi^{(3)} = 0$. The initial bright localized beam can easily be seen on the leftmost boundary, however with the lack of nonlinearity the beam rapidly diffracts and spreads out, creating a diffuse interference pattern as the wave rebounds off the PEC walls.

In contrast, Figure 2.6 shows the results of the exact same simulation except where the linear medium has been replaced with one exhibiting an instantaneous nonlinearity of the form presented in equation (1.8), with $\alpha = 1$, $\chi^{(1)} = 3.2$, and $\chi^{(3)} = 1.5\times10^{-19}\ m^2/V^2$. This time the focusing effects of the nonlinearity are abundantly clear, as the beam remains roughly confined to its original focused shape during propagation. Note that in this particular simulation, since the beam intensity is ramped up from zero, an initial non-focused wavefront occurs until such a time as the beam has reached sufficient strength for the nonlinearity to effectively mitigate diffraction.

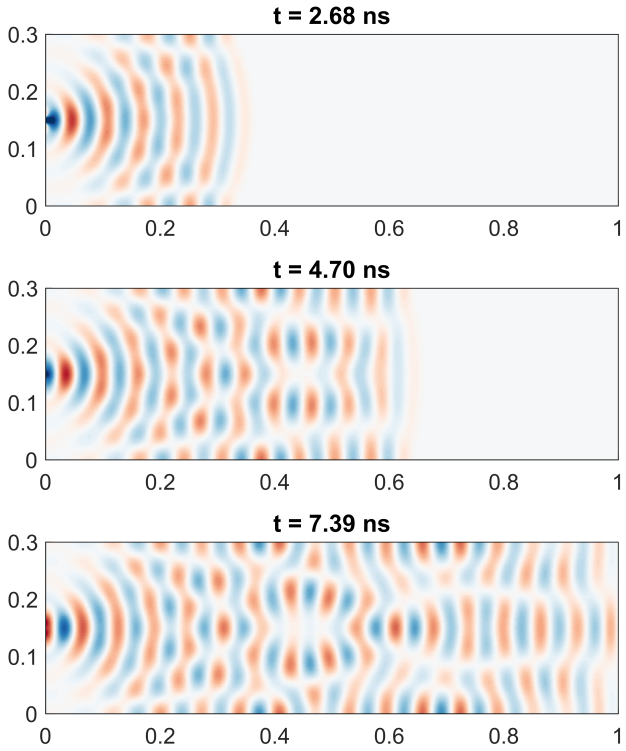

Figure 2.5: Diffuse interference pattern resulting from diffraction in a bulk linear medium.

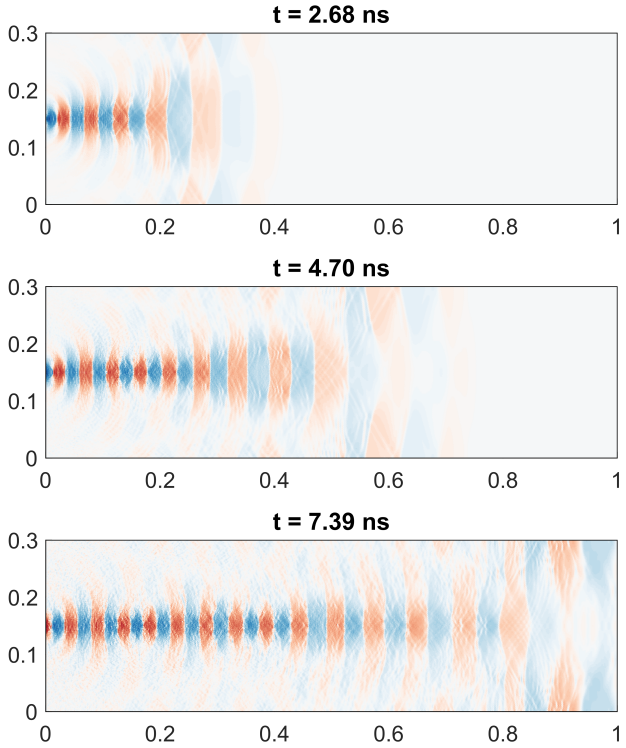

Figure 2.6: Creation of a focused spatial soliton due to the presence of an instantaneous nonlinearity.

2.5.2 Temporal Soliton

Whereas the previous example dealt with wave propagation in bulk media, here attention is turned toward the more common scenario of guided waves, such as those found in fibre optic cables. Due to the confining and guiding properties of the waveguide, unlike in bulk media, a propagating wave will not diffract, maintaining in general a fixed transverse profile as it travels. However, depending on the operating frequency and the materials from which the guide is made, the signal can still become distorted. Indeed, the most common source of such distortion is linear material or *chromatic* dispersion. As mentioned in previous sections, materials exhibiting linear dispersion in essence have frequency-dependent permittivities, meaning that each spectral component of a pulse or signal will propagate at a slightly different speed, a phenomena often characterized in terms of the Group Velocity Dispersion (GVD). Due to these differences in propagation speed, an initially well-formed pulse will gradually broaden, becoming increasingly distorted as its constituent spectral components separate during propagation. Generally speaking such pulse broadening is an unwanted and problematic phenomena, as any signal sent down the guide may not be intelligible upon reception at the other end. Dispersion is thus one main complication limiting bandwidth and transmission rates in optical fibres [83].

One possible way to mitigate the effects of GVD, however, is by leveraging nonlinearity within the optical fibre. Since within a nonlinear medium the permittivity is a function of field strength, different parts of the signal will experience varying propagation velocities according to the wave intensity at that point. In consequence, the resulting self-phase modulation leads to a change in frequency or "chirp" over the length of the pulse, with the leading edge decreasing in frequency and the trailing edge increasing in frequency [23].

In this manner, if a medium exhibits *anomalous* linear dispersion (that is, dispersion in which the high frequency components travel faster than the low frequency ones) as well as precisely the right amount of nonlinearity (tuned via the pulse shape and intensity) the two phenomena can be made to effectively cancel each other out. The resulting pulse thus

propagates through the material without significant distortion or broadening and is known as a *temporal soliton*.

As before, a problem domain was selected to recreate this phenomena, this time within the nonlinear dispersive VWE FETD framework. Rather than a bulk medium, however, a dielectric slab waveguide was chosen to more closely mimic the operation of an optical fibre. The resulting domain was rectangular in shape, measuring 10 μm wide by 100 μm long, and was composed of three dielectric layers. The two exterior cladding regions were selected to be free space, while the middle dielectric measured 2 μm in width and was made to exhibit the full range of phenomena discussed so far, including linear dispersion, instantaneous nonlinearity, and dispersive nonlinearity. More specifically, the linear dispersion was selected to be second-order Lorentz, with a time-dependent linear susceptibility given by:

$$\chi^{(1)}(t) = \chi_\infty \delta(t) + \frac{(\chi_s - \chi_\infty)\omega_0^2}{\sqrt{\omega_0^2 - \delta^2}} e^{-\delta t} \sin\left(\sqrt{\omega_0^2 - \delta^2}\, t\right) u(t) \tag{2.141}$$

where ω_0 is the resonant frequency, δ is the damping constant, $u(t)$ is the Heaviside step function, and $\delta(t)$ is the Dirac delta function. In the present case, the following values were selected for the linear dispersive parameters:

$$\omega_0 \approx 1.885 \times 10^{14} \quad (rad/s) \quad \delta = 2 \times 10^{11} \quad (rad/s) \quad \chi_s = 6.1 \quad \chi_\infty = 4.7 \tag{2.142}$$

and were chosen to yield easily observable pulse broadening over the distance simulated.

Additionally, the parameter α of equation (1.8) was selected to be 0.8, meaning both instantaneous and dispersive nonlinearities were present. In the dispersive case, the nonlinear response function $g(t)$ was also selected to be of the Lorentzian-type:

$$g(t) = \left(\frac{\tau_1^2 + \tau_2^2}{\tau_1 \tau_2^2}\right) e^{-t/\tau_2} \sin\left(\frac{t}{\tau_1}\right) u(t) \tag{2.143}$$

where τ_1 represents the optical phonon period and τ_2 the phonon lifetime [27]. Here, these

parameters were selected as $\tau_1 = 3.358 \times 10^{-14}\ s$ and $\tau_2 = 1.1 \times 10^{-13}\ s$, with the nonlinear susceptibility set to $\chi^{(3)} = 1.1 \times 10^{-18}\ m^2/V^2$.

Lastly, the incident pulse was again excited on the leftmost boundary of the waveguide, in the fundamental transverse magnetic (TM) mode, whose $\vec{B}$ field is given as:

$$\vec{B} = \begin{cases} H_2 e^{-a(y-y_0)} h(t)\ \hat{a}_y & 6\ \mu m < y < 10\ \mu m \\ H_1 \cos\big(k_x(y-y_0)\big) h(t)\ \hat{a}_y & 4\ \mu m < y < 6\ \mu m \\ H_2 e^{a(y-y_0)} h(t)\ \hat{a}_y & 0\ \mu m < y < 4\ \mu m \end{cases} \tag{2.144}$$

with $a = 1.443 \times 10^6\ m^{-1}$, $k_x = 1.368 \times 10^6\ rad/m$, $y_0 = 5 \times 10^{-6}\ m$, $H_1 = 10\ Wb/m^2$, and $H_2 = 11.71\ Wb/m^2$. Meanwhile, the temporal profile $h(t)$ was selected as a modulated hyperbolic secant given by:

$$h(t) = \begin{cases} \operatorname{sech}(\frac{t-t_0}{T}) \sin\big(\omega(t-t_0)\big) & 0 < t < t_p \\ 0 & \text{otherwise} \end{cases} \tag{2.145}$$

with $T = 2 \times 10^{-14}\ s$, $f = 5 \times 10^{13}\ Hz$, $\omega = 2\pi f$, $t_0 = 3/f$, and $t_p = 2t_0$. The resulting pulse had a FWHM of approximately 52.7 fs and a fundamental frequency of 50 THz, with roughly 6 periods of the carrier wave being contained within the pulse envelope.

To illustrate the negative effects of linear dispersion and GVD on their own, an initial simulation with this setup was performed in which all of the nonlinearity was turned off. The results of this simulation (performed with $h = 0.2\ \mu m$, $\Delta t = h/c$, and triangular first-order elements) is shown in Figure 2.7. This level of spatial discretization amounts to roughly 15 points per wavelength within the medium and should therefore result in negligible numerical vs chromatic dispersion. While the pulse is initially well localized and compact within the guide, as it propagates the effects of anomalous linear dispersion rapidly take hold, broadening it significantly over time. Indeed, the anomalous nature of the dispersion is partially visible in the pulse as higher frequency components have collected toward the

front, leaving the lower frequency components to the rear, resulting in a negative chirp. On the other hand, Figure 2.8 reveals the effects of that same linear dispersion, but with the added influence of the above nonlinearities in effect. This time, the linear dispersion has been largely counterbalanced by the self-phase modulation of the nonlinearity, yielding a temporal soliton. As the pulse propagates, therefore, its initial shape and size remain roughly intact, with no significant alteration in frequency or chirp detected.

2.5.3 Supercontinuum Generation

In this subsection, a last example of a real-world application of the simulation tools developed in this chapter will be demonstrated. More specifically, the phenomenon of *supercontinuum generation* will be exhibited, in which laser light undergoes extreme spectral broadening under the influences of dispersion and nonlinearity, resulting in a very wide continuous optical spectrum. This phenomena is significant not only for its spectacular visually observable effects, but also because it represents an important source of high power density ultra-broadband radiation [84].

The underlying physical processes behind supercontinuum generation in nonlinear optical fibres can be quite varied and depends on different parameters such as chromatic dispersion, laser pulse intensity, width, and duration, as well as the wavelength or frequency at which the system is operated. However, one common regime of operation in which supercontinuum generation is known to occur is when working with ultrafast femtosecond pulses excited in the vicinity of a waveguide's zero dispersion wavelength (ZDW), the point where the overall fibre transitions from being anomalously dispersive to normally dispersive (or vice versa).

As detailed extensively in [85], when femtosecond pulses are created or "pumped" at a wavelength or frequency in the anomalous dispersion regime, but close to the ZDW, several mechanisms come into effect. Initially, the high intensity of the beam causes the creation of a high-order temporal soliton, similar in nature to that described in Subsection 2.5.2 but with a more complex temporal evolution. After a short distance, however, perturbations

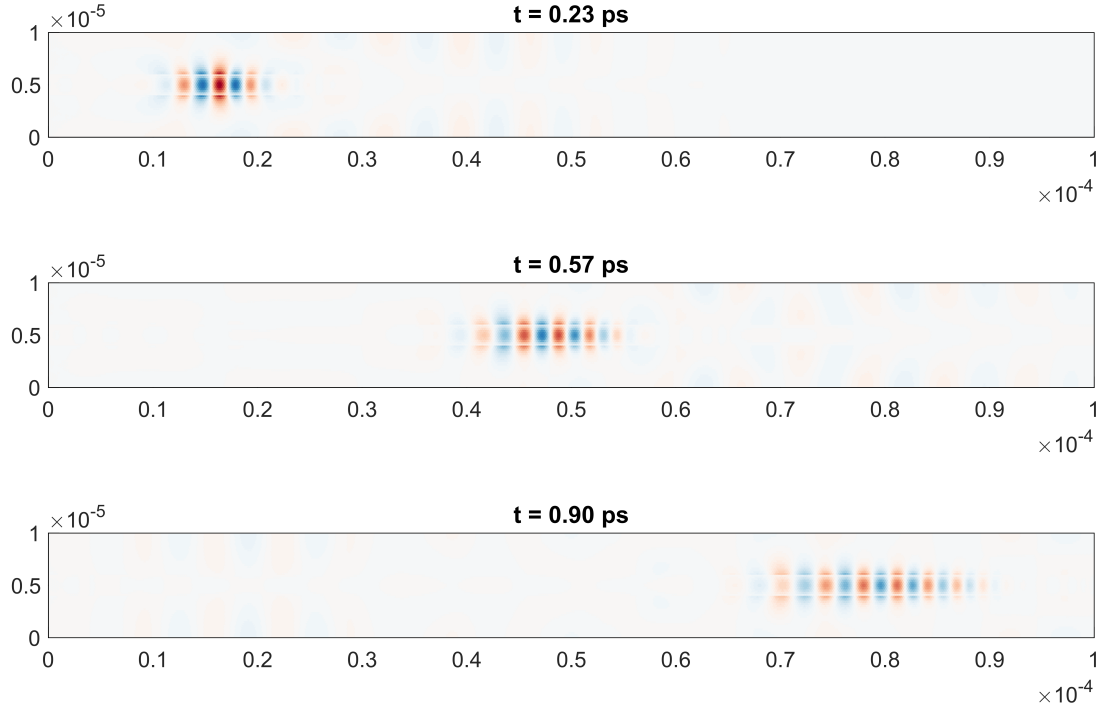

Figure 2.7: Demonstration of the effects of anomalous linear dispersion in a dielectric slab waveguide.

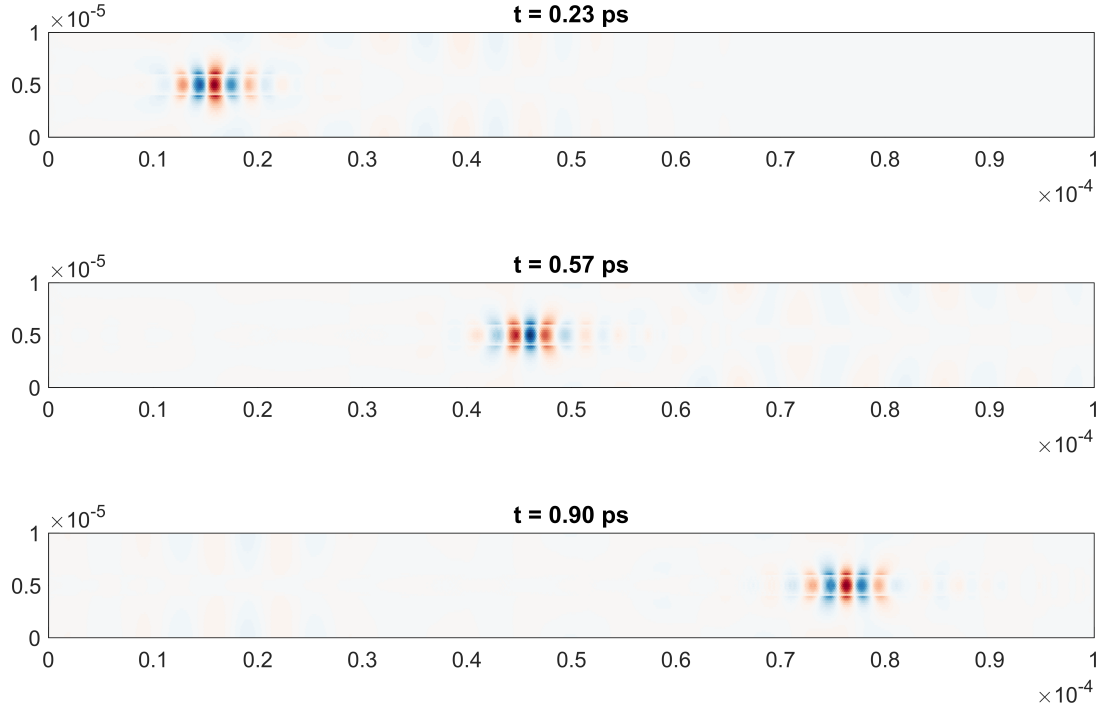

Figure 2.8: Creation of a temporal soliton via the introduction of dielectric nonlinearity.

cause this high-order soliton to be unstable, resulting in a process called soliton fission. By fissioning, the high-order soliton ejects energy into many smaller distinct low-order or fundamental soliton components. However, due to the proximity of the ZDW, some of this energy is shed into the normally dispersive region, creating dispersive waves. The energy which remains in the anomalous dispersion region, meanwhile, can itself continue to shift to longer wavelengths due to nonlinear soliton phenomena such as self-frequency shift and cross-phase modulation, among others. The result is that, due to the interactions of nonlinearity, anomalous dispersion, and normal dispersion, an initially spectrally confined pulse experiences dramatic spectral broadening, with most of the initial energy now spread out over a wide range of frequencies.

To demonstrate the occurrence of this phenomenon, a new numerical simulation was devised to which the mixed FETD method could be applied. This time, a parallel plate waveguide was created and filled with a material exhibiting both linear dispersion and an instantaneous nonlinearity ($\alpha = 1$). The waveguide measured 1 μm in width, 1 mm in length, and was equipped with PEC boundaries.

The pulse itself was excited on the leftmost boundary and had a constant transverse component with a modulated hyperbolic secant envelope in time. The corresponding magnetic field was given by:

$$\vec{B} = \begin{cases} A\,\mathrm{sech}(\frac{t-t_0}{T})\sin\big(\omega(t-t_0)\big)\hat{a}_y & 0 < t < 2t_0 \\ 0 & \text{otherwise} \end{cases} \tag{2.146}$$

in which $A = 10\ Wb/m^2$, $f = 2\times10^{14}\ Hz$, $t_0 = 30/f$, $\omega = 2\pi f$, and $T = 2.84\times10^{-14}\ s$. The resulting pulse had a FWHM of approximately 74.8 fs. The average element edge length in the mesh was set to $h = 0.1\ \mu m$, with first-order triangular elements, and a temporal time step size of $\Delta t = h/c$.

As for the material parameters, the linear dispersion was modeled by the following three-

term Sellmeier equation:

$$\chi^{(1)}(\lambda) = \sum_{i=1}^{3} \frac{B_i \lambda^2}{\lambda^2 - C_i} \tag{2.147}$$

where λ is the equivalent free-space wavelength measured in μm and the values of the coefficients B_i and C_i are given in Table 2.2. The nonlinear susceptibility, meanwhile, was

B_1	B_2	B_3	C_1	C_2	C_3
0.6962	0.4079	0.8975	0.0684	0.1162	6.3000

Table 2.2: Sellmeier coefficients for supercontinuum generation.

assigned the value of $\chi^{(3)} = 8 \times 10^{-21}\ m^2/V^2$. Note that the values given in Table 2.2 are those associated with fused silica [86], with the exception of the value of C_3. The deviations of C_3 and $\chi^{(3)}$ from the true values for silica were made to mildly alter the effects of dispersion and nonlinearity, such that supercontinuum generation could be observed over a shorter distance and thus require less computational resources. Given sufficient computational time and resources, however, the full true parameters could easily be used.

Figure 2.9 depicts a visual representation of the evolution of the pulse's spectral contents during propagation, as measured in the center of the guide at $y = 0.5\ \mu m$. The horizontal axis represents the equivalent free-space wavelength of each frequency, the vertical axis the propagation distance within the guide, and the color the spectral intensity at each point. While most of the spectral energy is initially clustered around the pulse's fundamental frequency of 200 THz (1.5 μm), the pulse rapidly fissions and decomposes, spreading its energy out into a broad range of adjacent wavelengths/frequencies. This is especially evident in Figure 2.10 where the spectral contents of the pulse have been individually plotted for three separate locations in the guide. Again, it can be clearly seen that while the initial pulse is nicely localized around the 200 THz (1.5 μm) fundamental frequency (wavelength), after propagating through the guide the energy has uniformly spread out over a much wider range. The resulting signal has roughly equal spectral power over a 1.4 μm range, whereas the original pulse's energy was contained within a 3 dB bandwidth of only roughly 50 nm.

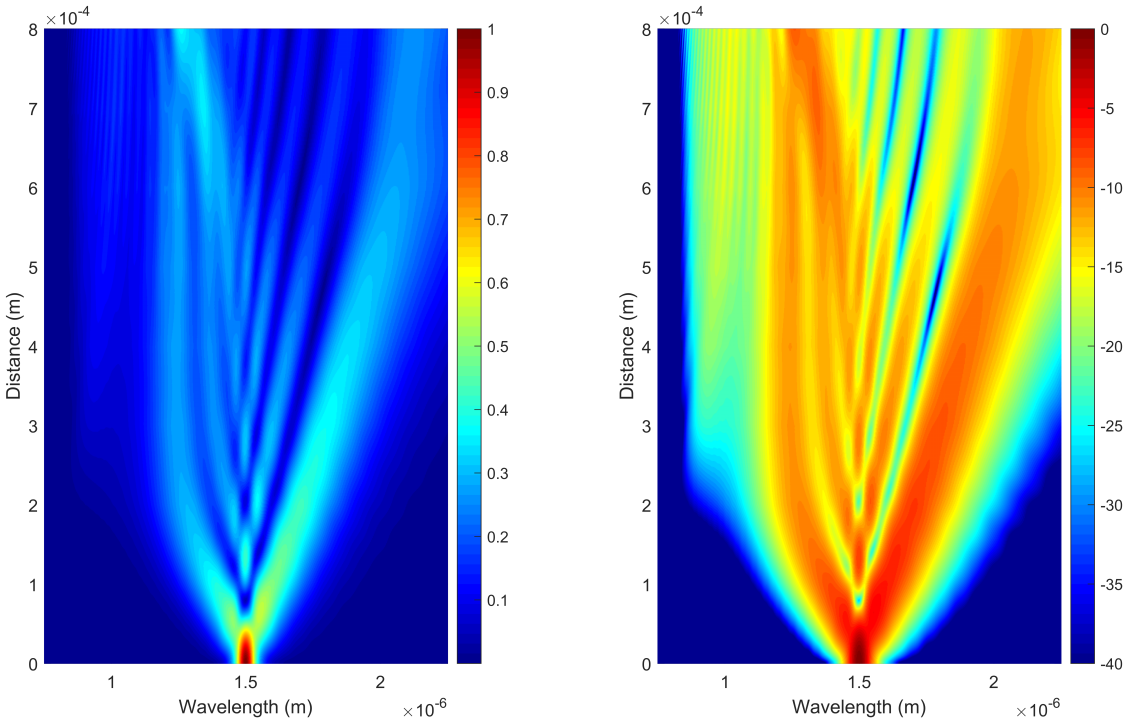

Figure 2.9: Linear (left) and logarithmic (right) visual depiction of the spectral contents of the pulse as it propagates down the guide. The intensity has been normalized to the highest value.

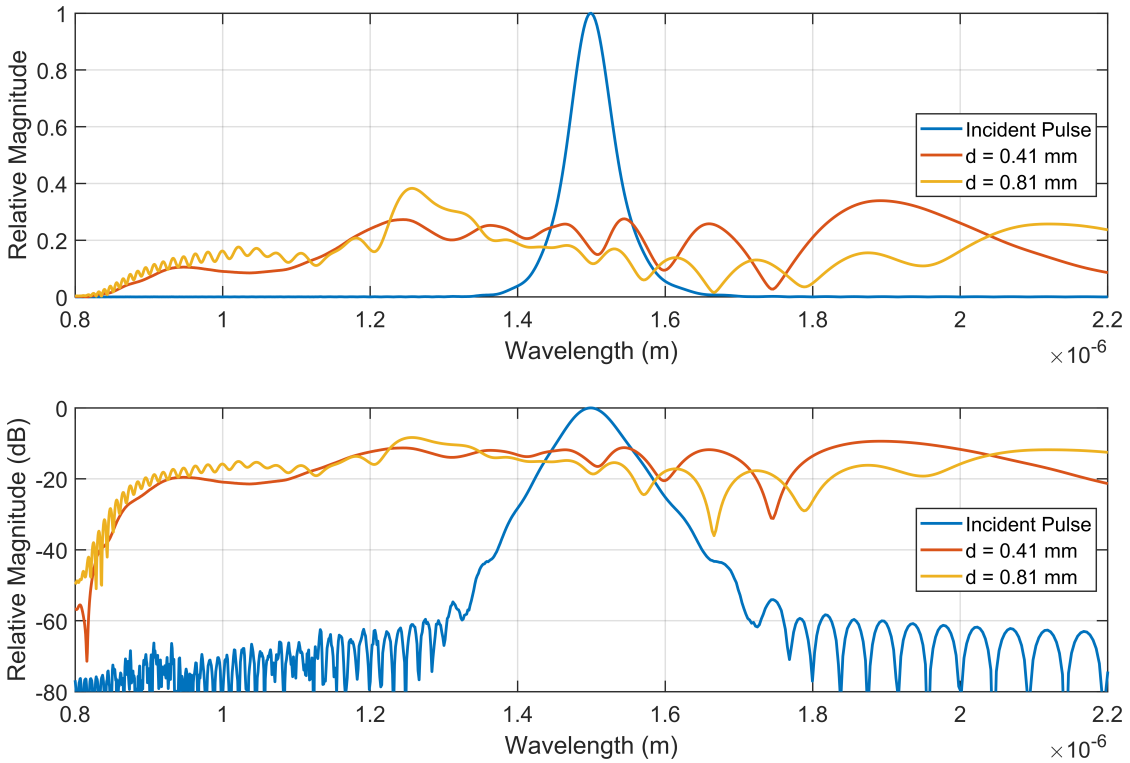

Figure 2.10: Linear (top) and logarithmic (bottom) plots of the pulse's normalized spectral composition after traveling 0.41 *mm* and 0.81 *mm* in the guide.

2.6 Summary

In this chapter a novel family of FETD-based methods was derived for the treatment of electrically complex media within the full non-approximative Maxwell's Equations. The resulting methods are capable of modeling very general combinations of linear dispersion, instantaneous nonlinearity, and dispersive nonlinearities up to, in principle, arbitrary dispersive and nonlinear orders.

Two versions of the FETD algorithm were derived and implemented. The first was based upon the mixed formulation making use of both edge and face elements as well as the Crank-Nicolson temporal discretization. The second, meanwhile, was based upon the Vector Wave Equation (VWE) formulation which used only edge elements and the Newmark-β temporal discretization. Both methods introduced dispersive modeling via the z-transform method, implemented via a series of auxiliary variable update equations. The nonlinearity, in turn, resulted in matrix quantities becoming functions of time, requiring the use of nonlinear root finding. Specifically, it was shown that the nonlinear system's Jacobian can be obtained in closed-form, easily allowing for the use of the Newton-Raphson method.

The convergence of the newly derived methods was then tested via several artificially manufactured solutions and was shown to be correctly approaching the exact solution as the mesh was refined. It was also demonstrated that the mixed and VWE formulations produce similar results with comparable levels of accuracy. Moreover, the utility and proper functioning of these methods was also showcased via the simulation of several physically significant nonlinear phenomena, including spatial and temporal solitons, and supercontinuum generation.

