Title
A fourth-order adaptive mesh refinement solver for Maxwell's Equations

Author
Chilton, Sven

Publication Date
2013

Peer reviewed|Thesis/dissertation

A fourth-order adaptive mesh refinement solver for Maxwell's Equations

by

Sven Chilton

A dissertation submitted in partial satisfaction of the
requirements for the degree of
Doctor of Philosophy

in

Engineering – Nuclear Engineering

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Phillip Colella, Co-chair
Professor Edward Morse, Co-chair
Professor Karl van Bibber
Professor Jonathan Wurtele

Fall 2013

Abstract

A fourth-order adaptive mesh refinement solver for Maxwell's Equations

by

Sven Chilton

Doctor of Philosophy in Engineering – Nuclear Engineering

University of California, Berkeley

Professor Phillip Colella, Co-chair

Professor Edward Morse, Co-chair

We present a fourth-order accurate, multilevel Maxwell solver, discretized in space with a finite volume approach and advanced in time with the classical fourth-order Runge Kutta method (RK4). Electric fields are decomposed into divergence-free and curl-free parts; we solve for the divergence-free parts of Faraday's Law and the Ampère-Maxwell Law while imposing Gauss' Laws as initial conditions. We employ a damping scheme inspired by the Advanced Weather Research and Forecasting Model to eliminate non-physical waves reflected off of coarse-fine grid boundaries, and Kreiss-Oliger artificial dissipation to remove standing wave instabilities. Surprisingly, artificial dissipation appears to damp the spuriously reflected waves at least as effectively as the atmospheric community's damping scheme.

För kärlek, i framtiden
Vox sola harmoniam eget

Contents

List of Figures

List of Tables

Acknowledgments

I am indebted to every member of the Applied Numerical Algorithms Group at Lawrence Berkeley National Laboratory, both past and present, with whom I interacted during this project. I owe the following people in particular thanks, though my list is by no means exhaustive:

First and foremost, Phil Colella, for taking a chance on me after I left my previous research group, and for his tireless mentorship and guidance over the intervening years;

Brian Van Straalen, whose extensive knowledge of algorithms and C++ coding practices helped me untangle myself from plenty of tricky situations;

Jeff Johnson, for his astounding grasp of both plasma physics and programming, and his ability to explain challenging concepts;

Peter McCorquodale, whose work in fourth-order conservation equation advance schemes formed the basis of my program to advance the Faraday-Ampère-Maxwell system;

Qinghai Zhang, who first showed me how to use gdb to debug C++ code;

Dan Graves, for helping me through many debugging sessions and offering a supportive ear;

Chris Chaplin, for making my last years of grad school less torturous with gallows humor, heartfelt conversations, and delectable beer.

I would also like to thank the following people outside of ANAG:

Jean-Luc Vay, for many helpful discussions on electromagnetism in general and on (A)PML methods in particular;

Karl van Bibber, who graciously joined my committee at the last minute, read my dissertation efficiently, and still offered excellent feedback;

Ed Morse, Jonathan Wurtele, and Stuart Bale, for gracing me with their vast knowledge of plasma physics;

Steve Lund and John Verboncoeur, both of whom were instrumental in starting my grad school career;

And finally, my friends and family (both too numerous to list here), without whose support and patience I would never have made it through.

Chapter 1

Introduction

1.1 Wave Reflection Problem

Multigrid and adaptive grid schemes are efficient techniques for solving systems of differential equations numerically. The benefits of such schemes manifest themselves prominently in systems with multiple, vastly different length scales, for example. Despite the complexity often required to implement such algorithms, resolving only certain regions of the problem domain finely saves significant computer time and processor power over covering the entire problem domain with a fine grid [Berger and Colella, 1989].

However, complications arise with propagating wave solutions to systems lacking significant dissipation, e.g. the simple wave equation, Maxwell's Equations and extensions thereof. In such systems, a wave traveling from a finely resolved region into a coarsely resolved region will not be transmitted correctly across the refinement boundary. Part of the wave is reflected from the coarse-fine grid boundary and trapped inside the fine region. This is the numerical analog of an impedance mismatch between two dielectrics.

The waves reflected from the coarse-fine grid boundaries are not physical. However, they can interfere with physical effects. This interference limits the degree to which grid size can be varied, which in turn limits the benefits of a multilevel scheme.

Analyzing and testing methods of overcoming these non-physical reflections and their effects comprised a large portion of this dissertation's background work. Ultimately, two schemes are employed to mitigate the reflection problem: damping and artificial dissipation. In damping schemes, one constructs a so-called sponge layer adjacent to the fine side of a coarse-fine grid boundary, in which one adds a term which induces exponential decay of the difference between a given variable at the current level of resolution and the next lower resolution to the right hand side (RHS) of the PDE one is advancing in time. Artificial dissipation schemes, in turn, entail adding appropriately-scaled, higher-derivative terms to the RHS, not merely inside a sponge layer, but over the entire problem domain.

The damping scheme implemented in this dissertation draws inspiration from two sources: a very similar damping scheme developed for atmospheric and climate modeling and (Asym-

metric) Perfectly Matched Layers ((A)PMLs). Damping in climate modeling problems is discussed in Sec. 1.2. While (A)PMLs also mitigate problems associated with non-physical reflections, they are used in a rather different context from those outlined in Sec. 1.2 and in this thesis, as Sec. 1.3 explains.

1.2 Damping in Climate Modeling

Designing appropriate radiation BCs for numerical simulations of wave equations is a long-standing issue, which is further complicated in simulations with nested grids at differing resolutions. Since wave equations occur frequently in climate modeling problems, the atmospheric modeling community has developed the Weather Research and Forecasting (WRF) Model and, more recently, the Advanced Weather Research and Forecasting Model (WRF-ARW) to address these difficulties [Davies, 1976, Skamarock et al., 2005, Harris and Durran, 2010]. WRF-ARW provides a damping scheme to minimize reflections of outgoing waves from finely resolved regions off of coarse-fine grid boundaries, and distortion of incoming waves from coarsely resolved regions.

Inside a so-called sponge layer adjacent to a refinement boundary, on the coarse side, damping terms are added to the RHS of each time evolution equation in the given wave system. One damping term is linearly proportional to the difference between fine and coarse quantities. The other, which is not implemented in this project, is linearly proportional to a numerical approximation of the Laplacian of that difference.

1.3 (Asymmetric) Perfectly Matched Layers

The Perfectly Matched Layer (PML) [Berenger, 1994, Berenger, 1996] and Asymmetric Perfectly Matched Layer (APML) [Vay, 2000, Vay, 2001, Vay, 2002] are useful techniques for solving Maxwell's Equations over an infinite domain numerically. In both cases, the system is closed via the insertion of an artificial layer outside the domain of interest wherein damping terms are added to the time derivatives. In APML implementations, damping terms are added to the spatial derivatives as well. PML methods thus form a subset of APML methods. The damping terms are chosen such that there are no non-physical reflections of EM field components back into the domain of interest.

There are two crucial differences between (A)PML methods and the damping scheme used in the present work. One, in (A)PMLs, the damping term associated with each field component is proportional to the entire field component, not the difference in that component between two successive levels of resolution. Two, (A)PMLs are placed outside the (truncated) problem domain, rather than inside finely resolved grids.

1.4 Artificial Dissipation

In testing our Maxwell solver on a problem with a divergence-free current source and all field components initially equal to 0, we observed a non-physical, standing wave phenomenon in the center of the problem domain (also the center of the current source) even with damping terms added. Thus, we added the well-known Kreiss-Oliger artificial dissipation term [Kreiss and Oliger, 1972, Kreiss and Oliger, 1973] to the Faraday-Ampère-Maxwell system, i.e. the subset of Maxwell's Equations being advanced in time. This term is added globally, not just in the sponge layer, and on all levels. Kreiss and Oliger added their dissipation terms to maintain stability in simulations of propagating waves with discontinuities, but the logic behind adding them in this context is similar.

[Harris and Durran, 2010] describe a fourth-derivative artificial dissipation term, which the authors obtained by analogy to the sixth-derivative scheme of [Knievel et al., 2007], who in turn cite [Xue, 2000] for proposing the general scheme. In both [Knievel et al., 2007] and [Xue, 2000], the dissipation terms are added as corrections to the flux, i.e. the first-order spatial derivative terms of the equations being advanced in time. Curiously, we did not observe the Gibbs overshoot mentioned in [Knievel et al., 2007].

For a more detailed discussion on artificial dissipation, in particular on choosing appropriate scaling coefficients, see Sec. 5.4.

1.5 Adaptive Mesh Refinement

Block-structured adaptive mesh refinement (AMR) is a powerful tool for computing numerical solutions to partial differential equations involving multiple length scales and strongly localized effects [Vay et al., 2002, Vay et al., 2004a, Vay et al., 2004b, Samtaney et al., 2004]. In the simplest form of AMR (to which we will limit ourselves in this document) a rectangular problem domain is first divided into evenly sized cells. Regions requiring greater resolution (according to a user-specified criterion) are identified and covered in a disjoint union of rectangles, which are then refined by an integer factor. The refinement procedure is applied recursively until a maximum number of levels is reached or an accuracy criterion is met. Fig. 1.1 depicts a typical hierarchy of AMR levels.