The nonlinear dispersive algorithms derived in this chapter will form the base for all further chapters in this thesis. In particular, the next chapter will provide a more in-depth look at the characteristics of these algorithms, whereas subsequent chapters will develop additional tools and techniques to improve their applicability and usefulness.

Chapter 3

Implementation, Stability, and Energy

Whereas the last chapter presented and derived the mathematical basis of implicit FETD methods for nonlinear dispersive media, in this chapter the focus will be on the implementation and characteristics of these algorithms. In particular, techniques for obtaining valid and efficient solutions from Newton-Raphson will be discussed, as well as details concerning the evaluation of elemental matrices and auxiliary variable updating. Lastly, the critical notion of stability will be discussed, including an analysis of numerically conserved quantities of interest, such as energy.

3.1 Newton-Raphson Iteration

A key part of the nonlinear algorithms presented in the last chapter was the iterative solution of a nonlinear system of equations via the Newton-Raphson method:

$$\{F\}\big(\{x\}\big) = 0 \tag{3.1}$$

$$\{x\}_{(k+1)} = \{x\}_{(k)} - [J]^{-1}_{(k)}\{F\}_{(k)}. \tag{3.2}$$

As was briefly mentioned, one of the major benefits of the Newton-Raphson method is its fast convergence. More specifically, it can be shown that the Newton-Raphson method exhibits

quadratic convergence, which is to say:

$$\left|\{x\}_{(k+1)} - \{\hat{x}\}\right| \leq C\left|\{x\}_{(k)} - \{\hat{x}\}\right|^2 \tag{3.3}$$

where $\{x\}_{(k)}$ is the solution after k iterations of the algorithm, $\{\hat{x}\}$ is the exact solution, C is some positive constant, and $|\cdot|$ denotes the Euclidean norm [87].

Of course, the quadratic convergence of equation (3.3) is only valid if the solution actually converges. Unfortunately, this is not always guaranteed, and generally requires a starting value $\{x\}_{(0)}$ sufficiently close to the true solution $\{\hat{x}\}$ to ensure convergence. Moreover, should the nonlinear system being solved have multiple possible solutions, the one to which the method converges can exhibit extreme sensitivity to the starting guess, leading to so-called basins of attraction [88]. As a result of these considerations it is imperative that an appropriate starting guess be selected not only to ensure convergence to the correct solution, but also convergence as quickly and efficiently as possible.

Luckily, since the FETD algorithms derived previously all involve time-stepping, a natural choice for the initial guess is available. To illustrate, consider a time-dependent function $u(t)$ whose values at time n and $n+1$ may be related via a Taylor series expansion:

$$u^{n+1} = u^n + \Delta t \left(\frac{\partial u}{\partial t}\right)^n + \frac{\Delta t^2}{2}\left(\frac{\partial^2 u}{\partial t^2}\right)^n + \cdots \tag{3.4}$$

Rearranging these terms it is thus fairly easy to conclude that

$$\left|u^{n+1} - u^n\right| \approx \mathcal{O}(\Delta t) \tag{3.5}$$

or, in other words, for sufficiently small time steps Δt, the solution at time $n+1$ will be relatively close to that at time n. This naturally suggests using the previous time step's

solution as the initial guess for the current time step [63]:

$$\{e\}_{(0)}^{n+1} = \{e\}^{n}. \tag{3.6}$$

Such a choice should not only result in the convergence of the Newton-Raphson algorithm, but convergence to the correct solution in an efficient manner.

Despite the availability of a good initial guess, it should be noted however that this is not in and of itself a guarantee of convergence to the correct solution. Owing to this, when performing the iterations it is important to not only monitor the relative change in the solution as a stopping criteria, but also the residual of the original nonlinear system $\{F\}$. Doing so may thus help permit the detection of convergence to an incorrect solution, such as convergence to a local minimum rather than a global one.

3.2 Elemental Matrix Properties

Many aspects of the implementation of a numerical algorithm depend upon the properties of its underlying matrix quantities. In consequence, here the properties of the matrices introduced in the previous chapter will be analyzed to help determine, among other things, which matrix solvers are most suitable for the solution of the Jacobian system.

3.2.1 Instantaneous Nonlinearity

Recall that in the case of an *instantaneous* nonlinearity, the key matrices of the mixed and VWE formulations were defined as:

$$[M^{(e)}]_{ij} = [T^{(e)}]_{ij} = \int_{\Omega^e} \epsilon_0\big(1 + \chi^{(1)} + \chi^{(3)}E^2\big)\vec{W}_i^{(1)(e)} \cdot \vec{W}_j^{(1)(e)}\, d\Omega \tag{3.7}$$

$$[S^{(e)}]_{ij} = [C^{(e)}][M_f^{(e)}][C^{(e)}]_{ij}^T = \int_{\Omega^e} \frac{1}{\mu}\big(\nabla \times \vec{W}_i^{(1)(e)}\big) \cdot \big(\nabla \times \vec{W}_j^{(1)(e)}\big)\, d\Omega \tag{3.8}$$

$$[J^{(e)}]_{ij} = [T^{(e)}]_{ij} + \frac{\Delta t^2}{4}[S^{(e)}]_{ij} + \int_{\Omega^e} 2\epsilon_0\chi^{(3)}\big(\vec{W}_i^{(1)(e)} \cdot \vec{E}\big)\big(\vec{W}_j^{(1)(e)} \cdot \vec{E}\big)\, d\Omega. \tag{3.9}$$

Firstly, it is fairly straightforward to conclude that an interchange of the indices i and j in each of these matrices leaves the result unchanged. Hence, it can be said that each of them is *symmetric*, and so, for example:

$$[M^{(e)}] = [M^{(e)}]^T. \tag{3.10}$$

Additionally, suppose the $[M^{(e)}]$ matrix above is both left and right multiplied by an arbitrary vector $\{x\}$ to yield a scalar quantity as follows:

$$\{x\}^T[M^{(e)}]\{x\} = \sum_i \sum_j x_i x_j \int_{\Omega^e} \epsilon_0\big(1 + \chi^{(1)} + \chi^{(3)}E^2\big)\vec{W}_i^{(1)(e)} \cdot \vec{W}_j^{(1)(e)}\, d\Omega \tag{3.11}$$

$$= \int_{\Omega^e} \epsilon_0\big(1 + \chi^{(1)} + \chi^{(3)}E^2\big)\Big(\sum_i x_i\vec{W}_i^{(1)(e)}\Big) \cdot \Big(\sum_j x_j\vec{W}_j^{(1)(e)}\Big)\, d\Omega. \tag{3.12}$$

Defining a new vector $\vec{\alpha}$ as

$$\vec{\alpha} = \sum_k x_k\vec{W}_k^{(1)(e)} \tag{3.13}$$

permits equation (3.12) equally be written as:

$$\{x\}^T[M^{(e)}]\{x\} = \int_{\Omega^e} \epsilon_0\big(1 + \chi^{(1)} + \chi^{(3)}E^2\big)\ \vec{\alpha}\cdot\vec{\alpha}\, d\Omega \tag{3.14}$$

$$= \int_{\Omega^e} \epsilon_0(1 + \chi^{(1)} + \chi^{(3)}E^2)\ |\vec{\alpha}|^2\ d\Omega. \tag{3.15}$$

For any passive material, the susceptibilities $\chi^{(1)}$ and $\chi^{(3)}$ are strictly non-negative, as is the quantity E^2 for any value of the electric field $\vec{E}$. Moreover, since the 1-form basis functions are linearly independent, $\vec{\alpha}$ can only be zero when $\{x\}$ itself is zero:

$$\{x\} \neq 0 \Leftrightarrow \vec{\alpha} \neq 0. \tag{3.16}$$

Since the magnitude $|\vec{\alpha}|^2$ is strictly nonzero for nonzero $\vec{\alpha}$, the result is that for any given nonzero vector $\{x\}$, the entire integrand of equation (3.12) is strictly non-negative, meaning:

$$\{x\}^T[M^{(e)}]\{x\} > 0\ \forall\ \{x\} \in \mathbb{R}^n \setminus \{0\}. \tag{3.17}$$

Equation (3.17) is recognized as the definition of *positive definiteness* for a matrix [89]. Thus, combining this with the earlier result of equation (3.10) shows that the $[M^{(e)}]$ matrix is *symmetric positive definite*. In fact, repeating a similar analysis for the remaining two matrices in (3.8) and (3.9) reveals they also share this property.

While the above proofs only demonstrate that the *local* matrices are symmetric positive definite, it can in fact be shown that this is a sufficient condition for the resulting *global* matrices to *also* possess these properties (see Appendix A). This therefore represents quite an important result, as the property of being symmetric positive definite not only has implications for stability (as discussed later), but also permits the use of highly efficient matrix solving algorithms such as the widely popular Cholesky Factorization and Preconditioned Conjugate Gradient (PCG) methods [90].

3.2.2 Dispersive Nonlinearity

In the case of a dispersive nonlinearity, the $[S]$ matrix is unchanged, however the $[K]$ and $[J]$ matrices are modified to include factors due to the nonlinear convolution:

$$[K^{(e)}]_{ij} = \int_{\Omega^e} \Big(a_0 + \epsilon_0 \alpha \chi^{(3)} E^2 + (1-\alpha)\epsilon_0 \chi^{(3)}(h_0 E^2 + \mathcal{G}_1)\Big) \vec{W}_i^{(1)(e)} \cdot \vec{W}_j^{(1)(e)} \, d\Omega \tag{3.18}$$

$$[J^{(e)}]_{ij} = [K^{(e)}]_{ij} + \frac{\Delta t^2}{4}[S^{(e)}]_{ij} + \int_{\Omega^e} 2\epsilon_0 \chi^{(3)}\big(\alpha + (1-\alpha)h_0\big)(\vec{W}_i^{(1)(e)} \cdot \vec{E})(\vec{W}_j^{(1)(e)} \cdot \vec{E}) \, d\Omega \tag{3.19}$$

As was the case in the previous subsection, an interchange of the indices in equations (3.18) and (3.19) leaves the result unchanged, meaning the $[K^{(e)}]$ and $[J^{(e)}]$ matrices are still symmetric.

The question of positive-definiteness is, however, a bit more complex. Applying a similar procedure to that outlined in the previous subsection to the $[K^{(e)}]$ matrix above now requires knowledge about the quantities a_0, h_0, and $\mathcal{G}_1$ (from Subsection 1.2.1 it is assumed that $0 \leq \alpha \leq 1$). As was detailed in Subsection 2.2.2 the constants a_0 and h_0 result from applying the z-transform to the linear and nonlinear convolutions, respectively. These constants are derived from the permittivity function $\epsilon_L(t)$ and the nonlinear term $g(t)$, as well as the temporal step size Δt, and thus depend upon the dispersive models being used. For many practical models it can be verified that the constants a_0 and h_0 will be strictly greater than zero for any positive value of Δt. Indeed, this was the case for the sample materials studied in Chapter 2.

Unfortunately, the behaviour of the $\mathcal{G}_1$ term is harder to characterize. In essence, determining whether $h_0 E^2 + \mathcal{G}_1$ will remain positive for all times is tantamount to determining whether the same is true of the convolution which it approximates. Sadly, for many materials of interest it is not immediately obvious whether the quantity $g(t) * E^2$ will be strictly positive for all time, and is thus hard to prove whether the $[K]$ and $[J]$ matrices of equations (3.18) and (3.19) are always positive definite. Despite this theoretical difficulty, numerical

studies have thus far suggested that these matrices do, in fact, remain symmetric positive definite throughout the course of a computation, though naturally this cannot be taken as definitive proof.

3.3 Evaluation of Elemental Matrices

For a *linear* FETD method, the $[M]$ and $[T]$ matrices do not change with time, as the permittivity is a simple scalar constant:

$$[M^{(e)}]_{ij} = [T^{(e)}]_{ij} = \int_{\Omega^e} \epsilon \, \vec{W}_i^{(1)(e)} \cdot \vec{W}_j^{(1)(e)} \, d\Omega. \tag{3.20}$$

Since the permittivity in the above equation is constant, the only spatially varying part of the integrand in (3.20) are the 1-form basis functions. At first it may seem a complex task to integrate these basis functions over a triangular or tetrahedral element, however, due to the properties of these basis functions the result is often surprisingly obtainable in closed form. For instance, in two dimensions it can be shown that [91]:

$$\int_{\Omega^e} (L_1^{(e)})^l (L_2^{(e)})^m (L_3^{(e)})^n \, d\Omega = \frac{l! \; m! \; n!}{(l+m+n+2)!} 2\Delta^{(e)} \tag{3.21}$$

where $L_i^{(e)}$ are the scalar interpolary basis functions from which the vector 1-forms are constructed, and $\Delta^{(e)}$ is the triangle area. Exact expressions for each entry of the $[M^{(e)}]$ and $[T^{(e)}]$ matrices can thus be found and tabulated, making the evaluation of the elemental matrices quite straightforward and efficient in the linear case.

Unfortunately, in the nonlinear case the dependence of the permittivity on the electric field, and thus indirectly on the spatial coordinates, precludes the use of equation (3.21). As a result, closed form expressions for the nonlinear $[M^{(e)}]$ and $[T^{(e)}]$ matrices are generally unavailable, meaning they must instead be evaluated numerically.

One approach to the numerical evaluation of the matrices in (3.20) is via Gaussian

Quadrature, in which integrals over the triangle, for example, may be approximated as [20]:

$$\iint_{\Omega^e} f(n_1, n_2, n_3)\, d\Omega \approx \sum_i w_i f(n_{1i}, n_{2i}, n_{3i}) \tag{3.22}$$

where (n_1, n_2, n_3) are simplex coordinates within the triangle [20], (n_{1i}, n_{2i}, n_{3i}) are the corresponding sampling points, and w_i their respective weights. The number of sampling points and weights used depends on the desired level of accuracy, with tabulated values of abscissae and weights available to quite high order [92]. The numerical integration scheme provided by equation (3.22) has the benefit of being straightforward to implement, and can achieve, in principle, any desired level of accuracy.

While the numerical integration of (3.22) may be inexpensive for any one nonlinear element, in aggregate the overhead imposed for the entire domain can quickly skyrocket. Indeed, for the worst case scenario of a domain completely filled with nonlinear media, the need to numerically integrate the local matrices of every single element, compounded with the fact that these matrices change and must be recomputed multiple times during Newton-Raphson and time-stepping, results in an immense computational burden. It is not entirely surprising, therefore, that nonlinear simulations generally dwarf their linear counterparts in terms of required execution time and resources.

This immense computational burden imposed by nonlinearity can unsurprisingly result in a significant barrier to entry. To mitigate this obstacle, a few different approaches are possible. One, as detailed and implemented in Chapter 5, is to attempt to exploit parallelism to accelerate these nonlinear algorithms. Another method, detailed below, has not been implemented or tested in this thesis, but is suggested as a potentially promising future improvement.

The main issue with the $[M^{(e)}]$ and $[T^{(e)}]$ matrices within a nonlinear simulation is in the dependence of the permittivity on the electric field. However, since the resulting permittivity is effectively a function of space and time, it can itself thus potentially be approximated via

scalar elemental interpolary basis functions as:

$$\epsilon(\vec{x},t) \approx \sum_k \tilde{\epsilon}_k(t) L_k^{(e)} \tag{3.23}$$

where $\tilde{\epsilon}_k$ are time-dependent expansion weights. Substituting this expression into the definition of the local $[M^{(e)}]$ and $[T^{(e)}]$ matrices would then yield:

$$[M^{(e)}]_{ij} = [T^{(e)}]_{ij} = \int_{\Omega^e} \epsilon \, \vec{W}_i^{(1)(e)} \cdot \vec{W}_j^{(1)(e)} \, d\Omega \tag{3.24}$$

$$= \int_{\Omega^e} \sum_k \tilde{\epsilon}_k L_k^{(e)} \vec{W}_i^{(1)(e)} \cdot \vec{W}_j^{(1)(e)} \, d\Omega \tag{3.25}$$

$$= \sum_k \tilde{\epsilon}_k \int_{\Omega^e} L_k^{(e)} \vec{W}_i^{(1)(e)} \cdot \vec{W}_j^{(1)(e)} \, d\Omega \tag{3.26}$$

$$= \sum_k \tilde{\epsilon}_k [M_k{}^{(e)}]_{ij}. \tag{3.27}$$

The expression for the matrices obtained in equation (3.26) is attractive since the integrand is now only a function of the scalar interpolary functions. Therefore, using equation (3.21), the integral in (3.26) is constant and obtainable in closed form, alleviating the need to use Gaussian Quadrature.

In fact, just such a technique was used in conjunction with the explicit auxiliary differential equation method in [67], however several challenges remain in adopting such a technique to *implicit* z-transform FETD methods. The first is that a relationship is ultimately required between the global permittivity weights $\tilde{\epsilon}_k$ and the working unknown of interest, the electric field weights $\{e\}$. The second is in the derivation of the Jacobian associated with the matrices in (3.26), which will naturally depend upon the relationship between $\tilde{\epsilon}_k$ and $\{e\}$. However, should the resulting Jacobian also be obtainable in closed form, the proposed approximation has the potential to alleviate a tremendous amount of computational resources, since not only are analytic expressions much faster to evaluate than Gaussian Quadrature, but also would not change over time.

3.4 Auxiliary Variable Updating

In the last chapter it was shown how the introduction of dispersion (either linear or nonlinear) resulted in convolutions whose update equations were implemented via the z-transform and simplified by the introduction of the Transposed Direct Form II technique. This resulted in a set of auxiliary variables which accumulate field and convolution values as the simulation progresses.

While the update procedure for the convolution variables in the linear case is relatively straightforward, a subtlety arises in the update equations for the nonlinear case. Recall that the nonlinear convolution:

$$\mathcal{B} = g(t) * E^2(t) \tag{3.28}$$

is approximated via the z-transform method as:

$$\mathcal{B}^n = h_0(E^2)^n + \mathcal{G}_1^{n-1} \tag{3.29}$$

for which the $[K]$ matrix becomes:

$$[K^{(e)}]_{ij}^{n+1} = \int_{\Omega^e} \Big(a_0 + \epsilon_0 \chi^{(3)} (E^2)^{n+1} + (1-\alpha)\epsilon_0 \chi^{(3)} (h_0 (E^2)^{n+1} + \mathcal{G}_1^n) \Big) \vec{W}_i^{(1)(e)} \cdot \vec{W}_j^{(1)(e)} d\Omega. \tag{3.30}$$

The complicating factor within equation (3.30), which differentiates it from the linear case, is that here the auxiliary variable $\mathcal{G}_1$ is found *within* the elemental integral. The main consequence of this is that, when performing the Gaussian Quadrature evaluation of the integral in (3.22), the value of $\mathcal{G}_1$ will be required at each quadrature or sample point within each element, for as many past time steps are required for the medium's dispersive order. Hence, at every time step, each of the auxiliary variables at each quadrature point must be advanced in time.

While not as significant as the numerical integration overhead, the fact that the update equation overhead is multiplied by the number of quadrature points nevertheless results in a

non-negligible burden. It is possible, however, that by adopting a procedure similar to that proposed in the last subsection and expanding the nonlinear convolution in terms of scalar basis functions, some of this overhead may be alleviated. Not only would this allow closed-form expressions for the matrices, but might also simplify the auxiliary variable updating by only needing to store and advance the expansion weights, at the cost of now interpolating the convolution at each quadrature point. However, many of the issues identified with interpolating the permittivity in the last section persist here, and so this approach has not been adopted in this thesis. Nevertheless, it may prove an interesting avenue of approach for future study.

3.5 Stability

When formulating any kind of numerical method or algorithm an important consideration is that of stability. Indeed, in the previous chapters the notion of numerical stability was mentioned several times to distinguish the methods derived in this thesis from existing methods.

Depending on context, the notion of stability can have different meanings. However, roughly speaking, a numerical method being *stable* results in the numerical solution to a homogeneous (source-free) problem being bounded by some limit for all time steps. For example, the most common choice usually defines stability as [93]:

$$\left|\{u\}^{n+1}\right| \leq C\left|\{u\}^{0}\right| \tag{3.31}$$

where $\{u\}$ is the numerical solution, C is some non-negative constant, and as before $|\cdot|$ is the magnitude or Euclidean norm. In essence, this definition of stability forbids the numerical solution from exhibiting exponential growth, and for linear problems equates to the numerical problem being *well-posed*. To demonstrate this, suppose some numerical algorithm obtains

$\{u\}^{n+1}$ from $\{u\}^n$ via the application of some operator $A(\cdot)$, such that:

$$\{u\}^{n+1} = A(\{u\}^n) \tag{3.32}$$

$$= A^n(\{u\}^0). \tag{3.33}$$

For the method represented by $A(\cdot)$ to be termed well-posed, small perturbations in the initial condition $\{u\}^0$ should result in a comparably small change in the solution at time $\{u\}^{n+1}$ [94]:

$$\left|\{\tilde{u}\}^{n+1} - \{u\}^{n+1}\right| \leq C\left|v^0\right| \tag{3.34}$$

where $\{\tilde{u}\}^{n+1}$ is the solution obtained from the perturbed initial condition

$$\{\tilde{u}\}^0 = \{u\}^0 + \{v\}^0. \tag{3.35}$$

This definition is the same as that used for well-posedness in partial differential equations.

If the operator $A(\cdot)$ is *linear*, the criteria in 3.34 can also be expressed as:

$$\left|\{\tilde{u}\}^{n+1} - \{u\}^n\right| \leq C\left|\{v\}^0\right| \tag{3.36}$$

$$\left|A^n(\{\tilde{u}\}^0) - A^n(\{u\}^0)\right| \leq C\left|\{v\}^0\right| \tag{3.37}$$

$$\left|A^n(\{u\}^0) + A^n(\{v\}^0) - A^n(\{u\}^0)\right| \leq C\left|\{v\}^0\right| \tag{3.38}$$

$$\left|A^n(\{v\}^0)\right| \leq C\left|\{v\}^0\right| \tag{3.39}$$

$$\left|\{v\}^{n+1}\right| \leq C\left|\{v\}^0\right| \tag{3.40}$$

which is precisely the original stability criteria stipulated in (3.31). Hence, in the case of a *linear* method, the stability criteria in (3.31) directly implies that condition (3.34) is satisfied.

Such a consistent definition of stability is important, as perturbations to the original differential equation, either due to discretization or finite floating point machine precision, could otherwise result in solutions which deviate substantially from the true solutions and non-

physically "explode" or "blow-up", meaning they become unbounded. Having a numerical method which is stable at a given operating point thereby guarantees that such perturbations do not grow, but rather remain bounded. In fact, for a consistent linear Finite-Difference scheme, due to the Lax equivalence theorem, stability is a sufficient condition to guarantee the solution converges, meaning it approaches the true solution as the grid is refined [95].

The stability of a given numerical algorithm can generally be classified as either *conditionally* stable or *unconditionally* stable. As the name implies, conditional stability dictates that there is a certain set of constraints or criteria which must be satisfied for the method to be stable. For Finite-Element analysis, these constraints are generally related to the temporal discretization parameter Δt, as well as the global matrices (themselves dependent on the spatial discretization, i.e. on the size of the elements in the mesh). For instance, when discretized via the central difference method, the linear VWE FETD method is conditionally stable with stability criterion given by [20]:

$$\Delta t \leq \frac{2}{\sqrt{\rho([T]^{-1}[S])}} \tag{3.41}$$

where $\rho(\cdot)$ denotes the spectral radius. In contrast, an unconditionally stable method is one in which there is no restriction on the temporal step size Δt in order to ensure the stability criteria of equation (3.31) is satisfied. For example, when the Newmark-β method is applied to linear FETD problems, it is unconditionally stable for $\beta \geq 1/4$ [20, 96].

While unconditionally stable algorithms tend to be more computationally expensive, this is mitigated by their ability to use larger time intervals and therefore fewer overall steps for the same amount of simulated time. Such methods are particularly attractive for nonlinear algorithms, such as those presented in the previous chapter, due to the constantly changing and evolving nature of the $[M]$, $[T]$, and $[J]$ matrices.

Unfortunately, while many methods have been devised to analyze the stability of linear FETD methods, the same cannot generally be said for their nonlinear counterparts. In fact,

analyzing the stability of nonlinear methods can pose a significant challenge. This difficulty stems, in part, from the fact that for a *nonlinear* operator $A(\cdot)$, the stability condition in (3.31) no longer directly implies (3.34) is satisfied. In other words, a bounded nonlinear operator is not necessarily continuous as it is in the linear case, meaning that additional care must be taken to avoid ambiguity in what is meant by a nonlinear method being termed *stable*. Moreover, there is currently no nonlinear analog of the Lax equivalence theorem, so that even if a nonlinear method is stable in the sense of (3.31) or (3.34) (or both), the solution cannot be guaranteed to converge to and correctly approximate the true solution [97]. Hence, while it is hoped that the unconditional stability of a linear method may be maintained when generalized to nonlinear problems, there is little guarantee that this will occur.

While attempts to prove the unconditional stability of the algorithms derived in Chapter 2 have so far been unsuccessful, two promising pieces of evidence do currently support the notion that these algorithms may in fact be numerically stable. The first is the result of empirical numerical studies, which demonstrate bounded solutions for large values of Δt. The second, which is of more theoretical interest, is related to the evolution of the discretized electromagnetic energy stored in the fields during the solution process. More specifically, in the next section it will be shown that the algorithms derived in the previous chapter conserve energy exactly, a criteria likely necessary, but which may not be sufficient, for unconditional stability.

3.6 Conservation of Energy

3.6.1 Linear Media

For any linear electromagnetic system, the energy density may be expressed as [2]:

$$u = u_m + u_e = \frac{1}{2}\vec{B}\cdot\vec{H} + \frac{1}{2}\vec{D}\cdot\vec{E} = \frac{1}{2\mu}|\vec{B}|^2 + \frac{\epsilon}{2}|\vec{E}|^2 \qquad \left(\frac{J}{m^3}\right) \tag{3.42}$$

where u_m and u_e are the magnetic and electric contributions, respectively, and from which the total energy is obtained via spatial integration:

$$U = U_m + U_e = \frac{1}{2}\int_\Omega \frac{1}{\mu}|\vec{B}|^2\, d\Omega + \frac{1}{2}\int_\Omega \epsilon\,|\vec{E}|^2\, d\Omega \qquad (J). \tag{3.43}$$

If the mixed Finite-Element approximation of the electric and magnetic fields in (2.4) and (2.5) are substituted into the total energy expression above, a straightforward comparison with the matrices of equations (2.19) and (2.22) shows that, after global assembly, the equivalent total numerical energy is given by:

$$U = U_m + U_e = \frac{1}{2}\{b\}^T[M_f]\{b\} + \frac{1}{2}\{e\}^T[M]\{e\} \tag{3.44}$$

which may also be written in the form of a partitioned matrix-vector product as:

$$U = \frac{1}{2}\begin{bmatrix}\{b\}^T & \{e\}^T\end{bmatrix}\begin{bmatrix}[M_f] & 0\\ 0 & [M]\end{bmatrix}\begin{bmatrix}\{b\}\\ \{e\}\end{bmatrix} \triangleq \frac{1}{2}\{y\}^T[\overline{M}]\{y\}. \tag{3.45}$$

Two interesting properties about the numerical energy in (3.45) should be noted. The first is that U can be shown to be conserved exactly by the *linear* Crank-Nicolson mixed

FETD method. The second is that it can also be considered as inducing an *energy norm*:

$$\|\{y\}\|_U = \sqrt{\frac{1}{2}\{y\}^T[\overline{M}]\{y\}} \tag{3.46}$$

since the quantity in (3.46) can be shown to obey the following three defining norm properties [98]:

1. $\|\alpha u\| = |\alpha|\|u\|$
2. $\|u + v\| \leq \|u\| + \|v\|$
3. $\|u\| \geq 0 \quad \text{with} \quad \|u\| = 0 \quad \text{if and only if} \quad u = 0$

where u and v are arbitrary members of a vector space, and α some arbitrary constant.

The first criteria is easily verified directly from the properties of matrix-vector algebra. The second criteria (the triangle inequality) can be shown to follow from the third condition via the Cauchy-Schwartz inequality, provided that the $[M_f]$ and $[M]$ matrices are symmetric. Lastly, the third criteria can be found to be satisfied if the $[M_f]$ and $[M]$ matrices are positive definite. Since these matrices are indeed symmetric positive definite in *linear* mixed FETD, equation (3.45) not only represents the physical electromagnetic energy, but can also be considered as inducing a valid energy norm. As will be shown next, the fact that (3.46) defines a valid norm will have a significant impact on the method's stability.