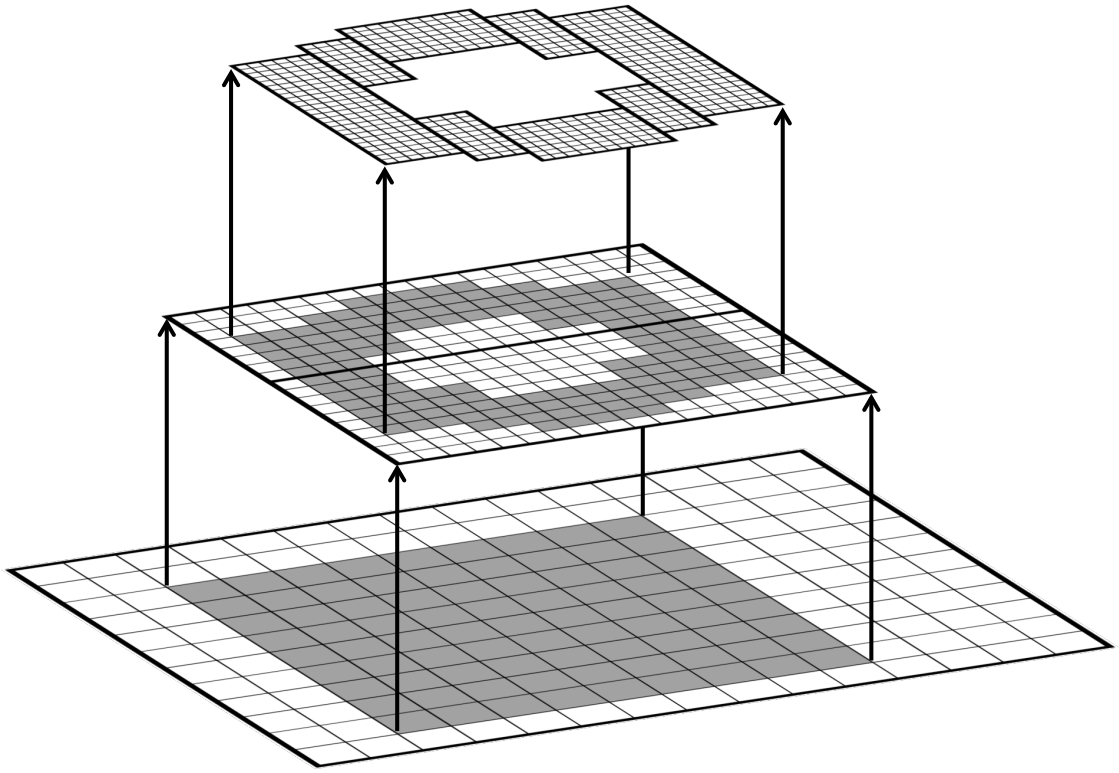

Figure 1.1: A typical hierarchy of AMR levels. In this case, there is a refinement ratio of 2 between each coarse level and the next finer level.

For time-dependent problems, we refine the time spacing as well as the spatial grid to use the same CFL number (wave propagation speed times time step divided by spatial grid spacing) at all levels of resolution. In this case, we typically advance the equation via subcycling in time (explained in Sec. 3.2) as opposed to advancing the equation on all refinement levels simultaneously.

We discretize the spatial domain using a finite volume approach. Each quantity for which we solve is approximated by its cell average, which in a given cell is the integral of the quantity over the cell in question divided by the cell volume. These volume integrals are typically converted to integrals over the cells' boundary faces through the divergence theorem and then approximated by quadratures [McCorquodale and Colella, 2011, McCorquodale, 2012]. We employ standard Cartesian finite difference approximations of spatial derivatives. We will not consider more complicated problem domains involving embedded boundaries (which cut across rectangular cells) or mapped grids (non-rectangular problem domains and cells mapped to rectangular problem domains and cells) in this dissertation.

The work in progress described in this thesis represents the first 4th-order, finite volume, AMR scheme for Maxwell's Equations. Most Maxwell solvers employ Yee-type schemes with staggered grids [Yee, 1966]. For example, if the magnetic fields in a Yee scheme are face-

centered, the electric fields are node-centered. Staggering has the advantage of ensuring that **B** and **E** are manifestly divergence-free in a vacuum. However, Yee schemes are in general 2nd-order accurate, and like many other types of spatial discretization, they lose an order of accuracy at boundaries of non-uniform grids [Monk and Süli, 1994]. Moreover, the author knows of no successful efforts to modify a Yee scheme to allow for 4th-order accuracy and AMR capabilities.

Chapter 2

Maxwell's Equations

2.1 Description of equations

Maxwell's Equations form the foundation of classical electromagnetism. They consist of a coupled system describing the evolution of electric and magnetic fields subject to charge densities, current densities, and boundary conditions. They can be expressed in differential or integral form, in terms of total or free charge and current, in SI or cgs units. The last distinction arises because charge is defined differently in cgs than SI units. Unless explicitly stated otherwise, all equations related to Maxwell's Equations in this document are represented in SI units. Likewise, we will not consider the integral forms of Maxwell's Equations.

In terms of total charge and current, Maxwell's Equations can be expressed as

$$\begin{aligned} \frac{\partial \mathbf{B}}{\partial t} &= -\boldsymbol{\nabla} \times \mathbf{E}, \\ \frac{\partial \mathbf{E}}{\partial t} &= c^2 \boldsymbol{\nabla} \times \mathbf{B} - \frac{\mathbf{J}}{\epsilon_0}, \\ \boldsymbol{\nabla} \cdot \mathbf{E} &= \frac{\rho}{\epsilon_0}, \\ \boldsymbol{\nabla} \cdot \mathbf{B} &= 0. \end{aligned} \tag{2.1}$$

When solving the first two of Eqs. (2.1) numerically on a refinement level $l > 0$, we replace the time derivatives as indicated in the damping scheme below:

$$\frac{\partial F^l}{\partial t} \to \frac{\partial F^l}{\partial t} + \sigma \left(F^l - I(F^{l-1}) \right). \tag{2.2}$$

The damping coefficient σ has units of inverse time. Associated numerical stability concerns and final choice of σ are discussed in Sec. 2.6 and Sec. 3.3, respectively. F stands for a component of either the magnetic field $\mathbf{B}$ or the electric field $\mathbf{E}$. In turn, ρ and $\mathbf{J}$ are the total electric charge and current densities, respectively. ϵ_0 is the permittivity of free space,

and $c = (\mu_0 \epsilon_0)^{-1/2}$ is the speed of light in a vacuum. μ_0, in turn, is the permeability of free space. Numerical values of these free space quantities are given in Table 2.1.

Table 2.1: Values of free space quantities in Maxwell's Equations.

Symbol	Value	Units
c	2.9979×10^8	m s^{-1}
ϵ_0	8.8542×10^{-12}	F m^{-1} = C^2 s^2 kg^{-1} m^{-3}
μ_0	$4\pi \times 10^{-7}$	H m^{-1} = kg m C^{-2}

The top line of Eqs. (2.1) is Faraday's Law, which states that time-varying magnetic fields yield electric fields. The second line, the Ampère-Maxwell Law, shows that magnetic fields arise from electric currents and so-called displacement currents, i.e. time-varying electric fields. The equations in Eqs. (2.1) involving divergences are Gauss' Laws of Electricity and Magnetism, respectively. The former states that electric charges yield electric fields, while the latter implies that magnetic monopoles do not exist.

In terms of free charges and currents, Maxwell's Equations can be expressed as

$$\begin{aligned} \frac{\partial \mathbf{B}}{\partial t} &= -\boldsymbol{\nabla} \times \mathbf{E}, \\ \frac{\partial \mathbf{D}}{\partial t} &= \boldsymbol{\nabla} \times \mathbf{H} - \mathbf{J}_f, \\ \boldsymbol{\nabla} \cdot \mathbf{D} &= \rho_f, \\ \boldsymbol{\nabla} \cdot \mathbf{B} &= 0. \end{aligned} \tag{2.3}$$

The subscript f stands for free. $\mathbf{D}$ is the electric displacement field, given by $\mathbf{D} = \epsilon \mathbf{E}$, where ϵ is the permittivity of the medium in which the equations are to be solved. Similarly, the magnetizing field $\mathbf{H}$ is given by $\mathbf{H} = \mathbf{B}/\mu$, where μ is the permeability of the medium. Note that if the medium is a vacuum, $\epsilon = \epsilon_0$, $\mu = \mu_0$, and Eqs. (2.3) are equivalent to Eqs. (2.1).

2.2 Helmholtz Decomposition

Helmholtz's Theorem guarantees that, as long as the necessary partial derivatives exist, any vector field $\mathbf{V}$ can be decomposed as follows:

$$\begin{aligned} \mathbf{V} &= \mathbf{V}_P + \mathbf{V}_T, \\ \mathbf{V}_P &= \boldsymbol{\nabla} \Delta^{-1} \boldsymbol{\nabla} \cdot \mathbf{V}, \\ &\equiv \mathbb{Q}(\mathbf{V}) \\ \mathbf{V}_T &= \left(\mathbb{I} - \boldsymbol{\nabla} \Delta^{-1} \boldsymbol{\nabla} \right) \cdot \mathbf{V}, \\ &\equiv \mathbb{P}(\mathbf{V}), \\ \boldsymbol{\nabla} \times \mathbf{V}_P &= \mathbf{0}, \\ \boldsymbol{\nabla} \cdot \mathbf{V}_T &= 0. \end{aligned} \tag{2.4}$$