If the Finite-Element formulation of a homogeneous (source-free) problem has non-increasing energy at each subsequent time step:

$$U^{n+1} \leq U^n \tag{3.47}$$

then it is relatively straightforward to conclude that:

$$U^{n+1} \leq U^0 \tag{3.48}$$

or, equivalently,

$$\|\{y\}^{n+1}\|_U \leq \|\{y\}^0\|_U. \tag{3.49}$$

Moreover, due to the equivalence of norms in a finite-dimensional vector space [98] the energy and Euclidean norms may be related by:

$$C_1\|\{y\}\|_U \leq |\{y\}| \leq C_2\|\{y\}\|_U \tag{3.50}$$

where C_1 and C_2 are positive constants. Combining (3.48) and (3.50) it is fairly straightforward to thus conclude that:

$$|\{y\}^{n+1}| \leq C|\{y\}^0| \tag{3.51}$$

where C is likewise a positive constant related to C_1 and C_2, meaning equation (3.51) above is precisely the stability constraint of equation (3.31). Alternatively, by substituting in the definition of the $\{y\}$ vector from equation (3.45) into the above one can also obtain:

$$|\{e\}^{n+1}|^2 + |\{b\}^{n+1}|^2 \leq C^2\Big(|\{e\}^0|^2 + |\{b\}^0|^2\Big). \tag{3.52}$$

The alternative relation obtained in equation (3.52) can in essence be thought of as a two variable equivalent of the stability condition stated in equation (3.31), and precludes the possibility of the solutions growing exponentially without bound. In other words, if a method can be shown to conserve energy, and that energy induces a valid norm, then the method can be said to be stable in the sense of equation (3.31) and thus also in the sense of (3.34) if linear. In fact, as long as a numerical method can be shown to conserve *some* norm, even if it is not representative of the physical energy, the above arguments still hold. This is the case in the linear VWE Newmark-β method, for example, which for $\beta = 1/4$ preserves the following norm exactly, regardless of the time step size [57]:

$$\|\{y\}\|_U = \sqrt{\frac{1}{2}\{e\}^T[S]\{e\} + \frac{1}{2}\frac{d\{e\}^T}{dt}[T]\frac{d\{e\}}{dt}} \tag{3.53}$$

and is thus unconditionally stable.

3.6.2 Nonlinear Mixed FETD

In the case of an electrically nonlinear medium, the expression for the electromagnetic energy becomes more complex. By integrating Poynting's Theorem [2], it can be shown that the general expression for the electromagnetic energy density in an electrically nonlinear region is given by:

$$u = u_m + u_e = \frac{1}{2\mu}\vec{B}\cdot\vec{B} + \int_0^{\vec{D}} \vec{E}\cdot d\vec{D}' \tag{3.54}$$

for which the total energy in a specified region is then given by

$$U = \frac{1}{2}\int_\Omega \frac{1}{\mu}\vec{B}\cdot\vec{B}\, d\Omega + \int_\Omega \int_0^{\vec{D}} \vec{E}\cdot d\vec{D}'\, d\Omega. \tag{3.55}$$

Since the displacement field is generally considered the dependent variable, the integral within the energy density expression can also be written in terms of the co-energy [38]:

$$\int_0^{\vec{D}} \vec{E}\cdot d\vec{D}' = \vec{E}\cdot\vec{D} - \int_0^{\vec{E}} \vec{D}\cdot d\vec{E}' \tag{3.56}$$

such that the total energy can equivalently be found via:

$$U = \frac{1}{2}\int_\Omega \frac{1}{\mu}\vec{B}\cdot\vec{B}\, d\Omega + \int_\Omega \left(\vec{E}\cdot\vec{D} - \int_0^{\vec{E}} \vec{D}\cdot d\vec{E}'\right) d\Omega. \tag{3.57}$$

For example, consider an instantaneous nonlinearity in which

$$\vec{D} = \epsilon_0 \chi^{(3)} E^2 \vec{E}. \tag{3.58}$$

If the fields are initially zero, the associated nonlinear electric energy density is found to be:

$$u_e = \vec{D} \cdot \vec{E} - \int_0^{\vec{E}} \vec{D} \cdot d\vec{E}' \tag{3.59}$$

$$= \epsilon_0 \chi^{(3)} E^4 - \frac{1}{4} \epsilon_0 \chi^{(3)} E^4 \tag{3.60}$$

$$= \frac{3}{4} \epsilon_0 \chi^{(3)} E^4. \tag{3.61}$$

Unfortunately, contrary to the linear case, the expression for the nonlinear energy in equation (3.61) is not preserved when passing to a discretized space. Since the integral in the nonlinear electric energy density in equation (3.54) implicitly depends on time via the electric and displacement fields, the numerical equivalent within the simulation naturally depends upon the temporal discretization chosen.

In the case of mixed Crank-Nicolson FETD, the temporal discretization is equivalent to an application of the well-known trapezoidal rule:

$$\int_a^b f(x)\, dx = (a - b) \left(\frac{f(a) + f(b)}{2} \right). \tag{3.62}$$

For a linear medium, this trapezoidal approximation is enough to obtain the exact value of the integral, thus explaining the similarity between the continuous and discrete energy expressions in that case. In contrast, due to the time-stepping nature of the numerical method, the integration in the nonlinear case can in essence be thought of as occurring in discrete time intervals:

$$\int_0^{\vec{D}} \vec{E} \cdot d\vec{D}' = \int_0^{\vec{D}^1} \vec{E} \cdot d\vec{D}' + \int_{\vec{D}^1}^{\vec{D}^2} \vec{E} \cdot d\vec{D}' + \cdots + \int_{\vec{D}^n}^{\vec{D}^{n+1}} \vec{E} \cdot d\vec{D}' \tag{3.63}$$

where each integral is approximated by the trapezoidal rule in (3.62). Thus, the equivalent discretized nonlinear electric energy is defined by the following recurrence relation:

$$U_e^{n+1} = U_e^n + \frac{1}{2} \int_\Omega (\vec{D}^{n+1} - \vec{D}^n)(\vec{E}^{n+1} + \vec{E}^n)\, d\Omega \tag{3.64}$$

from which an expression for the nonlinear electric energy at time step $n+1$ may be obtained:

$$U_e^{n+1} = \frac{1}{2}\sum_{k=0}^{n}\int_{\Omega}(\vec{D}^{k+1} - \vec{D}^{k})(\vec{E}^{k+1} + \vec{E}^{k})\, d\Omega. \tag{3.65}$$

To determine the Finite-Element representation of the above energy, the basis function representation of the electric field in (2.4) can again be substituted, as well as the appropriate constitutive relation. The fully discretized form of the recurrence relation in (3.64) is thus:

$$\begin{aligned} U_e^{n+1} = U_e^{n} &+ \frac{1}{2}(\{e\}^{n+1})^T[M]^{n+1}\{e\}^{n+1} \\ &+ \frac{1}{2}(\{e\}^{n+1})^T([M]^{n+1} - [M]^{n})\{e\}^{n} - \frac{1}{2}(\{e\}^{n})^T[M]^{n}\{e\}^{n} \end{aligned} \tag{3.66}$$

from which the total energy within the mixed FETD method at time $n+1$ is given by:

$$\begin{aligned} U^{n+1} = U_e^{0} &+ \frac{1}{2}(\{b\}^{n+1})^T[M_f]\{b\}^{n+1} + \frac{1}{2}(\{e\}^{n+1})^T[M]^{n+1}\{e\}^{n+1} \\ &- \frac{1}{2}(\{e\}^{0})^T[M]^{0}\{e\}^{0} + \frac{1}{2}\sum_{k=0}^{n}(\{e\}^{k+1})^T\big([M]^{k+1} - [M]^{k}\big)\{e\}^{k}. \end{aligned} \tag{3.67}$$

Unfortunately, contrary to the linear case, it is not obvious if the quantity in equation (3.67) can be associated with or induce a valid norm. Indeed, due to the nonlinearity within the $[M]$ matrix the first norm property in (1) is violated. Moreover, rather than simply being a function of the present values of the electric and magnetic fields, the numerical energy depends upon all past values of the electric field. Hence, even if the $[M]$ matrix were strictly positive definite, it is unclear whether this would be sufficient to guarantee the summation term in (3.67) is non-negative. In consequence, even if the energy were bounded, the solution itself may not be. However, despite the mathematical difficulty in developing a norm for the nonlinear energy above, it *can* actually nonetheless be shown that the nonlinear mixed Crank-Nicolson FETD method of Chapter 2 conserves this energy quantity exactly.

To facilitate the following analysis, the *jump* $[\![\cdot]\!]$ and *average* $\langle\!\langle\cdot\rangle\!\rangle$ operators will be defined

as:

$$[\![u]\!] = u^{n+1} - u^n \tag{3.68}$$

$$\langle\!\langle u \rangle\!\rangle = \frac{u^{n+1} + u^n}{2}. \tag{3.69}$$

With this notation, the homogeneous mixed Crank-Nicolson Maxwell's Equations of (2.32) and (2.33) can be written as:

$$[\![\{b\}]\!] = -\Delta t[C]^T \langle\!\langle \{e\} \rangle\!\rangle \tag{3.70}$$

$$[\![[M]\{e\}]\!] = \Delta t[C][M_f] \langle\!\langle \{b\} \rangle\!\rangle. \tag{3.71}$$

Left multiplying equation (3.70) by $\langle\!\langle \{b\}^T \rangle\!\rangle [M_f]$ and taking the transpose yields:

$$[\![\{b\}^T]\!][M_f] \langle\!\langle \{b\} \rangle\!\rangle = -\Delta t \langle\!\langle \{e\}^T \rangle\!\rangle [C][M_f] \langle\!\langle \{b\} \rangle\!\rangle \tag{3.72}$$

where the symmetry of the $[M_f]$ matrix has been used. Noticing the similarities between the right-hand sides of equations (3.72) and (3.71) then permits the elimination of $\{b\}$ from the right-hand side of (3.72):

$$[\![\{b\}^T]\!][M_f] \langle\!\langle \{b\} \rangle\!\rangle = -\langle\!\langle \{e\}^T \rangle\!\rangle [\![[M]\{e\}]\!]. \tag{3.73}$$

Lastly, a straightforward expansion of the left-hand side of the above equation shows that:

$$[\![\{b\}^T]\!][M_f] \langle\!\langle \{b\} \rangle\!\rangle = \frac{1}{2}[\![\{b\}^T [M_f] \{b\}]\!] \tag{3.74}$$

whereas for the right-hand side

$$\langle\!\langle \{e\}^T \rangle\!\rangle [\![[M]\{e\}]\!] = \frac{1}{2}[\![\{e\}^T [M] \{e\}]\!] + \frac{1}{2}\{e\}^{n+1} [\![[M]]\!] \{e\}^n. \tag{3.75}$$

Combining (3.73), (3.74), and (3.75) finally produces:

$$\frac{1}{2}[\![\{b\}^T[M_f]\{b\}]\!] + \frac{1}{2}[\![\{e\}^T[M]\{e\}]\!] + \frac{1}{2}\{e\}^{n+1}[\![[M]]\!]\{e\}^n = 0. \tag{3.76}$$

The first term on the left-hand side of equation (3.76) can be identified as the change in magnetic energy $[\![U_m]\!]$. As for the remaining two terms, a comparison with the recurrence relation of equation (3.66) reveals them to be equal to the change in the nonlinear electric energy, $[\![U_e]\!]$. As a result, equation (3.76) can also be written as:

$$[\![U]\!] = [\![U_m]\!] + [\![U_e]\!] = 0 \tag{3.77}$$

from which it is concluded that the nonlinear mixed Crank-Nicolson FETD implementation conserves the total electromagnetic energy exactly, in the absence of source terms.

This result is physically significant in that any true solution to Maxwell's Equations must also conserve this energy exactly. However, as mentioned, due to difficulties in showing that (3.67) induces a valid norm, nothing can currently be said about the stability of the method in the sense of either equation (3.31) or (3.34). Nevertheless, despite these difficulties, the notion that the overall energy of the simulation is non-increasing does elicit some degree of confidence.

3.6.3 Nonlinear VWE FETD

In the case of the Vector Wave Equation, the numerical electromagnetic energies of the previous section are not explicitly conserved in general. As mentioned earlier, this is due to the homogeneous VWE supporting solutions of the form $\vec{E} = -(at + b)\nabla\phi$ whose energy increases quadratically over time. Instead, for *linear* media, it can be shown that the

continuous quantity:

$$\tilde{U}_L = \frac{1}{2}\int_\Omega \frac{1}{\mu}(\nabla \times \vec{E})^2 \, d\Omega + \frac{1}{2}\int_\Omega \epsilon \left(\frac{\partial \vec{E}}{\partial t}\right)^2 d\Omega \tag{3.78}$$

$$= \tilde{U}_m + \tilde{U}_e \tag{3.79}$$

is always conserved by the VWE, whereas the discrete equivalent:

$$\tilde{U}_L = \frac{1}{2}\{e\}^T[S]\{e\} + \frac{1}{2}\frac{d\{e\}^T}{dt}[T]\frac{d\{e\}}{dt} \tag{3.80}$$

is conserved by the *linear* Newmark-β FETD VWE formulation [57]. A dimensional analysis of the above quantity shows that it has units of J/s^2, consistent with the quadratically increasing energy obtainable by the late-time growth solution in which $d^2U/dt^2 \neq 0$. It is thus not entirely surprising that this quantity should be conserved, with $d\tilde{U}/dt = d^3U/dt^3 = 0$. As mentioned previously, however, the fact that $\tilde{U}$ does not represent the actual physical energy does not preclude its utility in determining the stability of the method, as for linear media it still induces a valid norm.

In the continuous *nonlinear* setting, the quantity in (3.78) is no longer conserved, however the following quantity can be shown to be conserved by the nonlinear VWE instead:

$$\tilde{U} = \frac{1}{2}\int_\Omega \frac{1}{\mu}(\nabla \times \vec{E})^2 d\Omega + \psi \tag{3.81}$$

where

$$\psi = \int_\Omega \int_0^t \frac{\partial^2 \vec{D}}{\partial t'^2} \cdot \frac{\partial \vec{E}}{\partial t'} \, dt' d\Omega. \tag{3.82}$$

Indeed, taking the temporal derivative of the quantity in (3.81) yields:

$$\frac{d\tilde{U}}{dt} = \frac{1}{2}\int_\Omega \frac{1}{\mu}\frac{\partial}{\partial t}(\nabla \times \vec{E})^2 \, d\Omega + \int_\Omega \frac{\partial}{\partial t}\int_0^t \frac{\partial^2 \vec{D}}{\partial t'^2} \cdot \frac{\partial \vec{E}}{\partial t'} \, dt' d\Omega \tag{3.83}$$

$$= \int_\Omega \frac{1}{\mu}(\nabla \times \vec{E}) \cdot \left(\nabla \times \frac{\partial \vec{E}}{\partial t}\right) \, d\Omega + \int_\Omega \frac{\partial^2 \vec{D}}{\partial t^2} \cdot \frac{\partial \vec{E}}{\partial t} \, d\Omega. \tag{3.84}$$

Applying the vector identity (2.12) to the first term in (3.84) and grouping terms then results in:

$$\frac{d\tilde{U}}{dt} = \int_\Omega \left(\nabla \times \frac{1}{\mu}\nabla \times \vec{E} + \frac{\partial^2 \vec{D}}{\partial t^2}\right) \cdot \frac{\partial \vec{E}}{\partial t} \, d\Omega \tag{3.85}$$

where the bracketed quantity is easily identified as the original vector wave equation. Thus, if $\vec{E}$ and $\vec{D}$ are solutions to the source-free VWE, the above results in:

$$\frac{d\tilde{U}}{dt} = 0 \tag{3.86}$$

indicating the conservation of the quantity $\tilde{U}$.

Similarly to the mixed method, determining the discrete equivalent of $\tilde{U}$ depends upon the temporal discretization used, as ψ is a function of time. For the Newmark-β method with $\beta = 1/4$, this once again amounts to an application of the trapezoidal rule [20], such that (3.82) is approximated by the recurrence relation:

$$\psi^{n+1} = \psi^n + \frac{\Delta t}{2}\int_\Omega \left(\frac{\partial^2 \vec{D}}{\partial t^2}^{n+1} \cdot \frac{\partial \vec{E}}{\partial t}^{n+1} + \frac{\partial^2 \vec{D}}{\partial t^2}^{n} \cdot \frac{\partial \vec{E}}{\partial t}^{n}\right) d\Omega \tag{3.87}$$

which may be more succinctly written by making use of the jump and average operators from earlier, yielding:

$$[\![\psi]\!] = \Delta t \left\langle\!\!\left\langle \int_\Omega \frac{\partial^2 \vec{D}}{\partial t^2} \cdot \frac{\partial \vec{E}}{\partial t} \, d\Omega \right\rangle\!\!\right\rangle. \tag{3.88}$$

Lastly, as before, a straightforward substitution of the Finite-Element basis function expan-

sions for the electric field results in the fully discrete equivalents of ψ and $\tilde{U}$:

$$\psi^{n+1} = \psi^n + \Delta t \left\langle\!\!\left\langle \frac{d\{e\}^T}{dt} \frac{d^2}{dt^2} \Big([K]\{e\} \Big) \right\rangle\!\!\right\rangle \tag{3.89}$$

$$= \frac{\Delta t}{2} \sum_{k=0}^{n} \left(\left(\frac{d\{e\}^T}{dt} \frac{d^2}{dt^2} \big([K]\{e\} \big) \right)^{k+1} + \left(\frac{d\{e\}^T}{dt} \frac{d^2}{dt^2} \big([K]\{e\} \big) \right)^{k} \right) \tag{3.90}$$

$$\tilde{U}^{n+1} = \frac{1}{2} (\{e\}^{n+1})^T [S] \{e\}^{n+1} + \psi^{n+1}. \tag{3.91}$$

Similarly to the last section, it is not immediately obvious whether the discrete nonlinear quantity $\tilde{U}$ can be associated with a valid norm, due to the nonlinearity and complicated dependence on past field values. However, just as in the previous case, despite this mathematical difficulty, the nonlinear Newmark-β VWE method of Chapter 2 can be shown to conserve $\tilde{U}$ exactly.

Using the jump and average operators, the Newmark-β equations of (2.109) and (2.110) can be expressed as:

$$[\![u]\!] = \Delta t \left\langle\!\!\left\langle \frac{du}{dt} \right\rangle\!\!\right\rangle \tag{3.92}$$

$$\left[\!\!\left[\frac{du}{dt} \right]\!\!\right] = \Delta t \left\langle\!\!\left\langle \frac{d^2 u}{dt^2} \right\rangle\!\!\right\rangle. \tag{3.93}$$

Starting from the homogeneous semi-discrete VWE in (2.124) one can left-multiply by $d\{e\}^T/dt$ to obtain:

$$\frac{d\{e\}^T}{dt} \frac{d^2}{dt^2} \Big([K]\{e\} \Big) + \frac{d\{e\}^T}{dt} [S]\{e\} = 0 \tag{3.94}$$

to which the averaging operator may then be applied:

$$\left\langle\!\!\left\langle \frac{d\{e\}^T}{dt} \frac{d^2}{dt^2} \Big([K]\{e\} \Big) \right\rangle\!\!\right\rangle + \left\langle\!\!\left\langle \frac{d\{e\}^T}{dt} [S]\{e\} \right\rangle\!\!\right\rangle = 0. \tag{3.95}$$

Now, due to the symmetry of the $[S]$ matrix, the following identity can be easily verified:

$$\frac{d\{e\}^T}{dt}[S]\{e\} = \frac{d}{dt}\Big(\frac{1}{2}\{e\}^T[S]\{e\}\Big) \tag{3.96}$$

from which equation (3.95) can also be written as:

$$\Big\langle\!\Big\langle \frac{d\{e\}^T}{dt}\frac{d^2}{dt^2}\Big([K]\{e\}\Big)\Big\rangle\!\Big\rangle + \Big\langle\!\Big\langle \frac{d}{dt}\Big(\frac{1}{2}\{e\}^T[S]\{e\}\Big)\Big\rangle\!\Big\rangle = 0. \tag{3.97}$$

Multiplying through by Δt

$$\Delta t\Big\langle\!\Big\langle \frac{d\{e\}^T}{dt}\frac{d^2}{dt^2}\Big([K]\{e\}\Big)\Big\rangle\!\Big\rangle + \Delta t\Big\langle\!\Big\langle \frac{d}{dt}\Big(\frac{1}{2}\{e\}^T[S]\{e\}\Big)\Big\rangle\!\Big\rangle = 0 \tag{3.98}$$

the first term is recognized from (3.89) as the change in the nonlinear electric part, $[\![\psi]\!]$, whereas the second term is seen from (3.92) to be the change in the magnetic part, $[\![\tilde{U}_m]\!]$. Therefore, it is concluded that:

$$[\![\tilde{U}]\!] = [\![\tilde{U}_m]\!] + [\![\psi]\!] = 0 \tag{3.99}$$

and so the nonlinear Newmark-β VWE formulation conserves the quantity $\tilde{U}$ exactly.

Much as in the case of the mixed method the above result does not in and of itself prove the unconditional stability of the method neither in the sense of equation (3.31) nor (3.34), partly due to the difficulty in verifying whether (3.91) induces a valid norm. Moreover, due to the nature of nonlinear root finding and floating point precision, the numerical energy can still stray from its starting value. Nevertheless, the fact that the method exhibits the same conservation as the linear and continuous cases does, again, elicit some degree of confidence in its stability. Indeed, for all numerical studies undertaken in this thesis, no instabilities were ever encountered.

3.7 Summary

In this chapter several key details concerning the implementation and characteristics of the algorithms derived in the previous chapter were explored. Details concerning the Newton-Raphson method were addressed, including strategies for selecting good initial guesses which maximize likelihood and speed of convergence. Moreover, an analysis of the methods' nonlinear system matrices was performed to better understand how to optimize their solution. For instantaneous nonlinearities, the resulting matrices were shown to be symmetric positive definite, whereas for nonlinear dispersive media only the symmetry of the underlying matrices could be proven. Furthermore, some of the more subtle differences of implementing a nonlinear FETD solver were addressed, including the need to numerically evaluate elemental matrices and the updating of nonlinear dispersive auxiliary variables.

Lastly, the crucial notion of numerical stability was discussed. In particular, it was shown that the definition of stability in the nonlinear setting is more nuanced than for its linear counterpart and that the stability of a linear method cannot be assumed to be preserved when generalized. Despite this, however, it was successfully shown that both of the derived formulations have the desirable property of preserving the total numerical electromagnetic energy exactly, regardless of the time step size used. While this is an encouraging indication of the methods' stability, it cannot be taken as definitive proof due to difficulties in establishing an equivalent energy norm and the specifics of which particular definition of stability is used. Nevertheless, these promising developments point toward methods which are overall well-behaved.

Chapter 4

Perfectly Matched Layers

One of the intrinsic limitations of any numerical method lies in the fact that it must ultimately be executed on a finite computer system. As a result, computer simulations such as FETD are fundamentally limited in their scope and accuracy by the amount of computational resources available on the host machine. For finite problems, such limitations are often either inconsequential, mitigable, or circumventable. However, in contrast, many electromagnetic systems of practical interest operate in domains whose extent is theoretically infinite. This is the case, for example, with any kind of radiation problem in which an antenna is transmitting into the surrounding environment. Such a problem naturally poses a challenge for numerical methods as an infinite domain must now be represented by a finite-resource discretized numerical equivalent.

One obvious solution to this problem is to simply trim the infinite domain down to a much smaller finite one surrounding the object or area of interest, however, such a straightforward procedure would naturally alter the solution. Consider, for instance, if the domain were truncated via a Dirichlet or PEC boundary condition, then reflections from the newly truncated boundary would invariably result in a corruption of the solution on the interior. However, if the new truncated boundary were to instead be equipped with a system meant to absorb outward propagating waves, the solution on the interior would remain unchanged from

that which would have occurred in the original infinite domain. Such a system could even have additional practical uses for finite problems by restricting the simulation only to important sub-areas of interest, such as within the vicinity of a waveguide obstruction or defect. To date, many techniques based upon this approach have been successfully implemented, permitting not only the emulation of infinite domains, but also substantial computational savings due to reduced simulation area or volume.

In general, techniques for implementing absorbing boundaries can be broadly classified as falling into two main categories. The first is that of the so called Absorbing Boundary Conditions (ABC), in which either a "one-way" wave equation or an "annihilation operator" is enforced on the boundary. As the name implies, a one-way wave equation only supports wave propagation in one direction, which precludes any reflections when enforced on the boundary [99]. On the other hand, annihilation operators ensure the solution obeys the Sommerfeld Radiation Condition [100], resulting in exclusively outward propagating waves. In contrast, the second category, known as Perfectly Matched Layers (PML), creates a buffer region around the boundary which is not only tuned to the incident medium to prevent reflections, but also strongly artificially attenuates any propagating waves within it. When backed by a traditional PEC or other boundary condition, outward propagating waves must effectively travel through the PML twice before being able to re-enter the domain and are attenuated each time, resulting in very little energy bouncing back into the truncated domain [20].

While it is unlikely to have an antenna system radiating into an infinitely large nonlinear medium, such a system does have immediate applicability to waveguides, where the implementation of either ABCs or PMLs to reduce simulation size, and hence the number of required degrees of freedom and computation time, is highly attractive. For this reason, this chapter will focus upon the derivation and implementation of absorbing boundaries for use with nonlinear dispersive media. However, while both the ABC and PML methods have achieved noteworthy success for linear materials, ABC-based techniques can prove non-trivial

to derive. In particular, developing ABCs for *nonlinear* media can be especially challenging, requiring the derivation of equivalent or approximate one-way wave equations or radiation conditions within these complex materials, which must then be enforced [101]. Moreover, most ABC are ill-suited for use with dispersive media as they depend on the wave velocity near the boundary, a quantity ill-defined in such media in the time-domain [37]. In contrast, PML-based techniques require no such complementary equations, as the absorbing characteristics and lack of reflection are largely determined by the PML's structure itself. For this reason in the following sections the PML will be developed as the technique of choice for the absorption of dispersive nonlinear phenomena within the FETD method. More specifically, it will be shown that a particular implementation of the PML based on complex coordinate stretching is ideal for use with nonlinear dispersive media, requiring very little additional overhead or complications.

4.1 PML Overview

As mentioned, the Perfectly Matched Layer is a region of artificial attenuation which is perfectly matched to the incident medium (meaning there is no perceived discontinuity in wave impedance on the part of a propagating wave) and thereby suppresses reflections, mimicking a much larger or even infinite domain. First derived by Berenger [102], PMLs were initially formulated by splitting Maxwell's Equations and introducing artificial electric and magnetic conductivities whose values were carefully chosen so as to be reflectionless to the incident medium. In particular, Berenger showed that for a PML to be reflectionless, these electric and magnetic conductivities needed to satisfy an anisotropic matching condition related to the incident medium's material parameters, which in 2D is given by:

$$\frac{\sigma_x^E}{\epsilon} = \frac{\sigma_x^H}{\mu} \qquad \frac{\sigma_y^E}{\epsilon} = \frac{\sigma_y^H}{\mu} \tag{4.1}$$

where σ_i^E and σ_i^H are the artificial electric and magnetic conductivities, and ϵ and μ are the permittivity and permeability, respectively. While simple and straightforward in the case of linear media, the matching condition in equation (4.1) is substantially complicated by the introduction of electrically complex media, in which the permittivity can be a function of time and field strength.

Luckily, further investigation of the notion of a PML later revealed additional matching criteria and implementations that can also yield perfect transmission, but which do not explicitly depend upon ϵ and μ. For instance, Sacks [103] and Gedney [104] showed that Berenger's original PML could equally be interpreted as a uniaxial anisotropic dispersive medium, rather than split artificial conductivities. Thus, within the PML region, one could simply substitute the material parameters for their uniaxial frequency-dependent equivalents in the standard un-split Maxwell's Equations. While this considerably simplifies and unifies the implementation, it does however introduce additional complications in the need to model both dispersion and anisotropy within the PML. Moreover, such an implementation can run into additional issues when truncating dispersive, bianisotropic, lossy, or nonlinear media, as care must be taken in how the PML dispersion and anisotropy are combined with the material parameters present in the adjoining interior domain [105, 106].

However, as it turns out, the reinterpretation of the PML as an anisotropic medium would not be the only reformulation of the method. In fact, additional research into PMLs was able to reveal yet another interpretation of their absorptive and reflectionless properties, this time as a result of a complex coordinate stretching of space within the PML region [107, 108, 109]. As will be shown in the next section, such a coordinate stretching approach amounts to a change of variables within the PML, which essentially transforms propagating waves into decaying ones. Not only that, but coordinate stretching can also be made to only affect the spatial part of the underlying wave equations, rather than the temporal part. In consequence, this makes the coordinate stretching implementation of the PML an attractive candidate for nonlinear dispersive media, as ϵ is only found within the temporal derivatives.

Regardless of the specific interpretation or implementation, PMLs for *linear* media have become immensely popular, both within the FDTD *and* FETD methods, and have resulted in a wealth of research, optimizations, extensions to additional coordinate systems [110], and related techniques. However, their adaptation to complex nonlinear media has been less widespread. Nevertheless, to date several variants of the PML have in fact been successfully applied to nonlinear dispersive materials within the FDTD formulation. These methods, sometimes known as Material Independent Perfectly Matched Layers (MIPML), have been derived using many of the techniques described earlier, including Berenger's split conductivity approach [111, 112], the anisotropic approach [113], and the coordinate stretching technique [114, 115]. Alas, despite these achievements, as it currently stands no such PML formulation has ever been adapted to the FETD method for electrically nonlinear and dispersive media, an unsurprising fact given the general deficit of nonlinear dispersive FETD solvers.

The PML derived in the following sections is thus the first to truncate domains containing electrically nonlinear media within the FETD framework. As such, the resulting technique will have the potential to significantly improve the efficiency and modeling capabilities of the algorithms derived in Chapter 2.

4.2 Coordinate Stretching: Linear Media

For any linear medium, a wave solution propagating in the positive x direction can in general be expressed as a sum of plane waves of the form:

$$E(x,t) \approx e^{j(\omega t - k_x x)} \tag{4.2}$$

where k_x is the wavenumber in the x direction, ω is the angular frequency, and j is the imaginary unit. Evaluating the expression in equation (4.2) along the real x-axis naturally results in an oscillatory solution. However, since the solution in (4.2) is an analytic function,

it is equally valid to consider evaluating it for complex values of x, such as $x = x_R - jx_I$ [116]. As a result of this *analytic continuation* or *stretching*, within the regions where x has an imaginary part the solution changes to the following form:

$$\tilde{E}(x,t) \approx e^{j(\omega t - k_x[x_R - jx_I])} = e^{-k_x x_I} e^{j(\omega t - k_x x_R)}. \tag{4.3}$$

The exponential term related to x_I is now clearly resulting in a decrease in amplitude or attenuation during propagation. In other words, the wave is being *absorbed* within the analytically continued region of x. More importantly, however, is the fact that this behaviour has been obtained without changing the solution in equation (4.2). Thus, in regions where x is strictly real-valued, the solution is *unchanged*. The result is that not only does the stretched region now absorb incident waves, it does so *without* changing the original solution in the unstretched regions, i.e. without reflection. Hence to implement a PML one must simply "stretch" the spatial axes into the complex plane within the desired region of absorption. Moreover, while the above examples were concerned with propagation in the x direction, similar transformations could of course be applied to the y and z coordinates to yield absorption in those directions as well.

The simplest way to accomplish this is to perform a change of variables for Maxwell's Equations within the PML region:

$$\tilde{\xi} = \xi - jf(\xi) \tag{4.4}$$

where in Cartesian coordinates $\xi \in \{x, y, z\}$, $\tilde{\xi}$ is a stretched variable, and $f(\xi)$ is in principle an arbitrary positive function of ξ. As a consequence, any spatial derivatives which occur within the original PDEs must be replaced with their stretched equivalents within the PML

region, derived via the chain rule:

$$\frac{\partial}{\partial \xi} = \frac{\partial \tilde{\xi}}{\partial \xi}\frac{\partial}{\partial \tilde{\xi}} \tag{4.5}$$

$$= \left(1 - j\frac{df}{d\xi}\right)\frac{\partial}{\partial \tilde{\xi}} \tag{4.6}$$

$$= \left(1 + \frac{1}{j}\frac{df}{d\xi}\right)\frac{\partial}{\partial \tilde{\xi}}. \tag{4.7}$$

While many choices are possible for how exactly to stretch the spatial axes, one of the most common is to select:

$$\frac{df}{d\xi} = \frac{\sigma_\xi(\xi)}{\omega} \tag{4.8}$$

where $\sigma_\xi(\xi)$ is again in principle an arbitrary positive function of ξ. With this choice, the substitution to be made in the original differential equations within the PML region becomes

$$\frac{\partial}{\partial \xi} \mapsto \frac{\partial}{\partial \tilde{\xi}} = \frac{1}{1 + \frac{\sigma_\xi}{j\omega}}\frac{\partial}{\partial \xi} \tag{4.9}$$

which may be simplified somewhat by defining $s_\xi = 1 + \sigma_\xi / j\omega$:

$$\frac{\partial}{\partial \xi} \mapsto \frac{1}{s_\xi}\frac{\partial}{\partial \xi}. \tag{4.10}$$

As mentioned, in theory σ_ξ can be any positive function, including a simple scalar constant, and still result in perfect transmission. However, once discretized any abrupt transition will likely result in numerical reflections from the PML as it is unlikely to be exactly perfectly matched in the discrete problem space. In consequence, σ_ξ is more commonly chosen to be a slowly increasing function, providing a gradual transition into the stretched coordinate space and limiting numerical reflections [102, 104].

Furthermore, initially it may seem that the inclusion of the angular frequency ω in equation (4.8) needlessly introduces dispersion into the formulation. However, a closer inspection

of equation (4.3) shows that for a constant imaginary part, the resulting attenuation is dependent on k_x, meaning higher frequencies (shorter wavelengths) will be more strongly attenuated than lower frequencies (longer wavelengths). In contrast, in linear non-dispersive media, the frequency-dependence of the stretching in (4.8) results in an attenuation dependent on k/ω which equates to the phase velocity and is thus a *constant*. The dispersive nature of the coordinate stretching within the PML thereby guarantees that all frequency components will be equally absorbed.

4.3 Coordinate Stretching: Nonlinear Media

One of the main difficulties in moving from a linear domain to a nonlinear one is in the loss of linear superposition. As a result, any solution to the nonlinear wave equation cannot necessarily be considered as being the sum of individual plane wave solutions as in equation (4.2). In consequence, it is not immediately obvious whether the coordinate stretching technique detailed in the last section will immediately carry over to the nonlinear setting for the implementation of a PML.

However, luckily the solutions of the nonlinear Maxwell's Equations are fundamentally still propagatory in nature. Indeed, since the nonlinear parameter $\chi^{(3)}$ is generally very small as compared to its linear counterpart $\chi^{(1)}$, the presence of nonlinearity can be thought of as only a small perturbation to the original solution which would have resulted in linear media. For instance, using a small first order nonlinear perturbation to the wave equation results in the following approximative solution, common in guiding structures such as optical fibres [117]:

$$E(\vec{x},t) \approx \mathrm{Re}\left(F(x,y)A(z,t)e^{j(\omega t-\beta z)}\right) \tag{4.11}$$

where $\mathrm{Re}(\cdot)$ denotes the real part, $F(x,y)$ is the transverse shape of the guided mode, and $A(z,t)$ is the amplitude or envelope function. Accordingly, despite the nonlinearity, solutions to the nonlinear Maxwell's Equations should still largely be absorbed within a region where

the spatial coordinates have been stretched, due to the exponential term in (4.11).

While the solutions to the nonlinear Maxwell's Equations will be mostly absorbed within the coordinate stretched PML region, it is important to note that such absorption may not be exactly as uniform as in the linear case. This is a result of the fact that the ratio β/ω may not be exactly constant over the true nonlinear solution, coupled with the unknown behaviour of the $A(z,t)$ term. However, these issues may be potentially mitigated by tuning the parameters s_ξ and $\sigma(\xi)$, such as was done for the absorption of evanescent waves in linear media [106, 116].

4.4 PML for Nonlinear Dispersive VWE FETD

In this section, the coordinate stretching approach described in the previous sections will be applied to the nonlinear dispersive VWE-based FETD algorithm. While the focus here will be on the VWE formulation, it is worth noting that most of the derivation to follow can easily be carried over to the mixed formulation, given the many similarities between the two methods discussed in Chapter 2.

As a starting point, Faraday's and Ampère's Laws can be expressed in time-harmonic form within the PML region as:

$$\nabla_B \times \vec{E} = -j\omega\vec{B} \tag{4.12}$$

$$\nabla_D \times \frac{1}{\mu}\vec{B} = j\omega\vec{D} + \vec{J} \tag{4.13}$$

where the curl operators $\nabla_B\times$ and $\nabla_D\times$ have had their spatial derivatives modified according to the stretching in equation (4.10), to yield absorption in the desired direction(s). Note that the subscripts B and D in equations (4.12) and (4.13) allow for different stretching factors to be applied to each field, if desired. Moreover, it is worth reiterating that outside of the PML region the equations and formulation remain entirely unchanged from that presented in Chapter 2, with the standard curl operators in place.

Combining Faraday's and Ampère's Laws in the usual fashion results in the following vector wave equation within the PML region:

$$\nabla_D \times \frac{1}{\mu} \nabla_B \times \vec{E} + (j\omega)^2 \vec{D} = -j\omega \vec{J} \tag{4.14}$$

where it must again be emphasized that the curl operators are implicitly functions of frequency. Applying a Galerkin procedure, the dot product of the wave equation in (4.14) is taken with the 1-form basis functions, and the result integrated over each element:

$$\int_{\Omega^e} \left(\nabla_D \times \frac{1}{\mu} \nabla_B \times \vec{E}\right) \cdot \vec{W}_j^{(1)(e)} d\Omega + (j\omega)^2 \int_{\Omega^e} \vec{D} \cdot \vec{W}_j^{(1)(e)} d\Omega + \int_{\Omega^e} j\omega \vec{J} \cdot \vec{W}_j^{(1)(e)} d\Omega = 0. \tag{4.15}$$

The second and third terms in the above do not include spatial derivatives of any kind, and are thus completely unchanged from the non-PML region, resulting in the same $[K]$ matrix and $\{f\}$ vector defined previously:

$$\int_{\Omega^e} (\nabla_D \times \frac{1}{\mu} \nabla_B \times \vec{E}) \cdot \vec{W}_j^{(1)(e)} d\Omega + (j\omega)^2 \Big([K]\{e\} \Big) + \{f\} = 0. \tag{4.16}$$

As for the first term, a straightforward albeit it tedious calculation shows that the vector identity of equation (2.12) remains valid for the stretched curl and divergence operators in the frequency domain:

$$\nabla_D \cdot (\vec{A} \times \vec{B}) = (\nabla_D \times \vec{A}) \cdot \vec{B} - (\nabla_D \times \vec{B}) \cdot \vec{A} \tag{4.17}$$

such that the integral of equation (4.16) can be re-written as:

$$\int_{\Omega^e} \frac{1}{\mu} \big(\nabla_D \times \vec{W}_j^{(1)(e)}\big) \cdot (\nabla_B \times \vec{E})\, d\Omega - \int_{\Omega^e} \nabla_D \cdot \left(\frac{1}{\mu} \vec{W}_j^{(1)(e)} \times \nabla_B \times \vec{E} \right) d\Omega. \tag{4.18}$$

Here, the normal course of action would be to apply the divergence theorem to the second term in (4.18). However, since the spatial coordinates of the divergence operator have been

stretched, a slightly modified form of the divergence theorem must be used (see Appendix B for a derivation):

$$\int_{\Omega^e} \nabla_D \cdot \vec{F}\, d\Omega = \frac{1}{s_x^D}\int_{\partial\Omega^e} F_x \hat{a}_x \cdot d\vec{S} + \frac{1}{s_y^D}\int_{\partial\Omega^e} F_y \hat{a}_y \cdot d\vec{S} + \frac{1}{s_z^D}\int_{\partial\Omega^e} F_z \hat{a}_z \cdot d\vec{S}. \tag{4.19}$$

This version of the divergence theorem is only valid if the PML conductivity σ_ξ is assumed constant in space, meaning the original function $\sigma(\xi)$ is either itself constant or approximated in a piecewise-constant fashion over the length of the PML. While initially this may seem restrictive, a piecewise-constant permittivity can still adequately suppress numerical reflections if sufficient subdivisions are used. More importantly, however, as will be shown later this simplification also permits the elemental matrix expressions to be obtained in closed-form.

Applying this stretched version of the Divergence Theorem to equation (4.18) then results in:

$$\begin{aligned}\int_{\Omega^e} \frac{1}{\mu}\big(\nabla_D \times \vec{W}_j^{(1)(e)}\big) \cdot \big(\nabla_B \times \vec{E}\big)\, d\Omega - \frac{1}{s_x^D}\int_{\partial\Omega^e} \left(\frac{1}{\mu}\vec{W}_j^{(1)(e)} \times \nabla_B \times \vec{E}\right)_x \hat{a}_x \cdot d\vec{S} \\ - \frac{1}{s_y^D}\int_{\partial\Omega^e} \left(\frac{1}{\mu}\vec{W}_j^{(1)(e)} \times \nabla_B \times \vec{E}\right)_y \hat{a}_y \cdot d\vec{S} - \frac{1}{s_z^D}\int_{\partial\Omega^e} \left(\frac{1}{\mu}\vec{W}_j^{(1)(e)} \times \nabla_B \times \vec{E}\right)_z \hat{a}_z \cdot d\vec{S}.\end{aligned} \tag{4.20}$$

As mentioned earlier, however, a common choice of boundary condition for terminating the domain on the far side of the PML layer is a simple Dirichlet or PEC one, in which the tangential field value is simply set to zero. Under this assumption, all of the surface integral terms in equation (4.20) can be dropped. As for the remaining term, the interpolated basis function expansion of the electric field can be substituted as before, resulting in:

$$\sum_{i=1}^{l_e} \int_{\Omega^e} \big(\nabla_D \times \vec{W}_j^{(1)(e)}\big) \cdot \big(\nabla_B \times \vec{W}_i^{(1)(e)}\big)\{e^{(e)}\}_i\, d\Omega. \tag{4.21}$$

At this point, no further simplification of equation (4.21) is possible without knowing the

specific implementation details concerning the basis functions in question.

For instance, if using first-order 1-form basis functions in two spatial dimensions, it can be shown in the frequency-domain (see Appendix C) that

$$\nabla_D \times \vec{W}_j^{(1)(e)} = \frac{l_j^{(e)}}{2\Delta^{(e)}} \left(\frac{1}{s_x^D} + \frac{1}{s_y^D} \right) \tag{4.22}$$

$$\nabla_B \times \vec{W}_i^{(1)(e)} = \frac{l_i^{(e)}}{2\Delta^{(e)}} \left(\frac{1}{s_x^B} + \frac{1}{s_y^B} \right) \tag{4.23}$$

where $l_i^{(e)}$ and $l_j^{(e)}$ are the lengths of the edges associated with basis functions i and j, respectively, and $\Delta^{(e)}$ is the area of the triangular element. Combining this with equation (4.21) then results in:

$$\sum_{i=1}^{l_e} \left(\frac{1}{s_x^D} + \frac{1}{s_y^D} \right) \left(\frac{1}{s_x^B} + \frac{1}{s_y^B} \right) \left(\frac{l_i^{(e)} l_j^{(e)}}{4\Delta^{(e)}} \right) \{e^{(e)}\}_i \triangleq \left(\frac{1}{s_x^D} + \frac{1}{s_y^D} \right) \left(\frac{1}{s_x^B} + \frac{1}{s_y^B} \right) [\hat{S}^{(e)}] \{e^{(e)}\}. \tag{4.24}$$

Note that the $[\hat{S}^{(e)}]$ matrix defined above differs from the previously seen $[S^{(e)}]$ matrix only by a factor of four and in fact reduces to the same matrix if $\sigma = 0$ ($s_\xi = 1$), as expected. Moreover, while the expression in equation (4.24) is valid only for first-order two-dimensional basis functions, the generalization to three dimensions, while naturally more involved, maintains a similar product structure between a function of frequency, a matrix, and a vector (See Appendix C). Combining these results and returning to the time-domain at last yields the semi-discrete vector wave equation within the PML region, for first-order two-dimensional basis functions:

$$\frac{d^2}{dt^2} \Big([K]\{e\} \Big) + p(t) * [\hat{S}]\{e\} + \{f\} = 0 \tag{4.25}$$

in which

$$p(t) = \mathcal{F}^{-1} \left\{ \left(\frac{1}{s_y^D} + \frac{1}{s_x^D} \right) \left(\frac{1}{s_y^B} + \frac{1}{s_x^B} \right) \right\} \tag{4.26}$$

and $\mathcal{F}^{-1}$ denotes the inverse Fourier transform.

It is in equation (4.25) that the main advantage of the coordinate stretching formulation

of the PML becomes clear in this case. Not only is the PML isotropic, but has also only affected the spatial derivatives (corresponding to the $[\hat{S}]$ matrix), while the temporal term containing all of the material complexity is completely untouched. As a result, the PML can easily be incorporated into any existing solver for complex media, without altering its core structure or functionality. This is in contrast to other approaches in which factors are sometimes introduced within the temporal derivative, in conjunction with the original permittivity, and which usually require anisotropy. In addition, the coordinate stretching approach also has the benefit of maximizing the similarity between the formulation in PML and non-PML regions, thus decreasing overhead and algorithmic complexity.

Under these circumstances, applying Newmark-β to equation (4.25) only differs from the procedure used previously by the presence of the convolution with $p(t)$. However, from equations (4.9) and (4.24), $p(t)$ is a rational function of $j\omega$ in the frequency-domain. As a result, the convolution with $p(t)$ is of the exact same type as was studied for dispersive media in Chapter 2, and is easily addressed via the z-transform approach. Indeed, applying the z-transform and Transposed Direct Form II update procedures to the convolution $\{\mathcal{P}\} \triangleq p(t) * [\hat{S}]\{e\}$ results in the following auxiliary update equations:

$$\{\mathcal{Z}_\alpha\}^n = v_\alpha[\hat{S}]\{e\}^n - g_\alpha\{\mathcal{P}\}^n + \{\mathcal{Z}_{\alpha+1}\}^{n-1} \qquad \alpha < p \tag{4.27}$$

$$\{\mathcal{Z}_\alpha\}^n = v_\alpha[\hat{S}]\{e\}^n - g_\alpha\{\mathcal{P}\}^n \qquad \alpha = p \tag{4.28}$$

$$\{\mathcal{P}\}^n = v_0[\hat{S}]\{e\}^n + \{\mathcal{Z}_1\}^{n-1}. \tag{4.29}$$

Following through with the application of Newmark-β to equation (4.25) and substituting (4.29) for the PML convolution at last provides the update equation in the nonlinear

dispersive PML region:

$$\left([K]^{n+1}+\frac{\Delta t^2}{4}v_0[\hat{S}]\right)\{e\}^{n+1}=2\left([K]^{n}-\frac{\Delta t^2}{4}v_0[\hat{S}]\right)\{e\}^{n} - \left([K]^{n-1}+\frac{\Delta t^2}{4}v_0[\hat{S}]\right)\{e\}^{n-1}-\left(\{\mathcal{W}_1\}^n+2\{\mathcal{W}_1\}^{n-1}+\{\mathcal{W}_1\}^{n-2}\right) - \frac{\Delta t^2}{4}\left(\{\mathcal{Z}_1\}^n+2\{\mathcal{Z}_1\}^{n-1}+\{\mathcal{Z}_1\}^{n-2}\right)-\frac{\Delta t^2}{4}\left(\{f\}^{n+1}+2\{f\}^{n}+\{f\}^{n-1}\right). \quad (4.30)$$

Thus, by combining equation (4.30) in the PML region with equation (2.125) in the non-PML region, the domain can be effectively truncated without reflection. The update procedure in the two domains is effectively the same, with the exception of needing to take the extra step of updating the PML auxiliary variables within the PML region(s). In this way, the coordinate stretching approach requires minimal perturbation to the underlying method while providing effective domain truncation. Equation (4.30) is thus the main result of this chapter and represents the first time that the truncating effects of Perfectly Matched Layers are combined with dielectric nonlinearity within the context of the Finite-Element Time-Domain method.

4.5 Numerical Validation

Due to the lack of closed-form solutions, in order to verify the performance of the nonlinear PML devised in this chapter a solution obtained in a PML-truncated domain can instead be compared to one obtained in a domain which has been physically extended well beyond the truncation point. Assuming this extension itself is terminated with a traditional PEC boundary condition, and that the domain is extended a sufficient distance to ensure that reflections from this boundary do not have time to make it back to the PML comparison point, the result should be an accurate assessment of the PML's performance.

Given its physically significant nature, as well as the presence of linear dispersion, instantaneous nonlinearity, and dispersive nonlinearity all in one simulation, the dielectric slab

waveguide example of Subsection 2.5.2 was selected as the PML test problem. As a result, if successful, the PML should result in the truncation of the waveguide, providing reflectionless absorption of a temporal soliton.

The material parameters and transverse dimensions of the waveguide remained unchanged from those used earlier. The base waveguide section common to both the PML and extended domains measured 40 μm long, while the extended domain added an additional 180 μm beyond the shared region, for a total length of 220 μm. The PML, meanwhile, was made to truncate the rightmost portion of the domain, and measured either 5 μm, 10 μm or 20 μm in length beyond the shared region. Moreover, as explained earlier, it is helpful to have the coordinate stretching within the PML region turned on gradually in order to minimize numerical reflections. To that effect, within the PML region the stretching factors were chosen as follows:

$$\sigma_D = \sigma_B = \sigma_{max} \left(\frac{d}{L} \right)^4 \tag{4.31}$$

where d is the distance into the PML region, L is the length of the PML region, and σ_{max} was selected to be 3×10^{14}. However, as was the case in the derivation of the expressions in equations (4.25) and (4.26), it is often necessary to assume that the factors in (4.31) are constant within each element to facilitate the derivation. Thus, for each length the PML region was divided into 20 equal rectangular sub-regions, forming a piecewise constant approximation to the fourth-order polynomial profile in (4.31).

The results of running the same simulation on both the extended and PML-truncated domains are displayed in Figure 4.1, which shows a comparison plot of the y-component of the fields obtained in both cases as measured in the center of the guide approximately 10 grid points from the interface. For these simulations, the average element edge length was set to $h = 80\ nm$ with a 20 μm thick PML. Upon visual inspection the two signals appear to be virtually indistinguishable, with very little variation or reflection demonstrated by the PML.

Alternatively, to better quantify the PML's performance, the amount of reflection can

instead be characterized by defining a reflection coefficient as follows:

$$\Gamma(\omega) = 20\log_{10}\left|\frac{\mathcal{F}(E_{y,ext}) - \mathcal{F}(E_{y,pml})}{\mathcal{F}(E_{y,ext})}\right| \tag{4.32}$$

in which $E_{y,ext}$ is the reference solution obtained on the extended domain, $E_{y,pml}$ is the PML solution, and $\mathcal{F}(\cdot)$ denotes the Fourier transform. The reflection coefficient so defined is plotted in Figure 4.2 as a function of both frequency and PML thickness. For reference, the measured normalized spectrum of the incident pulse is also depicted.

For a 20 μm thick PML the reflection at the pulse's fundamental frequency of 50 THz is -60 dB, representing a reflection of roughly 0.1% of the incident wave or 0.0001% of the incident power. The PML also performs equally well over the rest of the pulse's bandwidth, with an average reflection of roughly -58.5 dB and peak reflection of -43.1 dB.

Interestingly, for all thicknesses the PML seems to suffer significant reflections in the 30 - 35 THz range, which is due in part to the proximity of the resonance frequency of the linear dispersive medium. While this is not an issue for the present simulation due to very little of the pulse's spectral energy being concentrated in this region ($\approx 0.5\%$), it is an important factor to consider when applying the PML to future problems. This breakdown of the PML in the vicinity of material resonance frequencies is not unique to the methods in this thesis, however, and has previously also been reported in FDTD implementations [114].

Finally, while for all of the above results a fixed number of 20 piecewise constant subdivisions were used within the PML layer, in principle this too can be varied and have an important impact on the accuracy of the PML. For instance, using fewer sub-divisions was shown to increase numerical reflections, as the transition between regions is made more abrupt. Conversely, increasing the number of subdivisions is likely to further increase accuracy up until other factors such as mesh size dominate the error term. The increased smoothness of the piecewise-constant approximation resulting from more subdivisions likely not only improves performance by minimizing numerical reflections, but also by better ap-

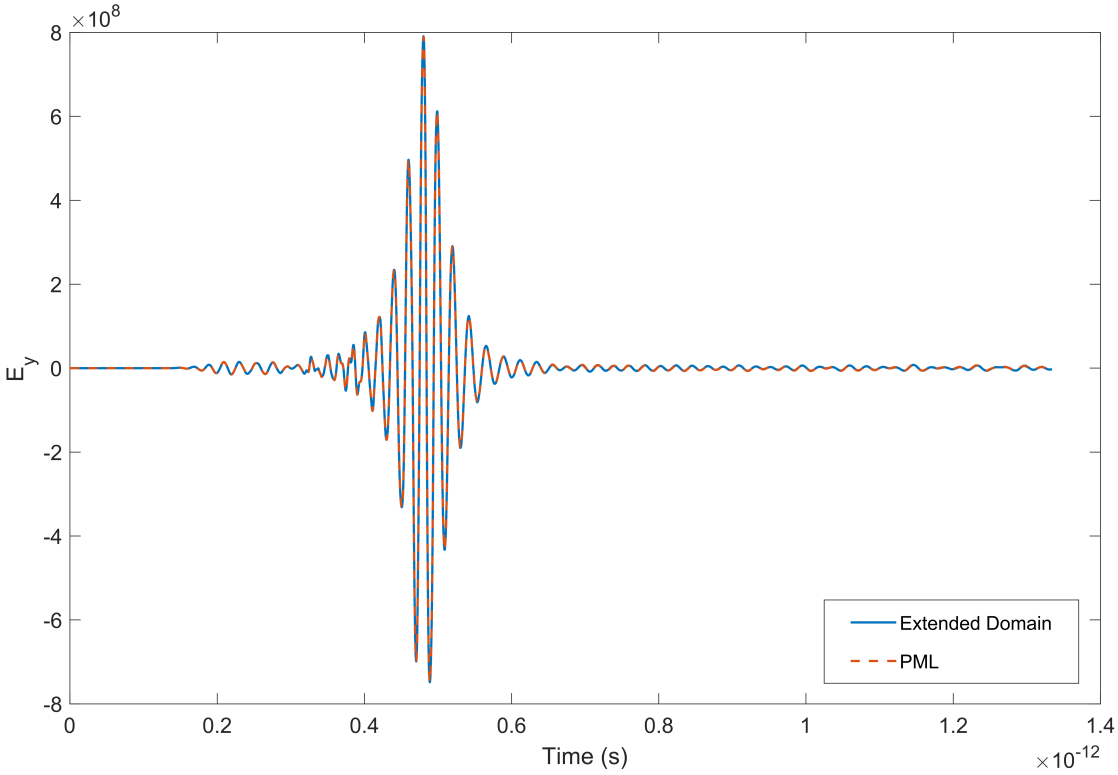

Figure 4.1: Comparison of extended domain and PML electric fields over time for a $20\mu m$ thick PML layer.

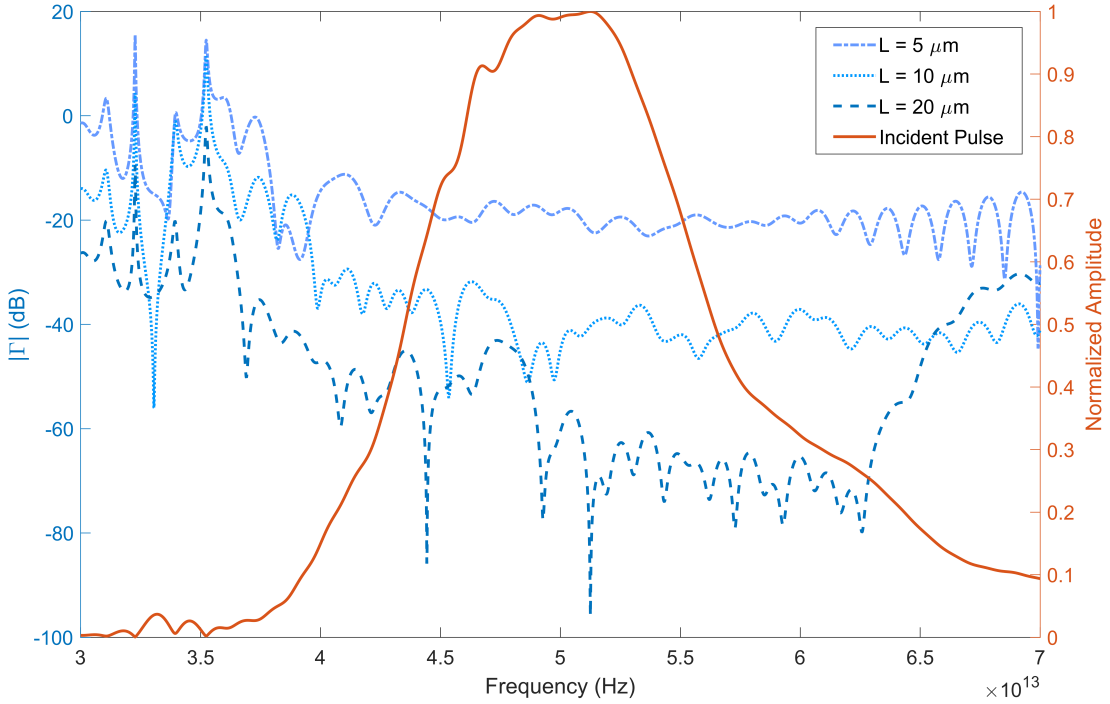

Figure 4.2: Reflection coefficient vs frequency and PML length for the absorption of a temporal soliton.

proximating a gradual adiabatic absorber [118].