Δ is the Laplacian operator, given in a Cartesian space of dimension $\mathbf{D}$ by

$$\Delta \equiv \sum_{d=0}^{\mathbf{D}-1} \frac{\partial^2}{\partial x_d^2}. \tag{2.5}$$

Formulas for the Laplacian in curvilinear coordinates are given in [Arfken et al., 2005]. Δ^{-1} is the inverse Laplacian, defined such that if $\Delta F = f$, $F = \Delta^{-1} f$. The form of $\Delta^{-1} f$ depends on the number of spatial dimensions $\mathbf{D}$ in the problem [Evans, 2010]. For $\mathbf{D} = 2$,

$$F(\boldsymbol{x}) = \Delta^{-1} f(\boldsymbol{x}) = \frac{1}{2\pi} \int f(\boldsymbol{x}') \ln |\boldsymbol{x} - \boldsymbol{x}'| \, d^2 x', \tag{2.6}$$

while for $\mathbf{D} = 3$,

$$F(\boldsymbol{x}) = \Delta^{-1} f(\boldsymbol{x}) = -\frac{1}{4\pi} \int \frac{f(\boldsymbol{x}')}{|\boldsymbol{x} - \boldsymbol{x}'|} \, d^3 x'. \tag{2.7}$$

Since $\mathbf{B}$ is divergence-free by Gauss' Law of Magnetism, we see that the Helmholtz decomposition of Maxwell's Equations can be written as

$$\begin{aligned} \frac{\partial \mathbf{B}}{\partial t} &= -\boldsymbol{\nabla} \times \mathbf{E}_T, \\ \frac{\partial \mathbf{E}_T}{\partial t} &= c^2 \boldsymbol{\nabla} \times \mathbf{B} - \frac{\mathbb{P}(\mathbf{J})}{\epsilon_0}, \\ \boldsymbol{\nabla} \cdot \mathbf{E}_P &= \frac{\rho}{\epsilon_0}, \\ \boldsymbol{\nabla} \cdot \mathbf{B} &= 0. \end{aligned} \tag{2.8}$$

The transverse electric field and current density are expressed with different (though still equivalent) notation because we assume that $\mathbf{J}$ is specified as a whole, while we are free to solve for $\mathbf{E}_T$ and $\mathbf{E}_P$ separately.

This decomposition of the electric fields in Maxwell's Equations allows us to recast the Faraday and Ampère-Maxwell Laws as two second-order equations in $\mathbf{B}$ and $\mathbf{E}_T$, namely

$$\frac{\partial^2 \mathbf{B}}{\partial t^2} = c^2 \Delta \mathbf{B} + \frac{1}{\epsilon_0} \boldsymbol{\nabla} \times \mathbf{J} \tag{2.9}$$

and

$$\frac{\partial^2 \mathbf{E}_T}{\partial t^2} = c^2 \Delta \mathbf{E}_T - \frac{1}{\epsilon_0} \frac{\partial \mathbb{P}(\mathbf{J})}{\partial t}. \tag{2.10}$$

In the limit of no current, both Eqs. (2.9) and (2.10) reduce to basic wave equations. Thus, we expect that our damping algorithm can be applied successfully to Maxwell's Equations.

2.3 Continuous Damped Maxwell Equations

Substituting Eq. (2.2) into Eqs. (2.8) yields the (continuous) damped Helmholtz- decomposed Maxwell Equations on refinement level $l > 0$:

$$\begin{aligned}
\frac{\partial \mathbf{B}^l}{\partial t} &= -\boldsymbol{\nabla} \times \mathbf{E}_T^l + \sigma \left(\mathbf{B}^l - I(\mathbf{B}^{l-1}) \right), \\
\frac{\partial \mathbf{E}_T^l}{\partial t} &= c^2 \boldsymbol{\nabla} \times \mathbf{B}^l - \frac{\mathbb{P}(\mathbf{J}^l)}{\epsilon_0} + \sigma \left(\mathbf{E}_T^l - I(\mathbf{E}_T^{l-1}) \right), \\
\boldsymbol{\nabla} \cdot \mathbf{E}_P^l &= \frac{\rho^l}{\epsilon_0}, \\
\boldsymbol{\nabla} \cdot \mathbf{B}^l &= 0.
\end{aligned} \tag{2.11}$$

Recall that for $l = 0$, the damping terms are not present. The charge and current densities are related by the continuity equation

$$\frac{\partial \rho^l}{\partial t} + \boldsymbol{\nabla} \cdot \mathbf{J}^l = 0, \tag{2.12}$$

and, by definition,

$$\boldsymbol{\nabla} \cdot \mathbf{E}_T^l = 0. \tag{2.13}$$

For the remainder of this document, we will concern ourselves primarily with the Faraday-Ampère-Maxwell (FAM) system, i.e. the top two equations in Eqs. (2.11). In solving this system approximately (or exactly, when possible) we will assume that $\mathbb{P}(\mathbf{J}^l)$ is known and the divergence-free conditions on the magnetic and transverse electric fields are imposed as initial conditions.