4.6 Summary

In this chapter a Perfectly Matched Layer (PML) compatible with the nonlinear dispersive FETD methods presented in Chapter 2 was derived for the truncation of computational domains containing electrically complex media. Of the various formulations of the PML that exist in the literature, the coordinate stretching version was found to be the ideal candidate for incorporation into these methods as it only produces modifications to the spatial parts of the underlying equations, which are unaffected by the complex dielectric behaviour. Moreover, despite the additional complications imposed by nonlinearity it was argued that the solutions to the nonlinear Maxwell's Equations are still propagatory in nature, allowing the coordinate stretched PML to effectively truncate the domain in the nonlinear regime. Lastly, the performance of the nonlinear PML was verified via a numerical study in which a temporal soliton propagating in a dielectric slab waveguide was shown to be successfully absorbed by the truncating layer, with an average reflection of -58.5 dB over the bandwidth of the pulse.

Chapter 5

Parallel Nonlinear FETD

As has been alluded to several times thus far, the added accuracy and flexibility of FETD methods for nonlinear dispersive media come at a computational cost. The need to re-evaluate, reassemble, and solve all nonlinear matrices multiple times per time step can, in many cases, be such a burden as to not warrant the added accuracy and capability versus a simpler approximative method unless absolutely necessary.

Due to this significant barrier to entry, this chapter will be dedicated to improving the efficiency and execution times of the algorithms developed thus far. In particular, this will be accomplished by exploring the use of parallelism to accelerate each step of these methods, using novel combinations of existing and original techniques. Moreover, due to their prevalence, affordability, and immense computational capacity, the implementation of these parallel algorithms will be discussed with the goal of execution on NVIDIA Graphics Processing Units (GPUs). It is imperative to note, however, that while the implementation details presented in the following sections are GPU-centric, in principle the parallel algorithms themselves are architecture independent. Thus, future studies could also investigate and compare the performance of other parallel systems, such as multi-core Central Processing Units (CPUs), as well as supercomputing clusters.

5.1 Graphics Processing Units (GPUs)

Graphics Processing Units (GPUs) have evolved tremendously over the course of their history. In the early days of computing, dedicated video hardware was required to handle little more than monochromatic text output on Cathode Ray Tubes (CRTs). However, as computer systems naturally evolved, so too did their ability to visually interact with the user. Indeed, dedicated graphics hardware rapidly transformed from simple non-programmable pipelines for the rendering of 2D wireframe drawings to full blown parallel architectures capable of rendering dozens of high definition 3D images in near real-time [119]. Today, GPUs continue to evolve and strive for ever increasing computational power, propelled by strong commercial and consumer demand for high definition video playback, video games, and research tools.

While Central Processing Units (CPUs) have also evolved rapidly over this time, they did so in a fundamentally different way to their GPU counterparts. Whereas CPUs are required to be general purpose, GPUs are hyper-specialized for graphics processing and have therefore largely been designed to exploit the features and structure of graphics computations. In particular, many of the operations associated with computer graphics are tedious and repetitive, requiring thousands of similar arithmetic operations be performed for each component in a scene. For example, one ubiquitous operation in graphics processing is texture mapping, in which a 2D image is transformed or overlaid on top of a polyhedral 3D surface. With the large number of polygons making up a 3D model, texture mapping consists of an immense number of independent floating-point coordinate transformations per video frame [119].

Seeing that texture mapping is but one example of a large set of independent computations associated with computer graphics, the advantages of parallel processing were quickly identified. Rather than having one large sophisticated processor churn through each computation in series, a much larger number of simpler workers could perform the same computations concurrently, translating into a significant increase in speed, throughput, and framerate, as well as smoother performance. It is no surprise, then, that GPUs today are

composed of thousands of specialized sub-processors, equipped with a multitude of intricate parallel pipelines, each capable of performing the operations required for 3D rendering.

This focus on parallelism has naturally caused GPU architecture and design philosophy to diverge from their CPU counterparts. As mentioned, CPUs tend to be optimized for general serial execution and branching, and as a result are equipped with sophisticated caching and control structures. GPUs, in contrast, are optimized for Floating Point Operations per Second (FLOPS) and memory bandwidth, at the expense of far less caching and control [120].

With this emphasis, it is unsurprising that modern GPUs have eclipsed their CPU counterparts in terms of FLOPS and memory bandwidth. However, despite the escalating gap in these metrics between the two, only in the past decade have GPUs begun to be widely used for general purpose computations in addition to graphics processing. This lag in general purpose GPU computing was partly due to the fact that not all algorithms are immediately suited to a parallel execution model, requiring the development of parallel alternatives. However, a more severe complicating factor was that prior to this time, programming a GPU was generally a difficult and unwieldy affair [119]. It was not until the advent of dedicated multipurpose GPU programming languages (such as NVIDIA CUDA [121] and OpenCL [122]) that GPUs were able to be widely integrated into scientific computing. Since then, they have grown to be a pervasive and dominant force in scientific computing, from fields as diverse as molecular biology and computational chemistry, to linear algebra, computational finance, and machine learning [123, 124].

Due to the massively parallel nature of the GPU architecture, as well as their relative affordability and ubiquity, the derivation and implementation of the parallel algorithms to follow will be presented with GPUs in mind. Specifically, the GPU architecture will be assumed to be that of NVIDIA's, and the resulting implementations programmed in their proprietary Compute Unified Device Architecture (CUDA) GPU language.

5.2 Derivation

In previous chapters, much of the overhead associated with nonlinear dispersive simulations was postulated to be due to the evaluation and assembly of elemental and global matrices. Indeed, when combined with the inversion of the Jacobian matrix system, these three operations together likely constitute the vast majority of computation time for nonlinear dispersive FETD. As a result, in this section each of these operations will be analyzed to develop a parallel alternative, in the hopes of improving the algorithm's overall performance.

It should equally be noted that the introduction of both linear and nonlinear dispersion also results in increased overhead, as time must be taken to update all of the auxiliary variables at each time step. However, the overhead imposed by these auxiliary variable updates, as well as possible parallelization strategies, has already been discussed in the literature [125], and in any case likely pales in comparison to the three components identified above. As a result, the present treatment will focus primarily on the mentioned hypothesized bottlenecks, which are due to the nonlinearity.

Lastly, because of the similarity between the mixed and VWE formulations noted earlier, these three operations are naturally shared between the two methods. In consequence, the following discussion will analyze the general nature of the operations to be parallelized without focusing on a particular formulation, knowing that it can easily be applied to either method. In fact, since the operations to be discussed are common across most if not all nonlinear Finite-Element implementations, the algorithms presented in this section may prove beneficial not only to the simulation of electrically nonlinear media, but any type of nonlinear problem with a similar formulation, such as those for magnetic machines.

5.2.1 Parallelization Strategy

With the prevalence of nonlinearity in other engineering fields, the issue of accelerating the solution process for nonlinear systems of equations is not new to electrically complex media.

In fact, many alternatives or modifications to the standard Newton-Raphson technique have been developed over the years to attempt to reduce the computational burden of nonlinear root finding. In particular, the Jacobian-Free Newton-Krylov (JFNK) method has become quite popular due to its ability to avoid calculating the Jacobian matrix explicitly [126]. It does this by leveraging the Krylov subspace (in which matrix inversion is instead replaced with a series of matrix-vector products) and approximating the Jacobian via a Taylor expansion in the system vector $\{F\}$. Despite the success and popularity of this method, in this chapter a different approach will be adopted. Rather than approximating the Jacobian as in JFNK, here the full non-approximative Newton-Raphson approach will be employed, and the overhead associated with forming the exact full Jacobian directly parallelized. The resulting method will hopefully thus exhibit the full accuracy and convergence of the Newton-Raphson method, but with a significant boost in speed and performance. Nevertheless, the JFNK method may prove a useful avenue for future study.

5.2.2 Elemental Matrix Evaluation

The first bottleneck to be addressed is that of elemental matrix evaluation. As was discussed in Chapter 2, to evaluate its local $[K^{(e)}]$ and $[J^{(e)}]$ matrices each element must perform numerical integration using Gaussian Quadrature. However, within the framework of the Finite-Element method, each of these elemental matrices is completely independent of the others. In essence, information is only "shared" between elements according to their connectivity when they are assembled into their global counterparts. The numerical evaluation of the elemental matrices thus requires the same computations be executed on different data sets completely independently. This thereby constitutes a so-called *embarrassingly parallel* problem.

This type of problem is ideally suited for implementation on a wide range of parallel systems. Indeed, this is particularly true of the GPU architecture, which is based on a Single Instruction Multiple Thread (SIMT) execution model. Under this model, NVIDIA GPUs

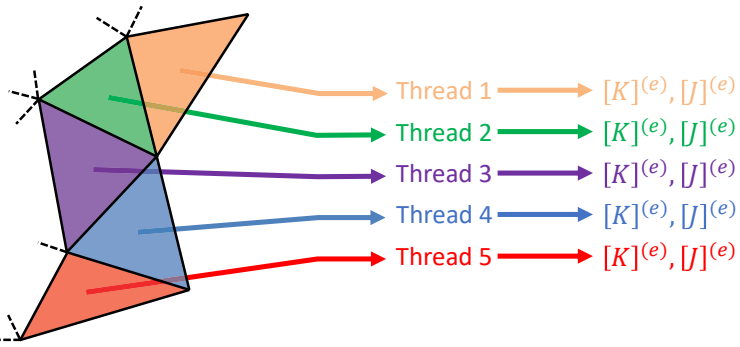

Figure 5.1: Straightforward parallelization of elemental matrix evaluation. Used with permission from [127], ©2020 IEEE.

group single execution instances (called *threads*) into structures called *blocks*. These blocks are then assigned to the GPU's execution hardware, known as *Streaming Multiprocessors* (SMs), in which batches of threads are executed in lock-step with each other [120]. The SIMT model and GPU hardware are thus most efficient when executing the exact same instructions over and over again, but on different memory locations.

Due to the embarrassingly parallel nature of the numerical matrix evaluation and the GPU's SIMT architecture, the required GPU algorithm is relatively straightforward to develop and implement. Indeed, if each nonlinear element in the mesh is assigned to a thread, then the Gaussian Quadrature calculations can proceed in lock step with each other, as each thread accesses the geometry and field data for its own element in memory. This straightforward parallelization scheme is depicted visually in Figure 5.1.

Despite the simplicity of this approach, its potential impact should not be underestimated. If, as hypothesized, the matrix evaluations do indeed constitute a significant fraction of computation time (as will soon be verified), substantial performance increase should result from this step alone. Nonetheless, the global matrix assembly and solving should not be neglected in terms of added overhead since they are repeated so often in the nonlinear regime. Moreover, as will be discussed in Section 5.3, additional overall improvements come from having as much of the total algorithm execute on the GPU as possible.

5.2.3 Matrix Assembly and Solving

The parallelization of global matrix assembly from local elemental matrices has been widely studied within Finite-Elements, particularly for frequency domain or quasi-static analyses where it generally represents a larger fraction of overall computation time [128, 129]. As mentioned earlier, the independent local elements only share information with each other during global matrix assembly. Since this requires connected elements to add their local contributions together to form the global matrix entries, a naive parallelization strategy would inevitably lead to race conditions, in which two elements attempt to write to the same memory location at the same time, yielding undefined behaviour. To mitigate this, additional processing layers are often required, such as coloring algorithms [130]. Additionally, parallel algorithms for the solution of sparse matrix systems constitutes a massive area of study in its own right, with direct methods (such as parallel Cholesky factorization [131]) and Krylov methods (such as Preconditioned Conjugate Gradient (PCG) [132] and Generalized Minimum Residual (GMRES)[133]) having attained widespread use.

However, rather than adopting some combination of these existing algorithms, here a different approach will be employed. Instead of parallelizing the assembly and solution procedures independently, the recently proposed Finite-Element Gaussian Belief Propagation (FGaBP) method will be used instead. As will be discussed in the next subsection, one of the true strengths of the FGaBP algorithm is that it is able to solve the matrix equations in parallel *without* ever explicitly assembling the global matrix. The FGaBP algorithm thereby addresses both of these issues at once, skipping the assembly step entirely while still providing a parallel solution to the global matrix system.

5.2.4 Gaussian Belief Propagation

Originally derived for frequency-domain problems, the Finite-Element Gaussian Belief Propagation (FGaBP) method is a parallel technique for the solution of linear systems arising from the Finite-Element method [134, 135, 136]. Interestingly, the main advantage of the FGaBP

method is not only that it solves matrix systems resulting from Finite-Element discretizations in parallel, but that it does so without ever explicitly forming the global matrix, resulting in a performance boost as compared to traditional parallel Finite-Element algorithms. Furthermore, the FGaBP algorithm has also recently been adapted to linear time-domain problems via the FETD method, with similar performance increase [137]. Within the present context, the ability of the FGaBP method to solve the Jacobian matrix system directly from the local matrices, without ever needing to assemble the global Jacobian matrix, makes the FGaBP method an ideal candidate for use with the algorithms derived in Chapter 2.

In essence, the FGaBP method works by reformulating the linear system into a maximization problem. Given the general global linear Jacobian system associated with a Newton-Raphson iteration:

$$[J]_{(k)}\{\Delta e\} = -\{F\}_{(k)} \tag{5.1}$$

where $\{\Delta e\} = \{e\}_{(k+1)} - \{e\}_{(k)}$, the equivalent maximization problem can be stated as:

$$\arg\max_{\{\Delta e\}} (P) = \arg\max_{\{\Delta e\}} \left(\exp\left(-\frac{1}{2}\{\Delta e\}^T [J]_{(k)}\{\Delta e\} - \{F\}_{(k)}^T\{\Delta e\} \right) \right). \tag{5.2}$$

Indeed, taking the derivative of P in equation (5.2) with respect to $\{\Delta e\}$ and setting it equal to zero yields:

$$\frac{\partial P}{\partial\{\Delta e\}} = \exp\left(-\frac{1}{2}\{\Delta e\}^T [J]_{(k)}\{\Delta e\} - \{F\}_{(k)}^T\{\Delta e\} \right) \left([J]_{(k)}\{\Delta e\} + \{F\}_{(k)} \right) = 0 \tag{5.3}$$

whose solution is exactly given by equation (5.1).

By itself, the reformulation of equation (5.2) is not particular useful at first glance. However, due to its exponential nature, it actually possesses two key properties which make the FGaBP algorithm possible. Firstly, recalling that within the FETD formulation the

global matrix $[J]$ is formed by the combination of elemental matrices $[J^{(e)}]$:

$$[J] = \sum_e [J'^{(e)}] \tag{5.4}$$

permits equation (5.2) to be re-written in terms of expanded local matrices as:

$$\arg\max_{\{\Delta e\}} \left(\exp\left(\sum_e -\frac{1}{2}\{\Delta e^{(e)}\}^T [J'^{(e)}]_{(k)} \{\Delta e^{(e)}\} - \{F'^{(e)}\}_{(k)}^T \{\Delta e^{(e)}\} \right) \right). \tag{5.5}$$

However, given the exponential nature of this expression, it may equally be expressed as the product of local factors:

$$\arg\max_{\{\Delta e\}} \left(\prod_e \exp\left(-\frac{1}{2}\{\Delta e^{(e)}\}^T [J'^{(e)}]_{(k)} \{\Delta e^{(e)}\} - \{F'^{(e)}\}_{(k)}^T \{\Delta e^{(e)}\} \right) \right). \tag{5.6}$$

The second key property of this formulation is that the global expression in equation (5.2), as well as each of the factors in (5.6), can be interpreted as unnormalized Gaussian distributions if the $[J]$ matrix is symmetric positive definite[1]. Under this interpretation, the solution to equation (5.2) is that which maximizes the equivalent Gaussian distribution, which is simply the mean value $\{\mu\}$:

$$\{\Delta e\} = \{\mu\}. \tag{5.7}$$

The ability of the FGaBP algorithm to solve the resulting matrix problem without ever explicitly forming the equivalent global problem is thanks to these two properties. The goal of the FGaBP algorithm is thus to have each elemental factor in equation (5.6) independently attempt to maximize the overall probability. Naturally, because of the interdependence and connectivity of the individual elemental factors, some amount of communication will be required between adjacent elements in order to jointly maximize the global probability. The

[1]Recall from Chapter 3 that the $[J]$ matrix is provably symmetric positive definite for *instantaneous* nonlinearities. For *dispersive* nonlinearities, the property of positive-definiteness is unproven, but supported by numerical studies.

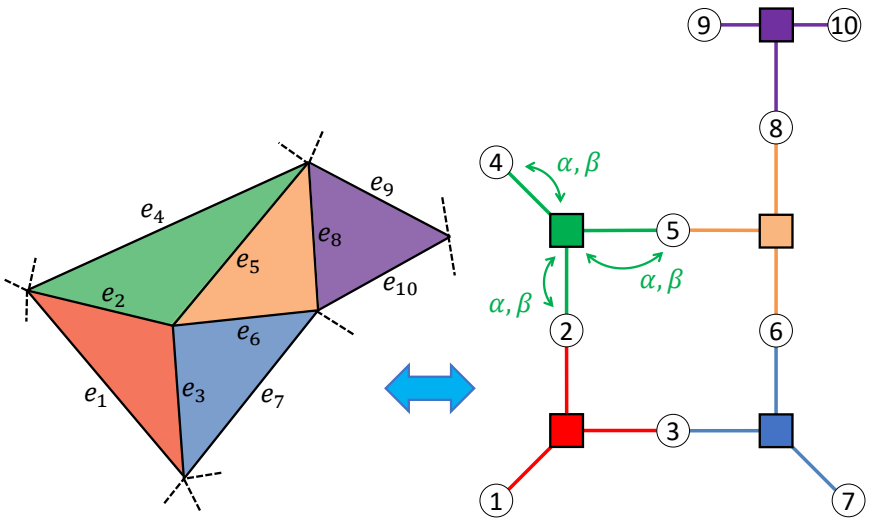

Figure 5.2: Visual depiction of the FGaBP algorithm. Used with permission from [127], ©2020 IEEE.

resulting messages shared between elements or factors essentially amount to conditional or marginal means, i.e. roughly speaking, given the values of neighboring variables, what is the likely value of the current unknown.

With the Gaussian nature of the local factors in equation (5.6), these messages can actually be found in closed-form, and are related to the local elemental matrices $[J^{(e)}]$. Thus, as the algorithm proceeds, adjacent elements will share information back and forth about the likely values of their unknowns, until these values globally converge to the most likely value: the mean $\{\mu\}$ which is the solution to the original global problem. This process is shown graphically in Figure 5.2, where the messages in question (α and β) are mediated between the unknowns (variable nodes) via the elements (factor nodes).

Let the messages from a variable node (i) to a factor node (a) be denoted by α_{ia} and β_{ia}, and those from a factor node (a) to variable node (i) by α_{ai} and β_{ai}. If two intermediate variables A_i and B_i are used to aggregate and relate the individual messages, then the FGaBP algorithm proceeds as follows:

1. For each variable node (i), receive the messages α_{ai} and β_{ai} from each adjoining factor node and use them to update A_i and B_i as well as calculate new messages to send back

to the factor nodes:

$$A_i = \sum_{k \in \mathcal{N}(i)} \alpha_{ki} \qquad B_i = \sum_{k \in \mathcal{N}(i)} \beta_{ki} \tag{5.8}$$

$$\alpha_{ia} = A_i - \alpha_{ai} \qquad \beta_{ia} = B_i - \beta_{ai} \tag{5.9}$$

where $\mathcal{N}(i)$ denotes all factor nodes (elements) in the neighborhood of (connected to) variable node (edge) (i).

2. For each factor node (a), receive the messages α_{ia} and β_{ia}. Then, define a new matrix $[W^{(a)}]$ and vector $\{S^{(a)}\}$ as:

$$[W^{(a)}] = [J^{(a)}] + \text{diag}(\alpha_{1a}, \alpha_{2a}, \cdots) \tag{5.10}$$

$$\{S^{(a)}\}_j = -\{F^{(a)}\}_j + \beta_{ja} \tag{5.11}$$

which will then permit the evaluation of the updated messages α_{ai} and β_{ai} to send back to the variable nodes. More specifically, for each factor node, the updated messages to be sent back to the variable nodes may be found via

$$\alpha_{ai} = [J^{(a)}]_{ii} - \{V^{(a)}\}^T [\tilde{W}^{(a)}]^{-1} \{V^{(a)}\} \tag{5.12}$$

$$\beta_{ai} = \{F^{(a)}\}_i - \{\tilde{S}^{(a)}\}^T [\tilde{W}^{(a)}]^{-1} \{V^{(a)}\} \tag{5.13}$$

where $\{V^{(a)}\}$ is the i^{th} row of the $[W^{(a)}]$ matrix with the i^{th} column removed, $[\tilde{W}^{(a)}]$ is the $[W^{(a)}]$ matrix with the i^{th} row and i^{th} column removed, and $\{\tilde{S}^{(a)}\}$ is the $\{S^{(a)}\}$ vector with the i^{th} row removed.

3. Repeat steps 1 and 2 above over and over, trading messages back and forth between factor and variable nodes, until the variables A_i and B_i converge. That is, until the values of A and B are no longer changing to within some tolerance.

4. Once convergence has been achieved, recover the mean (and therefore the solution) via the following:

$$\{\Delta e\}_i = \{\mu\}_i = \frac{B_i}{A_i}. \tag{5.14}$$

The FGaBP procedure outlined above not only results in the solution to the linear system of equation 5.1, but does so only ever using information from the local matrices and source vectors. The algorithm does, however, require some additional overhead. For instance, storing each of the individual elemental matrices does require more memory than the equivalent combined global matrix. Moreover, some initial overhead is still required in order to avoid conflicts in updating A and B. Even so, the overall resulting algorithm is immensely parallelizable: thousands of messages to and from the elements and unknowns can be computed entirely in parallel, using only local elemental data.

Lastly, a closer look at equations (5.12) and (5.13) shows that each element must still solve a linear system of the form:

$$[\tilde{W}^{(a)}]\{y\} = \{V^{(a)}\}. \tag{5.15}$$

However, in contrast to the original global problem, each of the elemental $[\tilde{W}^{(a)}]$ matrices is much, much smaller than $[J]$. In fact, for an element with p degrees of freedom, the resulting system is of order $p-1$. Hence, for triangular first-order elements, this equates to solving a 2×2 matrix, which is easily inverted directly in closed-form. For more details concerning the theory, derivation, convergence, and implementation of the FGaBP algorithm, the reader is encouraged to consult references [134, 135, 136] and [137].

With this development, the overall procedure for the parallel solution of a nonlinear system via the Newton-Raphson, parallel elemental evaluation, and FGaBP methods is now as follows:

1. Using the current solution estimate $\{e\}_{(k)}$, use the method described in Subsection 5.2.2 to evaluate the new $[K^{(e)}]$ and $[J^{(e)}]$ matrices for each element.

2. Using the elemental matrix data evaluated in the previous step, initiate the FGaBP algorithm to solve for $\{\Delta e\}$.

3. Once FGaBP has converged and the solution has been recovered, update the Newton-Raphson estimate using $\{e\}_{(k+1)} = \{e\}_{(k)} + \{\Delta e\}$.

4. Return to step 1 and perform a new iteration of Newton-Raphson until the estimate $\{e\}_{(k)}$ has converged to the desired tolerance.

5.3 Implementation Details

The combination of parallel elemental matrix evaluation and the FGaBP algorithm presented in the previous sections should, in principle, provide a significant performance boost when executed on massively parallel hardware such as a GPU. However, due to the nature of GPU architecture, a haphazard implementation of a particular algorithm can easily result in unwanted serializations and underwhelming performance. Specifically, within the NVIDIA GPU architecture there are two main points of optimization which can have a significant impact on performance and which will be discussed next.

The first is the interplay and distinction between GPU and CPU memory. In general, each of the SMs on the GPU has access to a large amount of storage known as global memory. While this global memory is physically located on the graphics card, it is not on the same chip as the processor itself. Moreover, it is imperative to note that this GPU global memory and the host CPU memory are fundamentally distinct, meaning that the CPU and GPU do not have access to a shared common memory space. A direct consequence of this is that any time information must be shared between the GPU and CPU it must be explicitly transferred from one to the other. Unfortunately, this memory transfer between host CPU memory and global GPU memory must happen over a bus, and is therefore much slower than simply accessing data already stored on the GPU. If data is constantly going back and forth between the CPU and GPU the resulting transfer overhead can rapidly overshadow

any performance increase afforded by the parallelization [120].

Minimizing memory transfers between the host CPU and the GPU is thus an important point of optimization. When implementing the parallel version of the algorithms detailed in the previous sections it is therefore imperative to ensure that as much of the computation occurs on the GPU as possible. For instance, in addition to the elemental matrix evaluation and the FGaBP algorithm, the evaluation of source terms, updating of the magnetic field (in the case of mixed methods), and updating of any auxiliary variables, should all be executed on the GPU within global memory. The result of this optimization strategy is that, for the algorithm implemented and tested in the next section, memory transfers occur only twice: once at the beginning to transfer mesh and geometric data to the GPU and once at the end to transfer the solution back to the CPU.

The second memory optimization consideration comes in the form of how NVIDIA GPUs access their global memory. Due to the SIMT design philosophy of the GPU, global memory is designed to be accessed in large contiguous chunks, meaning that all of the data required by a group of executing threads can be fetched in a single memory transaction. In contrast, if each executing thread were to access random scattered locations in global memory, this access pattern would result in serialization and necessitate 32 different memory transactions. Clearly, maximum memory bandwidth will thus result when executing threads require access to sequential memory addresses [120].

As a result of this global memory access pattern, the storage format of the algorithm's various data structures can have a significant impact on memory throughput and performance. For example, consider Figure 5.3 in which three threads, each associated with a different element, are each attempting to access their local matrix data from global memory. If these local matrices are stored in row-major format one after the other, the threads will end up requesting non-consecutive memory locations, requiring multiple memory transactions. In contrast, if the local matrices are interwoven, the resulting memory accesses can be made to be contiguous and served in a single transaction. The result is that for best

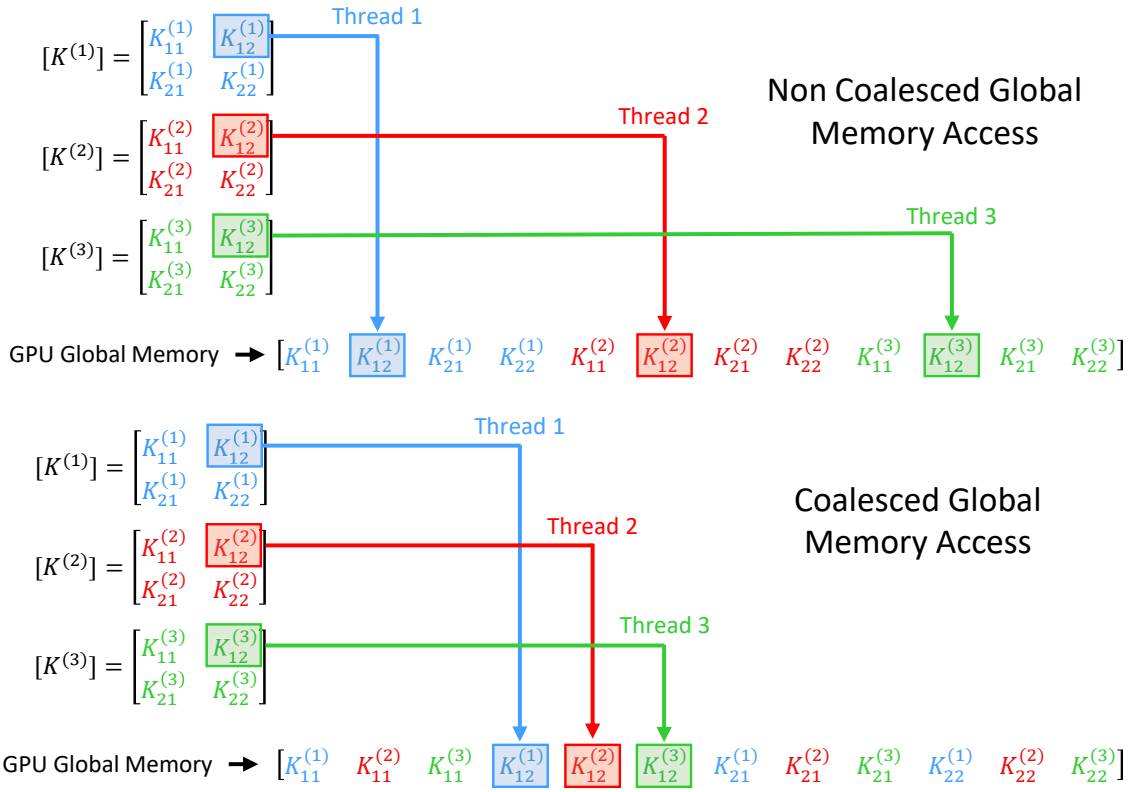

Figure 5.3: Depiction of non-coalesced vs coalesced GPU global memory access.

performance data should be stored in GPU memory in a way consistent with the required access patterns.

While the two memory considerations detailed above are generally where the most performance can be gained or lost, there are also other smaller optimizations possible which may also increase performance. For example, in addition to global memory, the GPU also has much smaller constant and shared memory available [120]. These differ in their access patterns and scope, but if used properly are much faster than global memory. The implementation tested in the next section, for instance, loads values such as the Gaussian Quadrature abscissae and weights, as well as physical parameters, into constant memory as they require little space and are unchanged over the course of the computation.

While having the entire computation execute on the GPU without the need for CPU intervention or memory transfers results in a much faster parallel algorithm, it must be noted that this is counterbalanced by the need to have the entire simulation fit into the GPU

memory. For small computations this is inconsequential, however for very large computations GPU memory is generally less abundant than CPU memory and may become a limiting factor. However, while GPUs offer the greatest amount of hardware parallelism (and were exclusively used for parallelization in this thesis), it is important to reiterate that the algorithms described in this chapter are generally architecture independent. Hence, while the implementation details will vary between different hardware, these algorithms could equally be adapted for execution on multi-core CPUs and supercomputing clusters.

5.4 Numerical Analysis

In this section the GPU parallelization schemes proposed in this chapter will be tested to determine their level of effectiveness in accelerating nonlinear FETD computations. In particular, the next subsection will empirically verify many of the postulates made concerning which aspects of the nonlinear FETD algorithms contribute the most overhead and are the most computationally intensive. The following subsection will then present the results of addressing this overhead via a GPU implementation of the proposed parallel algorithms.

As was mentioned earlier, the parallelization of the auxiliary variable update equations associated with dispersion has already been addressed in the literature [125]. As a result, the selected test problem for the present analyses was the spatial soliton problem of Subsection 2.5.1, and contained only an instantaneous nonlinearity. For all the results that follow, therefore, the exact same dimensions and simulation parameters as in Subsection 2.5.1 were used.

5.4.1 Nonlinear Overhead Analysis

To obtain performance metrics for the base FETD method, a serial version of the mixed algorithm was implemented in C++. Mesh and geometry data was generated and exported by MATLAB R2018A [82], and subsequently read into the test program from binary files,

with the remainder of the simulation being executed locally by the CPU. The solving of the global Jacobian system was done via the Preconditioned Conjugate Gradient (PCG) method, implemented in the open-source linear algebra package Eigen v3.3.5 [138]. The remainder of the algorithm was custom written, including the calculation of local elemental matrices and the assembly of the global system.

Once complete, the serial algorithm was compiled and optimized with the \O2 compiler flag and intrinsic functions enabled. The program was then executed on a single core of a workstation equipped with an Intel 8700K CPU clocked at 3.7 GHz. The machine was equally supplied with 16 GB of DDR4 RAM, operating at 3000 MHz. Chronometry was performed by querying the Windows performance counter [139], with all timing data being averaged over 7 runs. Finally, two versions of the CPU code were actually produced, one using single floating point precision arithmetic and the other using double precision. Despite double precision being far more common due to its increased accuracy, the decision was made to include single precision in the following analyses due to the fact that single precision operations tend to be much faster and may thus be useful for obtaining approximate results for parameter tuning during design before being followed up with a more accurate but costly double precision computation.

Figure 5.4 shows some of the results of profiling the serial CPU implementation. In particular, it shows the proportion of total computation time spent on two key parts of the simulation: elemental matrix evaluation/assembly and the Jacobian matrix solving. Even for the simulations with the least number of electric field degrees of freedom (DoF), it can be seen that the elemental matrix evaluation and assembly represents the vast majority of computation time, ranging from 83% to 86% depending on precision, versus 12% to 15% for matrix solving. As the number of elements and degrees of freedom increases, however, the situation becomes even more extreme. By the time the resolution has increased to almost 120,000 DoF, elemental matrix evaluation and assembly represent a staggering 97% to 98% of computation time, vs only 1% to 2% for matrix solving. These results are paradoxically

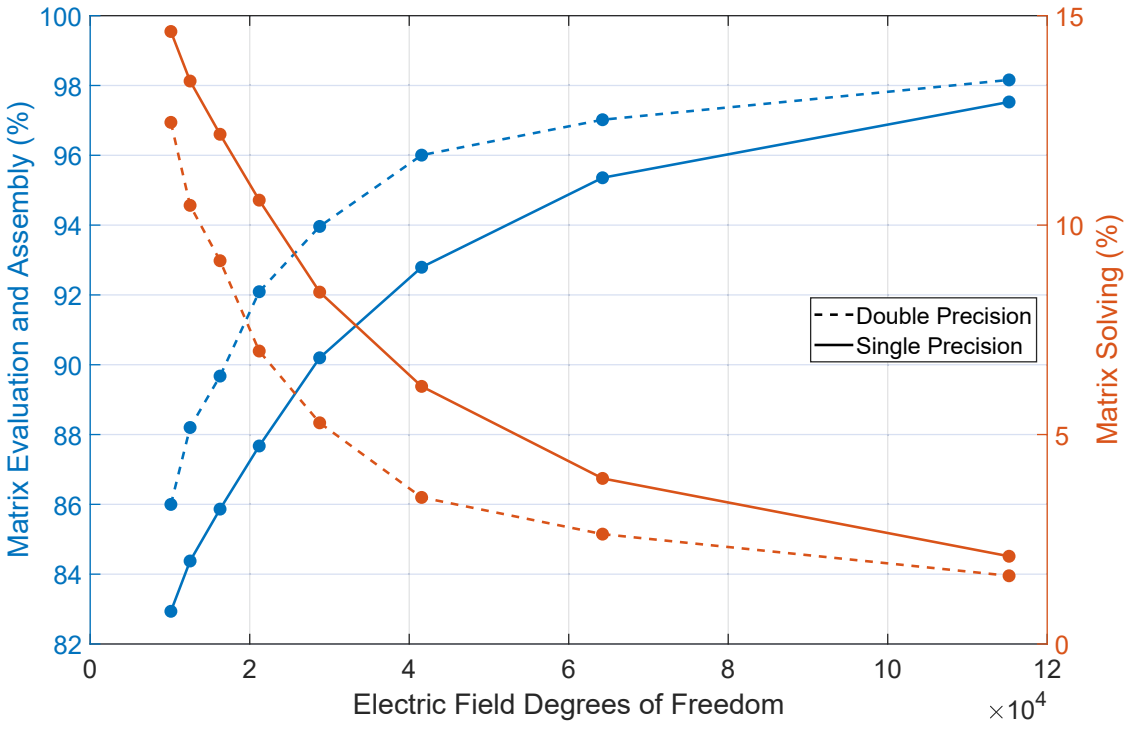

Figure 5.4: Breakdown of FETD computation time for an instantaneous nonlinearity in both single and double floating point precision.

both expected and surprising. With the number of elements in a given simulation and the computational effort required to perform the numerical integration, this data seemingly confirms the prediction that this part of the algorithm would contribute significantly to the overall execution time. However, that the effect would be so drastic as to dwarf even the repeated matrix solving is remarkable. Moreover, not only does the matrix evaluation far exceed time spent matrix solving, the trend of Figure 5.4 suggests the disparity gets *worse* as the number of variables is increased.

This result is odd considering that, for a specified order of basis functions and number of quadrature points, the work per element is constant, meaning that the computational complexity of matrix evaluation and assembly should be $\mathcal{O}(m)$, where m is the number of elements in the mesh. Since the number of edges scales roughly linearly with the number of elements, this means that the matrix evaluation and assembly should equally be $\mathcal{O}(n)$, where n is the number of edges or electric field degrees of freedom.

The solution of a matrix system via the conjugate gradient method, in contrast, is gener-

ally $\mathcal{O}(s\sqrt{\kappa})$ where s is the number of non-zeros in the matrix and κ is the matrix condition number [140]. For Finite-Element meshes of this kind, κ and s are generally $\mathcal{O}(n)$ [141], meaning the overall solution procedure is $\mathcal{O}(n^{3/2})$.

The expected portion of time spent matrix solving should thus be roughly approximated by:

$$\approx \mathcal{O}\left(\frac{n^{3/2}}{n^{3/2}+n}\right) \tag{5.16}$$

which is a monotonically increasing function in n. In other words, it should be expected for the proportion of time spent matrix solving to slowly *increase* rather than decrease as the simulation grows. The counterintuitive results of Figure 5.4 may nevertheless be explained by a few potential factors. The first is that, due to the nonlinearity, the Jacobian matrix fluctuates with each new iteration. In consequence, the condition number κ may vary significantly throughout the course of a given simulation and an estimate related n or m much more difficult to accurately produce. The second is that, due to the time-stepping nature of the simulation, a very good initial guess is available to the PCG and Newton-Raphson solvers. This can substantially reduce the number of Newton-Raphson and PCG iterations required to obtain the correct solution, accelerating the process considerably. For instance, despite the simulations depicted in Figure 5.4 having up to $\approx 10^5$ DoF, the PCG solver required on average no more than 45 iterations to obtain the solution, well below the estimated number of $\sqrt{n} \approx 340$. In fact, as the mesh is refined and the time step Δt becomes smaller and smaller, the difference between successive solutions decreases, as per equation (3.5). Hence, the average number of Newton-Raphson and PCG iterations needed to achieve convergence can remain roughly constant or actually *decrease* as the mesh is refined. The result is that, in reality, the computational complexity of root finding and matrix solving may be much less than that initially predicted, leading to the trends observed in Figure 5.4. This of course does not preclude the possibility of a turning point in which eventually the increase in condition number begins to dominate, reversing the observed trend. Moreover, due to the reliance of the Jacobian on the solution itself, this point may occur much sooner

or later depending on the exact problem being studied and its implementation. However, in general, it is quite safe to conclude that elemental matrix evaluation and assembly are the major bottlenecks within nonlinear FETD computations.

5.4.2 Speedup Results

With the hypothesized bottlenecks of nonlinear FETD methods confirmed, the parallel algorithm derived in Section 5.2 was implemented on an NVIDIA GPU using the CUDA language, with special attention given to the implementation details discussed in Section 5.3. Execution time was then compared between the CPU and GPU versions, with as much of the underlying structure and implementation of the CPU code being kept as possible in the GPU code to ensure a fair comparison. That being said, custom routines were written to perform the parallel elemental matrix evaluation, Gaussian Belief Propagation matrix solving, and other intervening operations. The GPU code was compiled in CUDA v9.2 and was executed on an NVIDIA GTX 1070Ti GPU with 2432 cores clocked at 1607 MHz with 8 GB of GDDR5 memory. The workstation hosting the graphics card was the same as that used for the CPU code in the previous section, meaning that any non-GPU sections of code were executed on the same processor as in the serial case. Since this model of GPU is capable of both double and single precision arithmetic, as with the CPU case, two versions of the code were written.

Figure 5.5 depicts the resulting speedup achieved by the GPU version of the algorithm, obtained by dividing the total execution time on the CPU by the total execution time on the GPU. For the smallest number of degrees of freedom tested (about 10^4), the GPU implementation is about 3-5 times faster than its CPU counterpart (note that the overhead time required to transfer data to and from host and GPU memory has been included in this data). As the number of degrees of freedom increases, however, the disparity between the two algorithms grows significantly, as does the difference between single and double precision performance. At the maximum extent tested, equal to roughly 115,000 DoF, the

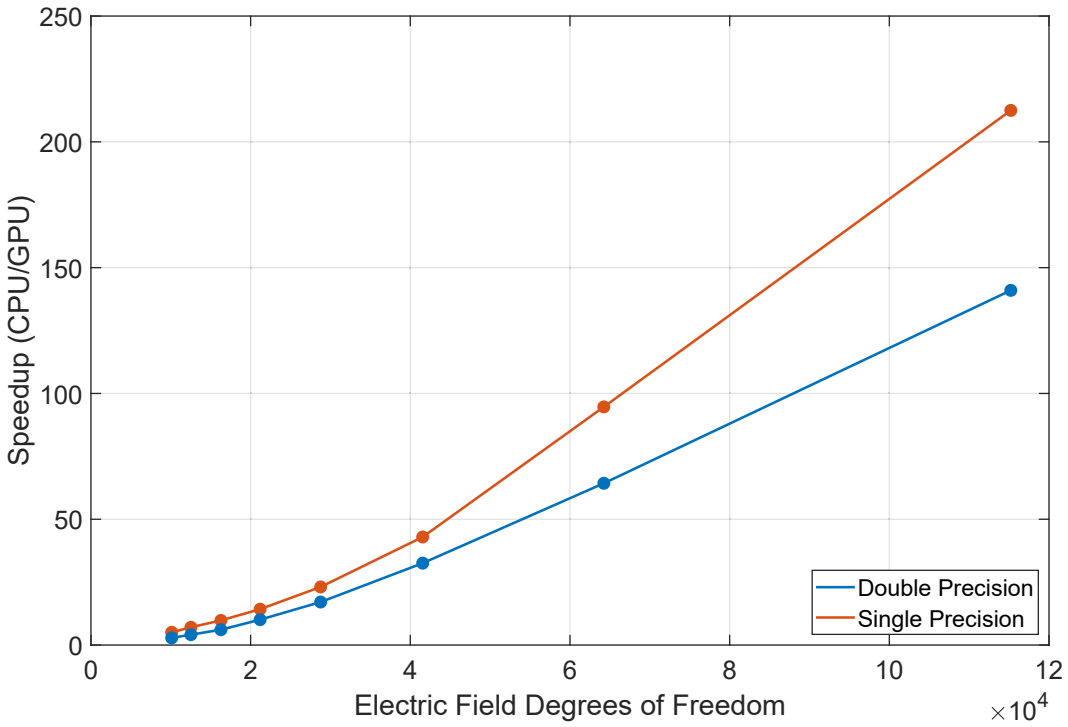

Figure 5.5: GPU over CPU speedup as a function of degrees of freedom for both single and double precision floating point simulations.

single precision GPU implementation performs an impressive 212 times faster than its CPU counterpart, whereas for double precision it performs 141 times faster. In the case of single precision, this represents a dramatic reduction in computation time from over 1.5 hours down to roughly 30 seconds. Further analysis of these results reveals, unsurprisingly, that most of the speedup comes from the parallelization of the elemental matrix evaluation. Indeed, with its independent nature and emphasis on simple multiplication and addition, numerical integration benefits immensely from the GPU's FLOPS. With the high proportion of computation time spent on these operations in the CPU version, it is unsurprising, therefore, that the result is a significant boost in speed. This isn't to say that the implementation of FGaBP is inconsequential. In fact, not only does the FGaBP algorithm solve the matrix system up to 5 times faster than the serial PCG, it also completely skips the step of matrix assembly, and allows data to remain on the GPU at all times, saving significantly on memory transfers.

As for the overall trend in Figure 5.5, in general it is found that the speedup increases

as the number of variables is increased. This is likely due to how the GPU schedules its threads internally. As the number of executing threads on the GPU increases, it is better able to interleave execution and hide latency [120]. For instance, if a group of threads stalls while waiting for data from memory, given sufficient additional threads, the GPU cores can pivot to other threads and perform useful work in the interim. Thus, performance increases as better utilization of GPU resources is achieved. When coupled with the data in Figure 5.4, this explains the increasing speedups observed in Figure 5.5. While not observed for the maximum number of variables studied here, it is however expected that at some point GPU utilization will max out, and the amount of speedup will stabilize.

Lastly, it is worth noting that these results are in some sense the best case performance speedups, as the domain being studied is 100% filled with nonlinear media. If a domain were to contain a smaller fraction of nonlinear media, the effects of parallelization would naturally decrease, since the number of elemental matrices which must be updated each iteration is reduced, decreasing the amount of overhead to start with. Moreover, while these results were presented for a two-dimensional mixed FETD test case, similar performance increase should be expected for both VWE and three-dimensional implementations. For 3D problems in particular, the number of degrees of freedom increases causing the work required to numerically evaluate each elemental matrix and solve the global systems to also increase. The result is that these operations will still constitute the vast majority of computation time, but are more able to take advantage of the GPU's resources due to the increase in workload. However, this is slightly counterbalanced by the higher memory requirements of 3D simulations hitting the GPU's global memory limit sooner than in 2D.

5.5 Summary

In this chapter the issue of the immense computational burden imposed by the inclusion of dielectric nonlinearity was addressed. While many approaches to reduce this burden are

possible, here the idea of exploiting parallelism within the nonlinear algorithms was put forth and developed. The Graphics Processing Unit (GPU) was introduced as a widely available and cost-effective source of wide-scale parallelism and some of the context and history of their use presented.

An analysis of the nonlinear algorithms was then presented with the goal of identifying and characterizing their overhead as well as determining their amenability to parallelization. Bottlenecks were identified in the repeated evaluation, assembly, and solution of the matrix systems which result from the nonlinearity, and mitigating parallel algorithms proposed. These included straightforward elemental matrix parallelization schemes, as well as the use of the Gaussian Belief Propagation algorithm to parallelize the solution of the Jacobian matrix system without ever explicitly forming the global matrices.

Finally, numerical studies were presented to verify and confirm the postulated bottlenecks as well as determine the performance of the parallelized GPU algorithms. To that end, both CPU and GPU algorithms were implemented in C++ and CUDA, respectively, and a series of profiling and comparison executions performed. These tests confirmed that matrix evaluation and assembly constitute a significant fraction of computation time, representing in the worst case up to 98% of total computation time. This helped explain, in part, the excellent performance of the GPU algorithm as compared to its CPU counterpart, which was found to perform up to 212 times faster. This represented a significant reduction in computation time, from roughly 1.5 hours down to 30 seconds, and demonstrates the GPU's effectiveness in making nonlinear computations more widely accessible.

Chapter 6

Conclusion

6.1 Summary

In conclusion, this thesis has introduced a family of Finite-Element-based numerical methods for the modeling of electrically complex material interactions with the electromagnetic field, including linear dispersion, instantaneous nonlinearity, and dispersive nonlinearity. The developed methods directly numerically approximate the nonlinear Maxwell's Equations and, contrary to many of the methods used to model these materials currently, do not rely upon approximative simplifications, giving them a much wider modeling ability and range of applicability. Moreover, by leveraging the Finite-Element method, as well as implicit temporal discretizations and the z-transform method, the derived techniques provide increased flexibility, accuracy, and stability over existing full-wave nonlinear Maxwell's Equations solvers.

Chapter 2 introduced the classic mixed and vector wave equation formulations of the FETD method for linear media. When temporally discretized via the Crank-Nicolson and Newmark-β formulations, respectively, these methods were shown to be remarkably similar in their characteristics and implementations, which aided greatly in their development. The z-transform was then introduced as the method of choice for incorporating linear dispersion into these base methods, allowing for the flexible modeling of dispersive media up to arbi-

trary order. With these foundations in place, instantaneous nonlinearity was introduced to the methods via the inclusion of higher-order susceptibilities within the permittivity. This resulted in significant modifications to the underlying methods, as the mass matrices effectively became functions of field strength and therefore time, changing values with each iteration. The solution of the resulting nonlinear system was obtained via the iterative Newton-Raphson method, with the required Jacobian being derived in closed-form. The inclusion of nonlinear dispersion was achieved by once again making use of the z-transform technique, with much of the derived nonlinear expressions, including the Jacobian, carrying over from the instantaneous case. Convergence studies were then undertaken with artificially manufactured solutions to demonstrate the proper functioning of the mixed and VWE methods and their comparable levels of accuracy. The applicability of nonlinear FETD analysis was also showcased via the simulation of several notable problems of physical significance, including the propagation of spatial and temporal solitons, and supercontinuum generation.

While Chapter 2 involved much of the mathematical derivation of these methods, Chapter 3 delved deeper into their implementation details and characteristics. In the case of instantaneous nonlinearities, it was shown that the elemental and global matrices are symmetric positive definite, whereas for dispersive nonlinearities, only symmetry could be conclusively proven. In terms of stability, both the mixed and VWE formulations were shown to numerically conserve the discrete analog of the electromagnetic energy, regardless of the time step size used, which is potentially an important indication of the unconditional numerical stability of the methods.

Chapter 4 introduced the notion of a perfectly matched layer and described how they can be of use for both the reduction in size of simulation domains and the emulation of ones which are infinite in extent. Given the immense proven utility of PMLs for linear media, a discussion of the various PML formulations was undertaken to find one suitable for adaptation to nonlinear media. In particular, it was found that the complex coordinate stretching interpretation of the PML was ideal for use with nonlinear media, due to its

ability to only affect the spatial part of the differential equations in question, while the material nonlinearity and dispersion are concentrated in the temporal part. This allowed the two techniques to be straightforwardly combined and minimized the amount of overhead required, thus simplifying the implementation considerably. Additionally, the absorption characteristics of the nonlinear PML were verified, demonstrating the simple and effective truncation of a temporal soliton to high accuracy.

Chapter 5, meanwhile, addressed one of the main drawbacks of the nonlinear FETD methods introduced in previous chapters: their immense computational burden. Since both the mass and Jacobian matrices are functions of time, they must be re-evaluated and re-assembled for each iteration of Newton-Raphson at each time step. Moreover, since the expressions for the local matrices rely on the field solution, the elemental matrices cannot be found in closed form and must be computed numerically. When combined with multiple Jacobian matrix solutions per time step, the resulting overhead as compared to simpler approximative methods is staggering. To mitigate these burdens, the use of parallelism was suggested in order to accelerate each step of the process. It was proposed that by using a simple novel parallel update scheme for elemental matrix evaluation, as well as the recently proposed Finite-Element Gaussian Belief Propagation method, that much of this debilitating overhead could be either accelerated or avoided entirely. While this parallelization strategy was architecture independent, a more focused discussion on how it could be implemented on GPU hardware was presented, as GPUs currently represent one of the most readily available and cost-effective sources of parallelism. Finally, a serial version of the algorithm was profiled to confirm suspected nonlinear overhead and bottlenecks and the resulting parallel GPU implementation benchmarked to determine its performance. Overall, the GPU version was found to significantly ease the computational burden and in the best case performed over 200 times faster than its serial counterpart.

6.2 Future Work

While the present thesis has made significant inroads toward the widespread development, refinement and, hopefully, adoption of FETD methods for electrically complex media, there nevertheless remains several promising avenues for future study.

For instance, while the implementation of the parallel algorithm presented in this thesis was GPU-centric, many other avenues of parallelization are possible. In the future, two such possible parallel platforms are multi-core CPUs as well as High Performance Computing (HPC) clusters. In particular, the use of large-scale computing clusters could be especially advantageous since in general they have far more memory available than current GPUs and can thus process much larger problems. Additionally, it may be possible to incorporate aspects of nonlinearity into other existing parallel algorithms to boost performance, such as within the Discontinuous Galerkin method [142]. Furthermore, initial studies, such as those reported in [143], also indicate it may be possible to directly integrate the Newton-Raphson iterations into the FGaBP solver, rather than treating the two as distinct sequential steps, yielding increased efficiency.

Of course, other optimizations to the serial nonlinear algorithm may also be possible. Notably, in Chapter 3 it was mentioned that it is likely possible to introduce interpolation basis functions for the permittivity such that exact closed-form expressions for the mass and Jacobian matrices are obtainable. If this were to be successfully implemented, significant efficiency gains would be experienced as it would do away entirely with the need to numerically integrate each nonlinear element. Another alternative avenue of exploration, as mentioned in Chapter 5, could be the adaptation of methods such as JFNK to nonlinear dielectric media.

Going forward, future investigations into these methods could also look into incorporating more recent tools and techniques. For instance, machine learning has the potential to furnish more accurate initial guesses for Newton-Raphson, based on previous data and material parameters. As long as a system can be trained to provide an initial guess which is better

than that used currently, the number of iterations can be reduced, further saving computation time.

While the numerical studies conducted in this thesis were all two-dimensional in nature, due in part to the extreme computational burden imposed by nonlinearity, none of the theory or derivations presented were dependent on this fact. Thus, given increased computational resources and the aid of the developed GPU algorithms (to reduce the immense computation times which are sure to result) future studies could further verify the effectiveness and accuracy of the methods derived in this thesis by applying them to problems in three spatial dimensions.

Lastly, while the permittivity model used in this thesis was quite general and capable of modeling most of the dominant effects routinely encountered in fields such as nonlinear optics, as mentioned in Subsection 1.2.1, there remain a variety of phenomena which were excluded. Further generalizing the methods discussed in this thesis to scenarios in which dielectric anisotropy, ferroelectricity, conductive losses, coupled problems, magnetic dispersion, or even magnetic nonlinearity are all present in conjunction with the phenomena studied here is undoubtedly possible. In particular, the extension of the present techniques to anisotropic media could prove particularly useful, given many glass and crystal structures tend to exhibit this behaviour. While at present use cases for such extremely varied phenomena existing in the same numerical simulation are few and far between, as technology continues to evolve it is reassuring to know that the foundations for such methods have nevertheless been laid, just in case.

Appendix A

Global and Local Matrices

A.1 Global Matrix Assembly

One of the key steps in implementing a Finite-Element method is in combining each of the local elemental matrix systems into a global one which spans the entire domain. If each element were completely independent of its neighbors, meaning they did not share any degrees of freedom, then this assembly procedure would simply consist of embedding each local system of the form:

$$[T^{(e)}]\{e^{(e)}\} = \{f^{(e)}\} \tag{A.1}$$

into a block-diagonal global system:

$$\begin{bmatrix} [T^{(1)}] & & & \\ & [T^{(2)}] & & \\ & & \ddots & \\ & & & [T^{(e)}] \end{bmatrix} \begin{bmatrix} \{e^{(1)}\} \\ \{e^{(2)}\} \\ \vdots \\ \{e^{(e)}\} \end{bmatrix} = \begin{bmatrix} \{f^{(1)}\} \\ \{f^{(2)}\} \\ \vdots \\ \{f^{(e)}\} \end{bmatrix}. \tag{A.2}$$

This block diagonal system may also be written as:

$$[T_{dis}]\{e_{dis}\} = \{f_{dis}\} \tag{A.3}$$

where the subscript "*dis*" indicates the matrices and vectors are *disjoint*, meaning all of the elemental matrices and degrees of freedom have been embedded independently and do not overlap.