2.4 Faraday-Ampère-Maxwell System

We recast Faraday's Law and the Ampère-Maxwell Law into a system of the form

$$\frac{\partial U^l}{\partial t} + \nabla \cdot F(U^l) = S^l - \sigma \left(U^l - I(U^{l-1}) \right). \tag{2.14}$$

Here,

$$U^l = \begin{pmatrix} \mathbf{B}^l \\ \mathbf{E}_T^l \end{pmatrix}, \tag{2.15}$$

$$F(U^l) = \begin{pmatrix} 0 & -E_{Tz}^l & E_{Ty}^l & 0 & c^2 B_z^l & -c^2 B_y^l \\ E_{Tz}^l & 0 & -E_{Tx}^l & -c^2 B_z^l & 0 & c^2 B_x^l \\ -E_{Ty}^l & E_{Tx}^l & 0 & c^2 B_y^l & -c^2 B_x^l & 0 \end{pmatrix}, \tag{2.16}$$

and

$$S^l = \begin{pmatrix} \mathbf{0} \\ -\dfrac{\mathbb{P}(\mathbf{J}^l)}{\epsilon_0} \end{pmatrix} \tag{2.17}$$

In Eqs. (2.14) – (2.17), the l superscript denotes data at the current level, while $l-1$ denotes data at the next coarser level, if it exists. The same will apply in all future equations in this document containing terms with such superscripts.

We define the divergence of a two-dimensional array F as

$$(\nabla \cdot F)_i = \frac{\partial F_{ji}}{\partial x_j}, \tag{2.18}$$

employing the Einstein summation convention. Eq (2.18) allows us to express the curl of an arbitrary vector field $\mathbf{V}$ as the divergence of a two-dimensional array T, as shown below:

$$\begin{aligned} \boldsymbol{\nabla} \times \mathbf{V} &= \epsilon_{ijk} \partial_j V_k \boldsymbol{e}_i \\ &= \partial_j (\epsilon_{ijk} V_k) \boldsymbol{e}_i \\ &= \partial_j (-\epsilon_{jik} V_k) \boldsymbol{e}_i \\ &= \partial_j T_{ji} \boldsymbol{e}_i \\ &= \boldsymbol{\nabla} \cdot T, \end{aligned} \tag{2.19}$$

with T defined as

$$T = \begin{pmatrix} 0 & -V_z & V_y \\ V_z & 0 & -V_x \\ -V_y & V_x & 0 \end{pmatrix}. \tag{2.20}$$

Before discussing the FAM system further, we will develop the mathematical machinery necessary to solve and analyze the system discretely.

2.5 Fourth-Order Discretization

2.5.1 Finite Volume Basics

Our discretization, advancement scheme, and interpolation algorithms are based heavily on the work described in [McCorquodale and Colella, 2011]. However, since our flux array F is a linear function of our field component array U (a consequence of the linearity of Maxwell's Equations) and the variables in our U are both conserved and primitive, we make many simplifications, which are noted accordingly.

Each level in our problem domain is divided into rectangular control volumes. In physical space, a control volume $V_{\boldsymbol{i}}$ is given by

$$V_{\boldsymbol{i}} = [\boldsymbol{i}h, (\boldsymbol{i} + \boldsymbol{u})h] \quad \text{for } \boldsymbol{i} \in \mathbb{Z}^{\mathbf{D}},\ \boldsymbol{u} = (1, ..., 1), \tag{2.21}$$

where $h = \Delta x = \Delta y = \Delta z$ is the grid spacing. For the remainder of this document, h, Δx, etc. will be used interchangeably. $\boldsymbol{i}$ is the control volume's cell index. The bracket notation in Eq. (2.21) denotes the lower left and upper right corners of the control volume in physical space, respectively. Elsewhere, this same notation is employed to denote the indices of the cells at the lower left and upper right corners of grids and problem domains.

We discretize U in space by taking its average over a control volume, i.e.

$$\langle U \rangle_{\boldsymbol{i}}(t) = \frac{1}{h^{\mathbf{D}}} \int_{V_{\boldsymbol{i}}} U(\boldsymbol{x}, t)\, d^{\mathbf{D}} x. \tag{2.22}$$

One may specify the initial conditions of $\langle U \rangle_{\boldsymbol{i}}$ either by an exact evaluation of Eq. (2.22) or a quadrature, which must be at least fourth-order accurate for the code to achieve fourth-order accuracy. Note that $\langle U \rangle_{\boldsymbol{i}}(t)$ is also an approximation of U at the *center* of cell $\boldsymbol{i}$ (a.k.a. control volume $V_{\boldsymbol{i}}$), i.e.

$$\langle U \rangle_{\boldsymbol{i}}(t) = U\left((\boldsymbol{i} + \tfrac{1}{2}\boldsymbol{u})h, t\right) + \mathcal{O}(h^2) \tag{2.23}$$

We compute the evolution of the spatially discretized system by a method-of-lines approach:

$$\begin{aligned} \frac{d\langle U \rangle_{\boldsymbol{i}}}{dt} &= \frac{1}{h^{\mathbf{D}}} \int_{V_{\boldsymbol{i}}} (S - \nabla \cdot F)\, d^{\mathbf{D}} x \\ &= \langle S \rangle_{\boldsymbol{i}} - \frac{1}{h} \sum_d \left(\langle F^d \rangle_{\boldsymbol{i} + \frac{1}{2}\boldsymbol{e}_d} - \langle F^d \rangle_{\boldsymbol{i} - \frac{1}{2}\boldsymbol{e}_d} \right) \end{aligned} \tag{2.24}$$

$$\langle F^d \rangle_{\boldsymbol{i} \pm \frac{1}{2}\boldsymbol{e}_d} = \frac{1}{h^{\mathbf{D}-1}} \int_{A_d^{\pm}} F^d d^{\mathbf{D}-1} x, \tag{2.25}$$

where $A_d^{\pm}$ are the high and low faces bounding $V_{\boldsymbol{i}}$, with normals pointing in the $\boldsymbol{e}_d$ direction. Note that $\langle F^d \rangle_{\boldsymbol{i} \pm \frac{1}{2}\boldsymbol{e}_d}$ is also an approximation of F at the center of the high or low d-face of $V_{\boldsymbol{i}}$, i.e.

$$\langle F^d \rangle_{\boldsymbol{i} \pm \frac{1}{2}\boldsymbol{e}_d} = F^d\left(U((\boldsymbol{i} + \tfrac{1}{2}\boldsymbol{u} \pm \tfrac{1}{2}\boldsymbol{e}_d)h, t)\right) + \mathcal{O}(h^2) \tag{2.26}$$

2.5.2 Spatial Discretization

[McCorquodale and Colella, 2011, Sec. 2.3] describes the spatial discretization of a general finite volume scheme for advancing hyperbolic equations. However, since our fluxes are linear and our system makes no distinction between conserved and primitive variables, our procedure for computing approximate face averages and approximations to derivatives is considerably simpler. Our fourth-order accurate approximation to face averages of U is given by

$$\langle U \rangle^d_{\boldsymbol{i} + \frac{1}{2}\boldsymbol{e}_d} = \frac{7}{12}\left(\langle U \rangle_{\boldsymbol{i} + \boldsymbol{e}_d} - \langle U \rangle_{\boldsymbol{i}} \right) - \frac{1}{12}\left(\langle U \rangle_{\boldsymbol{i} + 2\boldsymbol{e}_d} - \langle U \rangle_{\boldsymbol{i} - \boldsymbol{e}_d} \right). \tag{2.27}$$

Since, in general,

$$h \left\langle \frac{\partial U}{\partial x_d} \right\rangle_{\boldsymbol{i}} = \langle U \rangle^d_{\boldsymbol{i} + \frac{1}{2}\boldsymbol{e}_d} - \langle U \rangle^d_{\boldsymbol{i} - \frac{1}{2}\boldsymbol{e}_d}, \tag{2.28}$$

Eq. (2.27) is consistent with the fourth-order approximation

$$h \left\langle \frac{\partial U}{\partial x_d} \right\rangle_{\boldsymbol{i}} = \frac{8}{12}\left(\langle U \rangle_{\boldsymbol{i} + \boldsymbol{e}_d} - \langle U \rangle_{\boldsymbol{i} - \boldsymbol{e}_d} \right) - \frac{1}{12}\left(\langle U \rangle_{\boldsymbol{i} + 2\boldsymbol{e}_d} - \langle U \rangle_{\boldsymbol{i} - 2\boldsymbol{e}_d} \right) \tag{2.29}$$

Because F is linear,

$$\langle F^d \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}_d} = F^d\left(\langle U \rangle^d_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}_d}\right) \tag{2.30}$$