Naturally, however, in discretizing a problem domain the resulting elements will generally not be all independent and will share degrees of freedom. The result is that the disjoint and combined global node numberings will differ and the global matrix will no longer be block-diagonal. To account for this, a new connectivity matrix $[Q]$ can be defined which relates the disjoint local numbering to the *conjoint* global numbering according to:

$$\{e_{dis}\} = [Q]\{e_{con}\}. \tag{A.4}$$

For instance, consider the two-element system depicted in Figure A.1 below. Due to their shared edge, rather than having a global system with six degrees of freedom, in reality there are only five true independent variables. Thus, in this case, the $[Q]$ matrix would relate the two numberings as follows:

$$\begin{bmatrix} e_1^{(1)} \\ e_2^{(1)} \\ e_3^{(1)} \\ e_4^{(2)} \\ e_5^{(2)} \\ e_6^{(2)} \end{bmatrix} = \begin{bmatrix} 1 & & & & & \\ & 1 & & & & \\ & & 1 & & & \\ & & 1 & & & \\ & & & 1 & & \\ & & & & 1 & \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \\ e_5 \end{bmatrix}. \tag{A.5}$$

Substituting the relation between the disjoint and conjoint variables of equation (A.4) into the block-diagonal system of equation (A.3) results in the following:

$$[T_{dis}][Q]\{e_{con}\} = \{f_{dis}\}. \tag{A.6}$$

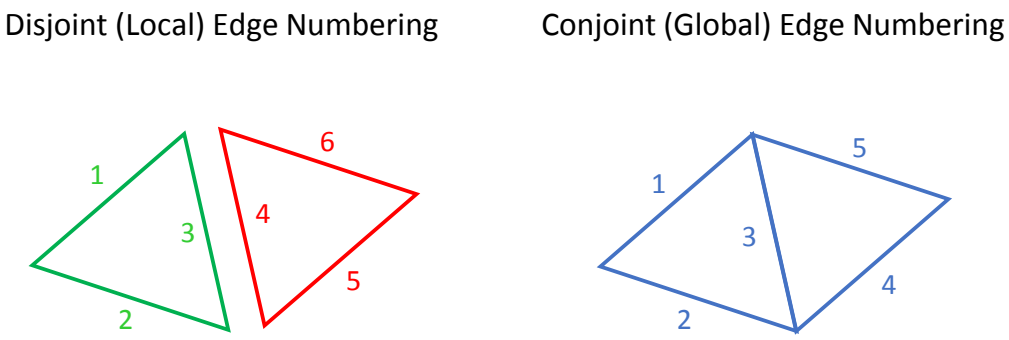

Figure A.1: Comparison of local vs global edge numberings for matrix assembly.

Unfortunately the system in (A.6), while only dependent on the unique global conjoint numbering, is overdetermined. This is due to the fact that the shared edges result in many equations (rows of $[T_{dis}]$) relating the same variables in a repetitive or redundant manner. In Figure A.1, for instance, relationships between local edges 1 & 3 and 4 & 6 do not represent two unique constraints, but rather a single one between global edges 1, 3, & 5. Thus, since the local equations do not represent unique relationships between distinct variables, they should instead be combined to yield a minimal set of constraints between the unique shared edges. Luckily, this can easily be accomplished by multiplying equation (A.6) by the transpose of the connectivity matrix:

$$[Q]^T[T_{dis}][Q]\{e_{con}\} = [Q]^T\{f_{dis}\}. \tag{A.7}$$

The above at last represents the global Finite-Element system, with the final conjoint global matrix $[T]$ being identified as:

$$[T] = [T_{con}] = [Q]^T[T_{dis}][Q] \tag{A.8}$$

and the equivalent conjoint global source term given by:

$$\{f\} = \{f_{con}\} = [Q]^T\{f_{dis}\}. \tag{A.9}$$

Alternatively, if the block-diagonal matrix $[T_{dis}]$ is expressed as a sum of elemental con-

tributions:

$$[T_{dis}] = \sum_e [T_{dis}^{(e)}] = \begin{bmatrix} [T^{(1)}] & & & \\ & 0 & & \\ & & 0 & \\ & & & 0 \end{bmatrix} + \begin{bmatrix} 0 & & & \\ & [T^{(2)}] & & \\ & & 0 & \\ & & & 0 \end{bmatrix} + \cdots + \begin{bmatrix} 0 & & & \\ & 0 & & \\ & & 0 & \\ & & & [T^{(e)}] \end{bmatrix} \quad \text{(A.10)}$$

the global matrix may also be expressed as:

$$[T] = [Q]^T \Big(\sum_e [T_{dis}^{(e)}] \Big) [Q] = \sum_e [Q]^T [T_{dis}^{(e)}] [Q] \triangleq \sum_e [T'^{(e)}] \quad \text{(A.11)}$$

which is of the exact same form as expressed in equation (2.26) of Chapter 2.

Lastly, in practice it is seldom necessary to explicitly construct the $[Q]$ matrix, with the global matrix being assembled from the local matrices according to the connectivity progressively as the local matrices are computed.

A.2 Symmetry and Positive-Definiteness

As was demonstrated in the previous section, a global Finite-Element matrix may be constructed from its local counterparts via a connectivity matrix:

$$[T] = [Q]^T [T_{dis}] [Q]. \quad \text{(A.12)}$$

In this section, several properties of this global matrix will be shown to stem directly from the elemental matrices from which it is constructed.

Firstly, taking the transpose of the above equation results in:

$$[T]^T = \big([Q]^T [T_{dis}] [Q] \big)^T = [Q]^T [T_{dis}]^T [Q] \quad \text{(A.13)}$$

from which it is straightforward to conclude that the global matrix $[T]$ will be symmetric if and only if the disjoint matrix $[T_{dis}]$ is also symmetric, i.e. if $[T_{dis}] = [T_{dis}]^T$. However, since the $[T_{dis}]$ matrix is block-diagonal, the requirement that it be symmetric is in turn equivalent to requiring that each of its constituent block matrices be symmetric. Thus, since here each of the constituent block matrices is a local elemental matrix, it is concluded that if the elemental matrices are symmetric, so too is the resulting global matrix.

As for positive-definiteness, equation (A.12) can be both left and right multiplied by an arbitrary vector $\{x\}$ such that:

$$\{x\}^T[T]\{x\} = \{x\}^T[Q]^T[T_{dis}][Q]\{x\}. \tag{A.14}$$

Defining a new vector $\{y\} = [Q]\{x\}$ the above can alternatively be expressed as:

$$\{x\}^T[T]\{x\} = \{y\}^T[T_{dis}]\{y\} \tag{A.15}$$

where it should be noted that $\{y\} = 0$ if and only if $\{x\} = 0$, since the $[Q]$ matrix has full column rank. Hence, it can once again be concluded that $[T]$ is symmetric positive-definite if and only if the disjoint matrix $[T_{dis}]$ has this property. As was the case for symmetry above, since $[T_{dis}]$ is block-diagonal, positive-definiteness thus also depends entirely on whether the constituent block matrices share this property. Hence, it is concluded that if the elemental matrices are positive-definite, so too is the resulting global matrix.

Appendix B

The Stretched Divergence Theorem

In this section an informal proof of the "stretched" Divergence Theorem used in the derivation of the coordinate-stretched PML of Chapter 4 will be shown for *constant* stretching factors s_ξ. In particular, it will be shown that:

$$\int_{\Omega^e} \nabla_D \cdot \vec{F}\, d\Omega = \frac{1}{s_x^D} \int_{\partial\Omega^e} F_x \hat{a}_x \cdot d\vec{S} + \frac{1}{s_y^D} \int_{\partial\Omega^e} F_y \hat{a}_y \cdot d\vec{S} + \frac{1}{s_z^D} \int_{\partial\Omega^e} F_z \hat{a}_z \cdot d\vec{S} \tag{B.1}$$

is true for an arbitrary cubic volume, as depicted in Figure B.1 below. The generalization to arbitrary volumes will not be explicitly shown, but is understood to follow by shrinking the cubic volume and using a tiling and gluing procedure akin to the limiting value of a Riemann sum to fill the macroscopic volume [144].

To begin, the stretched divergence operator on the left-hand side of equation (B.1) can

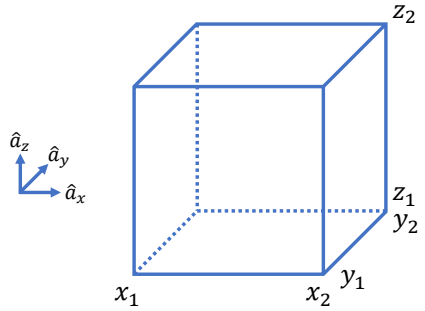

Figure B.1: Sample cubic volume for the application of the stretched Divergence Theorem.

be explicitly expanded and results in the following:

$$\int_{\Omega^e} \nabla_D \cdot \vec{F}\, d\Omega = \int_{\Omega^e} \frac{1}{s_x^D}\frac{\partial F_x}{\partial x} + \frac{1}{s_y^D}\frac{\partial F_y}{\partial y} + \frac{1}{s_z^D}\frac{\partial F_z}{\partial z} d\Omega. \tag{B.2}$$

Isolating the x component of the integrand, making use of the fact that s_ξ is assumed constant in space, and writing the volume integral explicitly gives:

$$\int_{\Omega^e} \frac{1}{s_x^D}\frac{\partial F_x}{\partial x}\, d\Omega = \frac{1}{s_x^D}\int_{z_1}^{z_2}\int_{y_1}^{y_2}\int_{x_1}^{x_2} \frac{\partial F_x}{\partial x}\, dxdydz. \tag{B.3}$$

However, using the Fundamental Theorem of Calculus [145], this may also be expressed as:

$$\int_{\Omega^e} \frac{1}{s_x^D}\frac{\partial F_x}{\partial x}\, d\Omega = \frac{1}{s_x^D}\int_{z_1}^{z_2}\int_{y_1}^{y_2} F_x\Big|_{x_2} - F_x\Big|_{x_1} dydz. \tag{B.4}$$

Moving on to the right-hand side of equation (B.1), the x-component surface integral can be evaluated solely over the left and right surfaces ($x = x_1$ and $x = x_2$), since there will be no contribution from other surfaces as their normals are orthogonal to $\hat{a}_x$:

$$\frac{1}{s_x^D}\int_{\partial\Omega} F_x\hat{a}_x \cdot d\vec{S} = \frac{1}{s_x^D}\int_{z_1}^{z_2}\int_{y_1}^{y_2} F_x\hat{a}_x \cdot (-\hat{a}_x) dydz\Big|_{x_1} + \frac{1}{s_x^D}\int_{z_1}^{z_2}\int_{y_1}^{y_2} F_x\hat{a}_x \cdot \hat{a}_x dydz\Big|_{x_2} \tag{B.5}$$

$$= \frac{1}{s_x^D}\int_{z_1}^{z_2}\int_{y_1}^{y_2} F_x\Big|_{x_2} - F_x\Big|_{x_1} dydz. \tag{B.6}$$

This last expression is found to be exactly equal to that obtained previously in equation (B.4), meaning that:

$$\int_{\Omega^e} \frac{1}{s_x^D}\frac{\partial F_x}{\partial x}\, d\Omega = \frac{1}{s_x^D}\int_{\partial\Omega^e} F_x\hat{a}_x \cdot d\vec{S}. \tag{B.7}$$

Repeating this exercise for the y and z components follows a similar procedure, and in like

manner yields:

$$\int_{\Omega^e} \frac{1}{s_y^D} \frac{\partial F_y}{\partial y} \, d\Omega = \frac{1}{s_y^D} \int_{\partial\Omega^e} F_y \hat{a}_y \cdot d\vec{S} \tag{B.8}$$

$$\int_{\Omega^e} \frac{1}{s_z^D} \frac{\partial F_z}{\partial z} \, d\Omega = \frac{1}{s_z^D} \int_{\partial\Omega^e} F_z \hat{a}_z \cdot d\vec{S}. \tag{B.9}$$

Combining the relations in equations (B.7), (B.8), and (B.9) thus proves equation (B.1) for the cubic volume of Figure B.1. As mentioned earlier, the generalization to arbitrary volumes will not be explicitly shown here, but follows using a similar cut and paste procedure to the "unstretched" Divergence Theorem.

Appendix C

PML Parameters

C.1 2D PML Parameters

In this section, the expression for the stiffness matrix associated with the coordinate-stretched PML will be derived for two-dimensional first-order basis functions. To begin, recall from Chapter 4, equation (4.21), that the PML stiffness matrix is associated with the following expression:

$$\sum_{i=1}^{l_e} \int_{\Omega^e} \left(\nabla_D \times \vec{W}_j^{(1)(e)}\right) \cdot \left(\nabla_B \times \vec{W}_i^{(1)(e)}\right) \{e^{(e)}\}_i \, d\Omega \tag{C.1}$$

where it must be remembered that the curl operators have been stretched.

As detailed in Jin [20], the 1-form basis functions may be expressed as:

$$\vec{W}_1^{(1)(e)} = l_1^{(e)} \left(L_1^{(e)} \nabla L_2^{(e)} - L_2^{(e)} \nabla L_1^{(e)} \right) \tag{C.2}$$

where $l_i^{(e)}$ is the length of the i^{th} edge, and the functions $L_i^{(e)}$ are the scalar triangle interpolation functions defined by:

$$L_i^{(e)} = \frac{1}{2\Delta^{(e)}} \left(a_i^{(e)} + b_i^{(e)} x + c_i^{(e)} y \right) \tag{C.3}$$

in which $\Delta^{(e)}$ is the triangle area and the constant weights $a_i^{(e)}$, $b_i^{(e)}$, and $c_i^{(e)}$ are deter-

mined from the triangle vertex coordinates. The remaining two vector basis functions are subsequently obtained by cyclically permuting the indices in (C.2).

The following stretched vector identity:

$$\nabla_D \times (f\vec{F}) = f\nabla_D \times \vec{F} + \nabla_D f \times \vec{F} \tag{C.4}$$

can be shown to hold true for a scalar function f and vector field $\vec{F}$ by a straightforward expansion of both sides. Then, taking the curl of equation (C.2) and applying this vector identity results in:

$$\nabla_D \times \vec{W}_1^{(1)(e)} = l_1^{(e)}\Big(L_1^{(e)}\nabla_D \times \nabla L_2^{(e)} + \nabla_D L_1^{(e)} \times \nabla L_2^{(e)} - L_2^{(e)}\nabla_D \times \nabla L_1^{(e)} - \nabla_D L_2^{(e)} \times \nabla L_1^{(e)}\Big). \tag{C.5}$$

However, since the basis functions $L_i^{(e)}$ are first-order, taking the gradient of equation (C.3) results in the following expression:

$$\nabla L_i^{(e)} = \frac{1}{2\Delta^{(e)}}\left(b_i^{(e)}\hat{a}_x + c_i^{(e)}\hat{a}_y\right) \tag{C.6}$$

which is *constant*. Hence, the curl of the $\nabla L_i^{(e)}$ terms vanish and equation (C.5) simplifies to:

$$\nabla_D \times \vec{W}_1^{(1)(e)} = l_1^{(e)}\left(\nabla_D L_1^{(e)} \times \nabla L_2^{(e)} - \nabla_D L_2^{(e)} \times \nabla L_1^{(e)}\right). \tag{C.7}$$

Substituting the definition of $L_i^{(e)}$ from equation (C.3) and performing the required normal and stretched gradients then yields the following expression for the curl of $\vec{W}_1^{(1)(e)}$:

$$\nabla_D \times \vec{W}_1^{(1)(e)} = \frac{l_1^{(e)}}{4(\Delta^{(e)})^2}\left(b_1^{(e)}c_2^{(e)} - b_2^{(e)}c_1^{(e)}\right)\left(\frac{1}{s_x^D} + \frac{1}{s_y^D}\right)\hat{a}_z. \tag{C.8}$$

Furthermore, as noted in Jin, the expression $b_1^{(e)}c_2^{(e)} - b_2^{(e)}c_1^{(e)}$ can be identified as twice the

area of the triangle, such that the curl is at last found to be:

$$\nabla_D \times \vec{W}_1^{(1)(e)} = \frac{l_1^{(e)}}{2\Delta^{(e)}} \left(\frac{1}{s_x^D} + \frac{1}{s_y^D} \right) \hat{a}_z. \tag{C.9}$$

Performing the same computation for the other two vector basis functions likewise yields a similar result, so that in general:

$$\nabla_D \times \vec{W}_j^{(1)(e)} = \frac{l_j^{(e)}}{2\Delta^{(e)}} \left(\frac{1}{s_x^D} + \frac{1}{s_y^D} \right) \hat{a}_z \tag{C.10}$$

$$\nabla_B \times \vec{W}_i^{(1)(e)} = \frac{l_i^{(e)}}{2\Delta^{(e)}} \left(\frac{1}{s_x^B} + \frac{1}{s_y^B} \right) \hat{a}_z. \tag{C.11}$$

Making this substitution into the expression for the stiffness matrix in equation (C.1) results in:

$$\sum_{i=1}^{l_e} \int_{\Omega^e} \frac{l_i^{(e)} l_j^{(e)}}{4(\Delta^{(e)})^2} \left(\frac{1}{s_x^D} + \frac{1}{s_y^D} \right) \left(\frac{1}{s_x^B} + \frac{1}{s_y^B} \right) d\Omega \{e^{(e)}\}_i \tag{C.12}$$

which after recognizing the integrand as constant in space at last results in the final expression for the stiffness matrix:

$$\begin{aligned} \sum_{i=1}^{l_e} \left(\frac{1}{s_x^D} + \frac{1}{s_y^D} \right) \left(\frac{1}{s_x^B} + \frac{1}{s_y^B} \right) \left(\frac{l_i^{(e)} l_j^{(e)}}{4\Delta} \right) \{e^{(e)}\}_i & \\ = \left(\frac{1}{s_x^D} + \frac{1}{s_y^D} \right) \left(\frac{1}{s_x^B} + \frac{1}{s_y^B} \right) [\hat{S}^{(e)}] \{e^{(e)}\} & \\ = p(\omega) [\hat{S}^{(e)}] \{e^{(e)}\} & \end{aligned} \tag{C.13}$$

which is precisely the expression reported in equation (4.24) of Chapter 4.

C.2 3D PML Parameters

In this section, a similar procedure to that presented in the previous section will be carried out in order to derive the stiffness matrix and convolution functions required to implement

a PML in three spatial dimensions, using first-order tetrahedral elements.

In like manner to the two-dimensional case, the expressions for the stiffness matrix and tetrahedral first-order 1-forms in three dimensions are given by:

$$\sum_{i=1}^{l_e} \int_{\Omega^e} \left(\nabla_D \times \vec{W}_j^{(1)(e)}\right) \cdot \left(\nabla_B \times \vec{W}_i^{(1)(e)}\right) \{e^{(e)}\}_i \, d\Omega \tag{C.14}$$

and

$$\vec{W}_i^{(1)(e)} = l_i^{(e)} \left(L_{i_1}^{(e)} \nabla L_{i_2}^{(e)} - L_{i_2}^{(e)} \nabla L_{i_1}^{(e)}\right) \tag{C.15}$$

where i_1 and i_2 are the vertices associated with the i^{th} edge, and $l_i^{(e)}$ is its length [20]. The scalar tetrahedral basis functions in turn are now given by:

$$L_i^{(e)} = \frac{1}{6V^{(e)}} \left(a_i^{(e)} + b_i^{(e)} x + c_i^{(e)} y + d_i^{(e)} z\right) \tag{C.16}$$

in which $V^{(e)}$ is the volume of the tetrahedron and the constants $a_i^{(e)}$, $b_i^{(e)}$, $c_i^{(e)}$ and $d_i^{(e)}$ are determined from its vertex coordinates.

Since the expression in equation (C.15) is precisely the same as that seen earlier in equation (C.2), and since the current basis functions are also first-order, applying the stretched vector identity of equation (C.4) to the curl of (C.15) results in the same expression obtained earlier in (C.7):

$$\nabla_D \times \vec{W}_i^{(1)(e)} = l_i^{(e)} \left(\nabla_D L_{i_1}^{(e)} \times \nabla L_{i_2}^{(e)} - \nabla_D L_{i_2}^{(e)} \times \nabla L_{i_1}^{(e)}\right). \tag{C.17}$$

This time, however, the gradients of the scalar basis functions are found to be:

$$\nabla L_i^{(e)} = \frac{1}{6V^{(e)}} \left(b_i^{(e)} \hat{a}_x + c_i^{(e)} \hat{a}_y + d_i^{(e)} \hat{a}_z\right) \tag{C.18}$$

such that when substituted into equation (C.17) above, results in the following:

$$\nabla_D \times \vec{W}_i^{(1)(e)} = \frac{1}{36(V^{(e)})^2}\Bigg(\left(\frac{1}{s_y^D}+\frac{1}{s_z^D}\right)\left(c_{i_1}^{(e)}d_{i_2}^{(e)} - c_{i_2}^{(e)}d_{i_1}^{(e)}\right)\hat{a}_x + \left(\frac{1}{s_x^D}+\frac{1}{s_z^D}\right)\left(d_{i_1}^{(e)}b_{i_2}^{(e)} - d_{i_2}^{(e)}b_{i_1}^{(e)}\right)\hat{a}_y + \left(\frac{1}{s_x^D}+\frac{1}{s_y^D}\right)\left(b_{i_1}^{(e)}c_{i_2}^{(e)} - b_{i_2}^{(e)}c_{i_1}^{(e)}\right)\hat{a}_z\Bigg). \quad \text{(C.19)}$$

Making the substitution of equation (C.19) into equation (C.14) and performing the integral over the element (noting again that the integrand is constant in space) at last yields the, quite involved, expression for the stiffness matrix:

$$\begin{aligned}\sum_{i=1}^{l_e}\int_{\Omega^e}(\nabla_D \times \vec{W}_j^{(1)(e)})\cdot(\nabla_B \times \vec{W}_i^{(1)(e)})\{e^{(e)}\}_i\, d\Omega = \\ \sum_{i=1}^{l_e}\Bigg(\frac{1}{6^4(V^{(e)})^3}\left(\frac{1}{s_y^D}+\frac{1}{s_z^D}\right)\left(\frac{1}{s_y^B}+\frac{1}{s_z^B}\right)\left(c_{i_1}^{(e)}d_{i_2}^{(e)} - c_{i_2}^{(e)}d_{i_1}^{(e)}\right)\left(c_{j_1}^{(e)}d_{j_2}^{(e)} - c_{j_2}^{(e)}d_{j_1}^{(e)}\right) \\ + \frac{1}{6^4(V^{(e)})^3}\left(\frac{1}{s_x^D}+\frac{1}{s_z^D}\right)\left(\frac{1}{s_x^B}+\frac{1}{s_z^B}\right)\left(d_{i_1}^{(e)}b_{i_2}^{(e)} - d_{i_2}^{(e)}b_{i_1}^{(e)}\right)\left(d_{j_1}^{(e)}b_{j_2}^{(e)} - d_{j_2}^{(e)}b_{j_1}^{(e)}\right) \\ + \frac{1}{6^4(V^{(e)})^3}\left(\frac{1}{s_x^D}+\frac{1}{s_y^D}\right)\left(\frac{1}{s_x^B}+\frac{1}{s_y^B}\right)\left(b_{i_1}^{(e)}c_{i_2}^{(e)} - b_{i_2}^{(e)}c_{i_1}^{(e)}\right)\left(b_{j_1}^{(e)}c_{j_2}^{(e)} - b_{j_2}^{(e)}c_{j_1}^{(e)}\right)\Bigg)\{e^{(e)}\}_i. \quad \text{(C.20)}\end{aligned}$$

However, by defining the following quantities:

$$p_x^{(e)}(\omega) = \left(\frac{1}{s_y^D}+\frac{1}{s_z^D}\right)\left(\frac{1}{s_y^B}+\frac{1}{s_z^B}\right) \quad \text{(C.21)}$$

$$p_y^{(e)}(\omega) = \left(\frac{1}{s_x^D}+\frac{1}{s_z^D}\right)\left(\frac{1}{s_x^B}+\frac{1}{s_z^B}\right) \quad \text{(C.22)}$$

$$p_z^{(e)}(\omega) = \left(\frac{1}{s_x^D}+\frac{1}{s_y^D}\right)\left(\frac{1}{s_x^B}+\frac{1}{s_y^B}\right) \quad \text{(C.23)}$$

$$[S_x^{(e)}]_{ij} = \frac{1}{6^4(V^{(e)})^3}\left(c_{i_1}^{(e)}d_{i_2}^{(e)} - c_{i_2}^{(e)}d_{i_1}^{(e)}\right)\left(c_{j_1}^{(e)}d_{j_2}^{(e)} - c_{j_2}^{(e)}d_{j_1}^{(e)}\right) \quad \text{(C.24)}$$

$$[S_y^{(e)}]_{ij} = \frac{1}{6^4(V^{(e)})^3}\left(d_{i_1}^{(e)}b_{i_2}^{(e)} - d_{i_2}^{(e)}b_{i_1}^{(e)}\right)\left(d_{j_1}^{(e)}b_{j_2}^{(e)} - d_{j_2}^{(e)}b_{j_1}^{(e)}\right) \quad \text{(C.25)}$$

$$[S_z^{(e)}]_{ij} = \frac{1}{6^4(V^{(e)})^3}\left(b_{i_1}^{(e)}c_{i_2}^{(e)} - b_{i_2}^{(e)}c_{i_1}^{(e)}\right)\left(b_{j_1}^{(e)}c_{j_2}^{(e)} - b_{j_2}^{(e)}c_{j_1}^{(e)}\right) \quad \text{(C.26)}$$

equation (C.20) can be simplified considerably to:

$$\sum_{i=1}^{l_e} \int_{\Omega^e} (\nabla_D \times \vec{W}_j^{(1)(e)}) \cdot (\nabla_B \times \vec{W}_i^{(1)(e)}) \{e^{(e)}\}_i \, d\Omega$$
$$= p_x^{(e)}(\omega)[S_x^{(e)}]\{e^{(e)}\} + p_y^{(e)}(\omega)[S_y^{(e)}]\{e^{(e)}\} + p_z^{(e)}(\omega)[S_z^{(e)}]\{e^{(e)}\}. \quad \text{(C.27)}$$

As was the case in two dimensions, the simplified expression of equation (C.27) consists entirely of frequency-dependent functions multiplied by constant matrices. Thus, the resulting convolutions in the time-domain can easily be modeled using the same techniques studied in Chapter 4. Lastly, it should be noted that if the PML is turned off, $p_x^{(e)} = p_y^{(e)} = p_z^{(e)} = 4$, and the stiffness matrix reduces exactly to the expression given in [20] for regular FETD.

Bibliography

[1] A. Z. Capri, *Quips, Quotes and Quanta: An Anecdotal History of Physics*. Singapore: World Scientific, 2011.

[2] J. D. Jackson, *Classical Electrodynamics*, 3rd ed. Hoboken, NJ: John Wiley & Sons, 1999.

[3] C. Gabriel, *Dielectric Properties of Biological Materials*, ser. Handbook of Biological Effects of Electromagnetic Fields: Bioengineering and Biophysical Aspects of Electromagnetic Fields. Boca Raton, FL: Taylor and Francis, 2007, vol. 1.

[4] C. Rappaport, "A dispersive microwave model for human breast tissue suitable for FDTD computation," *IEEE Antennas and Wireless Propagation Letters*, vol. 6, pp. 179 – 181, Apr. 2007.

[5] F. L. Teixeira, W. C. Chew, M. Straka, M. L. Oristaglio, and T. Wang, "Finite-difference time-domain simulation of ground penetrating radar on dispersive, inhomogeneous, and conductive soils," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 36, no. 6, pp. 1928 – 1937, Nov. 1998.

[6] T. Sato, T. Wakayama, and K. Takemura, "An imaging algorithm of objects embedded in a lossy dispersive medium for subsurface radar data processing," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 38, no. 1, pp. 296–303, Jan. 2000.

[7] M. DiDomenico, "Material dispersion in optical fiber waveguides," *Applied Optics*, vol. 11, no. 3, pp. 652–654, 1972.

[8] I. Walmsley, "The role of dispersion in ultrafast optics," *Review of Scientific Instruments*, vol. 72, no. 1, pp. 1–29, Jan. 2001.

[9] T. Kinjo, Y. Namihira, K. Arakaki, T. Koga, S. F. Kaijage, S. M. Abdur Razzak, F. Begum, S. Nozaki, and H. Higa, "Design of highly nonlinear dispersion-flattened square photonic crystal fiber for medical applications," *Optical Review*, vol. 17, no. 2, pp. 61–65, Mar. 2010.

[10] S. Kim, K. Kim, F. Rotermund, and H. Lim, "Computational design of one-dimensional nonlinear photonic crystals with material dispersion for efficient second-harmonic generation," *Optics Express*, vol. 17, no. 21, pp. 19 075–19 084, 2009.

[11] R. M. Joseph and A. Taflove, "Spatial soliton deflection mechanism indicated by FD-TD Maxwell's equations modeling," *IEEE Photonics Technology Letters*, vol. 6, no. 10, pp. 1251–1254, Oct. 1994.

[12] Z. Lubin, J. H. Greene, and A. Taflove, "FDTD computational study of ultra-narrow TM non-paraxial spatial soliton interactions," *IEEE Microwave and Wireless Components Letters*, vol. 21, no. 5, pp. 228–230, May 2011.

[13] A. M. Woodward and D. B. Kell, "On the nonlinear dielectric properties of biological systems," *Bioelectrochemistry and Bioenergetics*, vol. 24, pp. 83 – 100, 1990.

[14] A. M. Woodward and D. B. Kell, "Confirmation by using mutant strains that the membrane-bound H^+-ATPase is the major source of nonlinear dielectricity in saccharomyces cerevisiae," *FEMS Microbiology Letters*, vol. 84, no. 1, pp. 91–96, Nov. 1991.

[15] A. M. Woodward and D. B. Kell, "On the relationship between the nonlinear dielectric properties and respiratory activity of the obligately aerobic bacterium micrococcus luteus," *Bioelectrochemistry and Bioenergetics*, vol. 26, no. 3, pp. 423–439, Dec. 1991.

[16] G. T. S. Mercier, A. Palanisami, and J. H. M. Jr., "Nonlinear dielectric spectroscopy for label-free detection of respiratory activity in whole cells," *Biosensors and Bioelectronics*, vol. 25, pp. 2107 – 2114, Feb. 2010.

[17] J. Li and H. Chen, "Negative optical phenomena," in *Proceedings of Advances in Optoelectronics and Micro/Nano-Optics*. Guangzhou, China: IEEE, Dec. 2010.

[18] J. Lin, D. Li, and W. Yu, "A review of zero index metamaterials," in *Proceedings of the 2017 IEEE Electrical Design of Advanced Packaging and Systems Symposium (EDAPS)*. Haining, China: IEEE, Dec. 2017.

[19] A. Taflove and S. C. Hagness, *Computational Electrodynamics: The Finite-Difference Time-Domain Method*, 3rd ed. Boston, MA: Artech House, 2005.

[20] J. Jin, *The Finite Element Method in Electromagnetics*, 3rd ed. Hoboken, NJ: Wiley-IEEE Press, 2014.

[21] W. H. Hayt and J. A. Buck, *Engineering Electromagnetics*, 8th ed. New York, NY: McGraw-Hill, 2011.

[22] A. V. Oppenheim, A. S. Willsky, and S. H. Nawab, *Signals and Systems*, 2nd ed. Upper Saddle River, NJ: Prentice Hall, 1996.

[23] R. Boyd, *Nonlinear Optics*, 3rd ed. Burlington, MA: Academic Press, 2008.

[24] I. Garrett, "Nonlinear effects in fibre communications: An overview," in *Proceedings of the IEE Colloquium on Non-Linear Effects in Fibre Communications*. London, UK: IET, Nov. 1990.

[25] J. M. Dudley, G. Genty, and S. Coen, "Supercontinuum generation in photonic crystal fiber," *Reviews of Modern Physics*, vol. 78, pp. 1135–1184, Oct. 2006.

[26] L. Sirleto, M. A. Ferrara, and A. Vergara, "Towards applications of stimulated Raman scattering in nanophotonics," in *Proceedings of the 20th International Conference on Transparent Optical Networks*. Bucharest, Romania: 2018, Jul. 2018.

[27] R. M. Joseph and A. Taflove, "FDTD Maxwell's equations models for nonlinear electrodynamics and optics," *IEEE Transactions on Antennas and Propagation*, vol. 45, no. 3, pp. 364–374, Mar. 1997.

[28] K. J. Blow and D. Wood, "Theoretical description of transient stimulated Raman scattering in optical fibers," *IEEE Journal of Quantum Electronics*, vol. 25, no. 12, pp. 2665 – 2673, Dec. 1989.

[29] M. F. S. Ferreira, *Nonlinear Effects in Optical Fibers*. Hoboken, NJ: Wiley-OSA, 2011.

[30] R. Bonifacio, R. M. Caloi, and C. maroli, "The slowly varying envelope approximation revisited," *Optics Communications*, vol. 101, no. 3-4, pp. 185–187, Aug. 1993.

[31] M. D. Feit and A. Fleck, "Calculation of dispersion in graded-index multimode fibers by a propagating-beam method," *Applied Optics*, vol. 18, no. 16, pp. 2843–2851, 1979.

[32] J. V. Roey, J. van der Donk, and P. E. Lagasse, "Beam-propagation method: Analysis and assessment," *J. Opt. Soc. Am.*, vol. 71, no. 7, pp. 803 – 810, Jul. 1981.

[33] J. A. C. Weideman and B. M. Herbst, "Split-step methods for the solution of the nonlinear Schrödinger equation," *SIAM Journal on Numerical Analysis*, vol. 23, no. 3, pp. 485–507, 1986.

[34] M. Aleshams, A. Zarifkar, and M. H. Sheikhi, "Split-step Fourier transform method in modeling of pulse propagation in dispersive nonlinear optical fibers," in *Proceedings of*

the Second International Conference on Advanced Optoelectronics and Lasers. Yalta, Ukraine: IEEE, Sept. 2005.

[35] J. K. Ranka and A. L. Gaeta, "Breakdown of the slowly varying envelope approximation in the self-focusing of ultrashort pulses," *Optics Letters*, vol. 23, no. 7, pp. 534–536, 1998.

[36] L. Thylen, "The beam propagation method: An analysis of its applicability," *Optical and Quantum Electronics*, vol. 15, no. 5, pp. 433–439, 1983.

[37] F. L. Teixeira, "Time-domain finite-difference and finite-element methods for Maxwell equations in complex media," *IEEE Transactions on Antennas and Propagation*, vol. 56, no. 8, pp. 2150–2166, Aug. 2008.

[38] C. T. A. Johnk, *Engineering Electromagnetic Fields and Waves*, 2nd ed. New York, NY: John Wiley & Sons, 1988.

[39] 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, May 1966.

[40] R. Luebbers, F. P. Hunsberger, K. S. Kunz, R. B. Standler, and M. Schneider, "A frequency-dependent finite-difference time-domain formulation for dispersive materials," *IEEE Transactions on Electromagnetic Compatibility*, vol. 32, no. 3, pp. 222–227, Aug. 1990.

[41] D. F. Kelley and R. J. Luebbers, "Piecewise linear recursive convolution for dispersive media using FDTD," *IEEE Transactions on Antennas and Propagation*, vol. 44, no. 6, pp. 792–797, Jun. 1996.

[42] R. M. Joseph, S. C. Hagness, and A. Taflove, "Direct time integration of Maxwell's equations in linear dispersive media with absorption for scattering and propagation of

femtosecond electromagnetic pulses," *Optics Letters*, vol. 16, no. 18, pp. 1412–1414, 1991.

[43] D. M. Sullivan, "Frequency-dependent FDTD methods using z transforms," *IEEE Transactions on Antennas and Propagation*, vol. 40, no. 10, pp. 1223–1230, Oct. 1992.

[44] D. M. Sullivan, "Z-transform theory and the FDTD method," *IEEE Transactions on Antennas and Propagation*, vol. 44, no. 1, pp. 28–34, Jan. 1996.

[45] A. Akbarzadeh-Sharbaf and D. D. Giannacopoulos, "Finite-element time-domain solution of the vector wave equation in doubly dispersive media using mobius transformation technique," *IEEE Transactions on Antennas and Propagation*, vol. 61, no. 8, pp. 4158–4166, Aug. 2013.

[46] P. Tran, "Optical switching with a nonlinear photonic crystal: A numerical study," *Optics Letters*, vol. 21, no. 15, pp. 1138–1140, 1996.

[47] V. Van and K. Chaudhuri, Sujeet, "A hybrid implicit-explicit FDTD scheme for nonlinear optical waveguide modeling," *IEEE Transactions on Microwave Theory and Techniques*, vol. 47, no. 5, pp. 540–545, May 1999.

[48] P. M. Goorjian and A. Taflove, "Direct time integration of Maxwell's equations in nonlinear dispersive media for propagation and scattering of femtosecond electromagnetic solitons," *Optics Letters*, vol. 17, no. 3, pp. 180–182, Feb. 1992.

[49] D. M. Sullivan, "Nonlinear FDTD formulations using z-transforms," *IEEE Transactions on Microwave Theory and Techniques*, vol. 43, no. 3, pp. 676 – 682, Mar. 1995.

[50] A. C. Cangellaris and D. B. Wright, "Analysis of the numerical error caused by the stair-stepped approximation of a conducting boundary in FDTD simulations of electromagnetic phenomena," *IEEE Transactions on Antennas and Propagation*, vol. 39, no. 10, pp. 1518–1525, Oct. 1991.

[51] A. Akyurtlu, D. H. Werner, V. Veremey, D. J. Steich, and K. Aydin, "Staircasing errors in FDTD at an air-dielectric interface," *IEEE Microwave and Guided Wave Letters*, vol. 9, no. 11, pp. 444–446, Nov. 1999.

[52] A. Pereda, L. A. Vielva, A. Vegas, and A. Prieto, "Analyzing the stability of the FDTD technique by combining the Von Neumann method with the Routh-Hurwitz criterion," *IEEE Transactions on Microwave Theory and Techniques*, vol. 49, no. 2, pp. 377–381, Feb. 2001.

[53] S. Wang and F. L. Teixeira, "Some remarks on the stability of time-domain electromagnetic simulations," *IEEE Transactions on Antennas and Propagation*, vol. 52, no. 3, pp. 895–898, Mar. 2004.

[54] K.-P. Hwang and A. C. Cangellaris, "Effective permittivities for second-order accurate FDTD equations at dielectric interfaces," *IEEE Microwave and Wireless Components Letters*, vol. 11, no. 4, pp. 158–160, Apr. 2001.

[55] D. S. Abraham, A. N. Marques, and J.-C. Nave, "A correction function method for the wave equation with interface jump conditions," *Journal of Computational Physics*, vol. 353, pp. 281–299, Jan. 2018.

[56] S.-B. Shi, W. Shao, X.-K. Wei, X.-S. Yang, and B.-Z. Wang, "A new unconditionally stable FDTD method based on the Newmark-beta algorithm," *IEEE Transactions on Microwave Theory and Techniques*, vol. 64, no. 12, pp. 4082–4090, Dec. 2016.

[57] A. Akbarzadeh-Sharbaf and D. D. Giannacopoulos, "On the development of nonoverlapping and stable hybrid FETD-FDTD formulations," *IEEE Transactions on Antennas and Propagation*, vol. 62, no. 12, pp. 6299–6306, Dec. 2014.

[58] D. Jiao and J.-M. Jin, "Time-domain finite-element modeling of dispersive media," *IEEE Microwave and Wireless Components Letters*, vol. 11, no. 5, pp. 220 – 222, May 2001.

[59] N. S. Stoykov, T. A. Kuiken, M. M. Lowery, and A. Taflove, "Finite-element time-domain algorithms for modeling linear Debye and Lorentz dielectric dispersions at low frequencies," *IEEE Transactions on Biomedical Engineering*, vol. 50, no. 9, pp. 1100–1107, Sept. 2003.

[60] B. Donderici and F. L. Teixeira, "Mixed finite-element time-domain method for transient Maxwell equations in doubly dispersive media," *IEEE Transactions on Microwave Theory and Techniques*, vol. 56, no. 1, pp. 113–120, Jan. 2008.

[61] A. Akbarzadeh-Sharbaf and D. D. Giannacopoulos, "A stable and efficient direct time integration of the vector wave equation in the finite-element time-domain method for dispersive media," *IEEE Transactions on Antennas and Propagation*, vol. 63, no. 1, pp. 314–321, Jan. 2015.

[62] S. Yan and J.-M. Jin, "Theoretical formulation of a time-domain finite element method for nonlinear magnetic problems in three dimensions," *Progress in Electromagnetics Research*, vol. 153, pp. 33–55, 2015.

[63] S. Yan and J.-M. Jin, "Analysis of nonlinear electromagnetic problems using time-domain finite element method," *Radio Science Meeting (Joint with AP-S Sympsium), USNC-URSI*, p. 99, Jul. 2013.

[64] S. Yan and J.-M. Jin, "Three-dimensional time-domain finite-element simulation of dielectric breakdown based on nonlinear conductivity model," *IEEE Transactions on Antennas and Propagation*, vol. 64, no. 7, pp. 3018–3026, Jul. 2016.

[65] S. Yan and J.-M. Jin, "Time-domain finite element modeling of nonlinear conductivity using Newton's method," *2015 IEEE International Symposium on Antennas and Propagation & USNC/URSI National Science Meeting*, Jul. 2015.

[66] B. Zhu, H. Yang, and J. Chen, "A novel finite element time domain method for nonlinear Maxwell's equations based on the parametric quadratic programming method," *Microwave and Optical Technology Letters*, vol. 57, no. 7, pp. 1640–1645, Jul. 2015.

[67] A. Fisher, D. White, and G. Rodrigue, "An efficient vector finite element method for nonlinear electromagnetic modeling," *Journal of Computational Physics*, vol. 225, no. 2, pp. 1331–1346, Feb. 2007.

[68] D. A. White, "Discrete time vector finite element methods for solving Maxwell's equations on 3d unstructured grids," Ph.D. dissertation, University of California, Davis, 1997.

[69] W. A. Artuzi, "Improving the Newmark time integration scheme in finite element time domain methods," *IEEE Microwave and Wireless Components Letters*, vol. 15, no. 12, pp. 898 – 900, Dec. 2005.

[70] R. A. Chilton and R. Lee, "The discrete origin of FETD-Newmark late time instability, and a correction scheme," *Journal of Computational Physics*, vol. 224, no. 2, pp. 1293 – 1306, June 2007.

[71] D. Sun, J. Manges, X. Yuan, and Z. Cendes, "Spurious modes in finite-element methods," *IEEE Antennas and Propagation Magazine*, vol. 37, no. 5, pp. 12 – 24, Oct. 1995.

[72] H. Whitney, *Geometric Integration Theory*. Princeton, NJ: Princeton University Press, 1957.

[73] J. C. Nedelec, "Mixed finite elements in $\mathbb{R}^3$," *Numerische Mathematik*, vol. 35, no. 3, pp. 315 – 341, Sept. 1980.

[74] A. Bossavit, "Whitney forms: A class of finite elements for three-dimensional computations in electromagnetism," *IEEE Proceedings A - Physical Science, Measurement*

and Instrumentation, Management and Education - Reviews, vol. 135, no. 8, pp. 493 – 500, Nov. 1988.

[75] Y. Zhu and A. C. Cangellaris, *Multigrid Finite Element Methods for Electromagnetic Field Modeling*. Hoboken, NJ: John Wiley and Sons, 2006.

[76] J. Crank and P. Nicolson, "A practical method for numerical evaluation of solutions of partial differential equations of the heat-conduction type," *Mathematical Proceedings of the Cambridge Philosophical Society*, vol. 43, no. 1, pp. 50 – 67, Jan. 1947.

[77] M. Movahhedi, A. Abdipour, A. Nentchev, M. Dehghan, and S. Selberherr, "Alternating direction implicit formulation of the finite-element time-domain method," *IEEE Transactions on Microwave Theory and Techniques*, vol. 55, no. 6, pp. 1322 – 1331, Jun. 2007.

[78] B. Tatian, "Fitting refractive-index data with the Sellmeier dispersion formula," *Applied Optics*, vol. 23, no. 24, pp. 4477 – 4485, Dec. 1984.

[79] E. Kreyszig, *Advanced Engineering Mathematics*. John Wiley and Sons, 1979.

[80] T. J. Ypma, "Historical development of the Newton-Raphson method," *SIAM Review*, vol. 37, no. 4, pp. 531 – 551, Dec. 1995.

[81] N. M. Newmark, "A method of computation for structural dynamics," *Journal of the Engineering Mechanics Division*, vol. 85, no. 3, pp. 67 – 94, Jul. 1959.

[82] MathWorks, "Matlab R2018a," https://www.mathworks.com/products/matlab.html, Accessed: Mar. 28, 2020.

[83] J.-M. Liu, *Photonic Devices*. Cambridge, UK: Cambridge University Press, 2005.

[84] A. Dubietis, A. Couairon, and G. Genty, "Supercontinuum generation: Introduction," *Journal of the Optical Society of America: B*, vol. 36, no. 2, Feb. 2019.

[85] J. M. Dudley and J. R. Taylor, *Supercontinuum Generation in Optical Fibers*. Cambridge, UK: Cambridge University Press, 2010.

[86] I. H. Malitson, "Interspecimen comparison of the refractive index of fused silica," *Journal of the Optical Society of America*, vol. 55, no. 10, pp. 1205 – 1209, Oct. 1965.

[87] V. S. Ryanben'kii and S. V. Tsynkov, *A Theoretical Introduction to Numerical Analysis*. Chapman & Hall/CRC, 2007.

[88] B. I. Epureanu and H. S. Greenside, "Fractal basins of attraction associated with a damped Newton's method," *SIAM Review*, vol. 40, no. 1, pp. 102 – 109, Mar. 1998.

[89] G. Strang, *Linear Algebra and Its Applications*, 4th ed. Thomson, Brooks/Cole, 2006.

[90] L. N. Trefethen and D. B. III, *Numerical Linear Algebra*. Philadelphia, PA: Society for Industrial and Applied Mathematics, 1997.

[91] P. P. Silvester and R. L. Ferrari, *Finite Elements for Electrical Engineers*, 3rd ed. Cambridge, UK: Cambridge University Press, 1996.

[92] D. A. Dunavant, "High degree efficient symmetrical Gaussian quadrature rules for the triangle," *International Journal for Numerical Methods in Engineering*, vol. 21, pp. 1129 – 1148, Jun. 1985.

[93] J. W. Thomas, *Numerical Partial Differential Equations: Finite Difference Methods*. Berlin, Germany: Springer Science and Business Media, 1995.

[94] J. C. Strikwerda, *Finite Difference Schemes and Partial Differential Equations*, 2nd ed. Philadelphial, PA: Society for Industrial and Applied Mathematics, 2004.

[95] P. D. Lax and R. D. Richtmyer, "Survey of the stability of linear finite difference equations," *Communications on Pure and Applied Mathematics*, vol. 9, pp. 267 – 293, 1956.

[96] S. D. Gedney and U. Navsariwala, "An unconditionally stable finite element time-domain solution of the vector wave equation," *IEEE Microwave and Guided Wave Letters*, vol. 5, no. 10, pp. 332 – 334, Oct. 1995.

[97] M. Ortiz, "A note on energy conservation and stability of nonlinear time-stepping algorithms," *Computers and Structures*, vol. 24, no. 1, pp. 167 – 168, 1986.

[98] R. A. Horn and C. R. Johnson, *Matrix Analysis*, 2nd ed. Cambridge, UK: Cambridge University Press, 1985.

[99] T. G. Moore, J. G. Blaschak, A. Taflove, and G. A. Kriegsmann, "Theory and application of radiation boundary operators," *IEEE Transactions on Antennas and Propagation*, vol. 36, no. 12, pp. 1797 – 1812, Dec. 1988.

[100] S. H. Schot, "Eighty years of Sommerfeld's radiation condition," *Historia Mathematica*, vol. 19, no. 4, pp. 385 – 401, Nov. 1992.

[101] H. Li and X. Wu, "Local absorbing boundary conditions for nonlinear wave equation on unbounded domain," *Physical Review E*, vol. 84, no. 3, Feb. 2011.

[102] J.-P. Berenger, "A perfectly matched layer for the absorption of electromagnetic waves," *Journal of Computational Physics*, vol. 114, no. 2, pp. 185 – 200, Oct. 1994.

[103] Z. S. Sacks, D. M. Kingsland, R. Lee, and J.-F. Lee, "A perfectly matched anisotropic absorber for use as an absorbing boundary condition," *IEEE Transactions on Antennas and Propagation*, vol. 43, no. 12, pp. 1460 – 1463, Dec. 1995.

[104] S. D. Gedney, "An anisotropic perfectly matched layer-absorbing medium for the truncation of FDTD lattices," *IEEE Transactions on Antennas and Propagation*, vol. 44, no. 12, pp. 1630 – 1639, Dec. 1996.

[105] F. L. Teixeira and W. C. Chew, "General closed-form PML constitutive tensors to match arbitrary bianisotropic and dispersive linear media," *IEEE Microwave and Guided Wave Letters*, vol. 8, no. 6, pp. 223 – 225, Jun. 1998.

[106] S. D. Gedney, "An anisotropic PML absorbing media for the FDTD simulation of fields in lossy and dispersive media," *Electromagnetics*, vol. 16, no. 4, pp. 399 – 415, 1996.

[107] C. M. Rappaport, "Perfectly matched absorbing boundary conditions based on anisotropic lossy mapping of space," *IEEE Microwave and Guided Wave Letters*, vol. 5, no. 3, pp. 90 – 92, Mar. 1995.

[108] A. Giannopoulos, "Unsplit implementation of higher order PMLs," *IEEE Transactions on Antennas and Propagation*, vol. 60, no. 3, pp. 1479 – 1485, Mar. 2012.

[109] F. L. Teixeira and W. C. Chew, "Complex space approach to perfectly matched layers: A review and some new developments," *International Journal of Numerical Modelling: Electronic Networks, Devices, and Fields*, vol. 13, no. 5, pp. 441 – 455, Sep. 2000.

[110] F. L. Teixeira and W. C. Chew, "Systematic derivation of anisotropic PML absorbing media in cylindrical and spherical coordinates," *IEEE Microwave and Guided Wave Letters*, vol. 7, no. 11, pp. 371 – 373, Nov. 1997.

[111] A. P. Zhao, J. Juntunen, and A. V. Raisanen, "Generalized material-independent PML absorbers for the FDTD simulation of electromagnetic waves in arbitrary anisotropic dielectric and magnetic media," *IEEE Microwave and Guided Wave Letters*, vol. 8, no. 2, pp. 52 – 54, Feb. 1998.

[112] A. P. Zhao, "Application of the material-independent PML absorbers to the FDTD analysis of electromagnetic waves in nonlinear media," *Microwave and Optical Technology Letters*, vol. 17, no. 3, pp. 164 – 168, Feb. 1998.

[113] M. Fujii and P. Russer, "A nonlinear and dispersive APML ABC for the FD-TD methods," *IEEE Microwave and Wireless Components Letters*, vol. 12, no. 11, Nov. 2002.

[114] J. Li, L. Mao, J. Miao, C. Guo, and X. Wang, "Efficient PML implementation for truncating nonlinear FDTD lattices based on the z-transform method," in *2009 International Conference on Microwave Technology and Computational Electromagnetics (ICMTCE 2009)*, Nov. 2009, pp. 375 – 378.

[115] O. Ramadan, "Efficient PML formulations for truncating nonlinear FDTD domains," in *Workshops on Computational Electromagnetics in Time-Domain (CEM-TD)*, 2005.

[116] S. G. Johnson, "Notes on perfectly matched layers (PMLs)," https://math.mit.edu/~stevenj/18.369/pml.pdf, Mar. 2010, accessed: Nov. 5, 2019.

[117] G. Agrawal, *Nonlinear Fiber Optics*, 5th ed. Cambridge, MA: Elsevier Academic Press, 2013.

[118] A. F. Oskooi, L. Zhang, Y. Avniel, and S. G. Johnson, "The failure of perfectly matched layers and towards their redemption by adiabatic absorbers," *Optics Express*, vol. 16, no. 15, Jul. 2008.

[119] D. B. Kirk and W. Hwu, *Programming Massively Parallel Processors: A Hands on Approach*, 3rd ed. Burlington, MA: Morgan Kaufmann Publishers, 2017.

[120] *CUDA C++ Programming Guide*, NVIDIA Corporation, Nov. 2019, PG-02829-001_v10.2.

[121] "Nvidia high performance computing," https://developer.nvidia.com/hpc, accessed: Mar. 27, 2020.

[122] "Khronos group - OpenCL overview," https://www.khronos.org/opencl/, accessed: Mar. 27, 2020.

[123] W. W. Hwu, *GPU Computing Gems*, ser. Applications of GPU Computing. Burlington, MA: Morgan Kaufmann Publishers, 2012, Jade Edition.

[124] W. W. Hwu, *GPU Computing Gems*, ser. Applications of GPU Computing. Burlington, MA: Morgan Kaufmann Publishers, 2011, Emerald Edition.

[125] D. S. Abraham and D. D. Giannacopoulos, "Dispersive Möbius transform finite-element time-domain method on graphics processing units," *IEEE Transactions on Magnetics*, vol. 52, no. 3, Mar. 2016.

[126] D. A. Knoll and D. E. Keyes, "Jacobian-free Newton-Krylov methods: A survey of approaches and applications," *Journal of Computational Physics*, vol. 193, no. 2, Jan. 2004.

[127] D. S. Abraham and D. D. Giannacopoulos, "A parallel finite-element time-domain method for nonlinear dispersive media," *IEEE Transactions on Magnetics*, vol. 56, no. 2, Feb. 2020.

[128] Y. V. Khalevitsky, N. V. Burmasheva, and A. V. Konovalov, "An approach to the parallel assembly of the stiffness matrix in elastoplastic problems," in *AIP Conference Proceedings*, 2016.

[129] E. Scholz, H. Ye, S. Schöps, and M. Clemens, "A parallel FEM matrix assembly for electro-quasistatic problems on GPGPU systems," *IEEE Transactions on Magnetics*, vol. 49, no. 5, May 2013.

[130] U. Kiran, D. Sharma, and S. S. Gautam, "GPU-warp based finite element matrices generation and assembly using coloring method," *Journal of Computational Design and Engineering*, vol. 6, no. 4, Oct. 2019.

[131] S. C. Rennich, D. Stosic, and T. A. Davis, "Accelerating sparse Cholesky factorization on GPUs," *Parallel Computing*, vol. 59, Nov. 2016.

[132] R. Helfenstein and J. Koko, "Parallel preconditioned conjugate gradient algorithm on GPU," *Journal of Computational and Applied Mathematics*, vol. 236, no. 15, Sep. 2012.

[133] K. He, S. X.-D. Tan, H. Zhao, X.-X. Liu, H. Wang, and G. Shi, "Parallel GMRES solver for fast analysis of large linear dynamic systems on GPU platforms," *Integration*, vol. 52, Jan. 2016.

[134] Y. El-Kurdi, W. J. Gross, and D. Giannacopoulos, "Parallel multigrid acceleration for the finite-element Gaussian belief propagation algorithm," *IEEE Transactions on Magnetics*, vol. 50, no. 2, Feb. 2014.

[135] Y. El-Kurdi, "Parallel finite element processing using Gaussian belief propagation inference on probabilistic graphical models," Ph.D. dissertation, McGill University, 2015.

[136] Y. El-Kurdi, W. J. Gross, and D. Giannacopoulos, "Efficient implementation of Gaussian belief propagation solver for large sparse diagonally dominant linear systems," *IEEE Transactions on Magnetics*, vol. 48, no. 2, Feb. 2012.

[137] D. Fernández, A. Akbarzadeh-Sharbaf, and D. D. Giannacopoulos, "Solving finite-element time-domain problems with GaBP," *IEEE Transactions on Magnetics*, vol. 53, no. 6, Jun. 2017.

[138] G. Guennebaud and B. Jacob, "Eigen v3.3.5," http://eigen.tuxfamily.org, Accessed: Mar. 28, 2020.

[139] Microsoft, "Acquiring high-resolution time stamps," https://docs.microsoft.com/en-us/windows/win32/sysinfo/acquiring-high-resolution-time-stamps, Accessed: Mar. 28, 2020.

[140] Y. Saad, *Iterative Methods for Sparse Linear Systems*, 2nd ed. Philadelphia, PA: Society for Industrial and Applied Mathematics, 2003.

[141] A. Ern and J.-L. Guermond, *Theory and Practice of Finite Elements*. New York, NY: Springer, 2004.

[142] S. D. Gedney, C. Luo, B. Guernsey, J. A. Roden, R. Crawford, and J. A. Miller, "The discontinuous Galerkin finite element time domain method (DGFETD): A high order, globally-explicit method for parallel computation," in *2007 IEEE International Symposium on Electromagnetic Compatibility*, 2007.

[143] A. Akbari and D. D. Giannacopoulos, "A parallel Newton-Raphson algorithm for strong coupling modeling of multi-physics problems," in *2019 International Compumag Conference*, 2019.

[144] R. C. Buck and E. F. Buck, *Advanced Calculus*, 3rd ed. New York, NY: McGraw-Hill Inc., 1978.

[145] J. Stewart, *Single Variable Essential Calculus: Early Transcendentals*, 1st ed. Belmont, CA: Thomson Brooks/Cole, 2007.