Eq. (2.28), in turn, suggests the following formula for computing (fourth-order accurate approximations to) divergences:

$$(\nabla \cdot F)_{\boldsymbol{i}} \to (D \cdot F)_{\boldsymbol{i}} = \frac{1}{h}\sum_d \left(\langle F^d \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}_d} - \langle F^d \rangle_{\boldsymbol{i}-\frac{1}{2}\boldsymbol{e}_d}\right) \tag{2.31}$$

2.5.3 Temporal Discretization

On a single level, we advance Eq. (2.24) via the standard RK4 scheme. We approximate it as

$$\begin{aligned} \frac{d\langle U \rangle_{\boldsymbol{i}}}{dt} &= \langle S \rangle_{\boldsymbol{i}} - (D \cdot F)_{\boldsymbol{i}} \\ &= \langle S \rangle_{\boldsymbol{i}} - \frac{1}{h}\sum_d \left(\langle F^d \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}_d} - \langle F^d \rangle_{\boldsymbol{i}-\frac{1}{2}\boldsymbol{e}_d}\right), \end{aligned} \tag{2.32}$$

Given $\langle U \rangle^n \approx \langle U \rangle(t^n)$, we compute $\langle U \rangle^{n+1} \approx \langle U \rangle(t^n + \Delta t)$ as follows:

$$\langle U \rangle(t^n + \Delta t) = \langle U \rangle(t^n) + \tfrac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) + \mathcal{O}(\Delta t)^5, \tag{2.33}$$

with

$$\langle U \rangle^{(0)} = \langle U \rangle(t^n) \qquad k_1 = \left(\langle S \rangle(t^n) - D \cdot F(\langle U \rangle^{(0)})\right)\Delta t \tag{2.34}$$

$$\langle U \rangle^{(1)} = \langle U \rangle^{(0)} + \tfrac{1}{2}k_1 \qquad k_2 = \left(\langle S \rangle(t^n + \tfrac{1}{2}\Delta t) - D \cdot F(\langle U \rangle^{(1)})\right)\Delta t \tag{2.35}$$

$$\langle U \rangle^{(2)} = \langle U \rangle^{(0)} + \tfrac{1}{2}k_2 \qquad k_3 = \left(\langle S \rangle(t^n + \tfrac{1}{2}\Delta t) - D \cdot F(\langle U \rangle^{(2)})\right)\Delta t \tag{2.36}$$

$$\langle U \rangle^{(3)} = \langle U \rangle^{(0)} + k_3 \qquad k_4 = \left(\langle S \rangle(t^n + \Delta t) - D \cdot F(\langle U \rangle^{(3)})\right)\Delta t \tag{2.37}$$

2.5.4 Interpolation in time

When filling in ghost cells outside a grid in level $l > 0$, or in computing the damping term in Eq. (2.14) (which only applies inside the sponge layer described in Sec. 3.4) we must interpolate $\langle U \rangle^{l-1}(t^{l-1})$ to $\langle U \rangle^{l-1}(t^l)$, with $t^l \in [t^{l-1}, t^{l-1} + \Delta t^{l-1}]$.

In [Fok and Rosales, 2008], the authors derive the following formula for interpolating $\langle U \rangle^{l-1}$ to times between t^{l-1} and $t^{l-1} + \Delta t^{l-1}$:

$$\begin{aligned} \langle U \rangle^{l-1}(t^{l-1} + \chi\Delta t^{l-1}) = \langle U \rangle^{(0)} &+ \chi k_1 + \frac{\chi^2}{2}(-3k_1 + 2k_2 + 2k_3 - k_4) \\ &+ \frac{2\chi^3}{3}(k_1 - k_2 - k_3 + k_4) + \mathcal{O}((\Delta t^{l-1})^4) \end{aligned} \tag{2.38}$$

with $\chi \in [0, 1]$, $\langle U \rangle^{(0)} = \langle U \rangle^{l-1}(t^{l-1})$, and each k_i defined as in Eqs. (2.34) – (2.37). Given $n^{l-1}_{\text{ref}} \equiv \Delta t^{l-1}/\Delta t^l$, we must interpolate to each time $t^l = t^{l+1} + s\Delta t^l$, with $s = 0, 1, ..., n^{l-1}_{\text{ref}} -$

$1 \equiv \chi n_{\text{ref}}^l$. At each of these times, we need to find fourth-order approximations to each $\langle U \rangle^{(i)}$, as defined in Eqs. (2.34) – (2.37).

Through a complicated derivation [McCorquodale and Colella, 2011,McCorquodale, 2012] we find

$$\begin{aligned}
\langle U \rangle^{(0)} =& U(t^n + s\Delta t^l) \\
=& \langle U \rangle^{(0)} + \left(\frac{s}{n_{\text{ref}}^{l-1}} \right) k_1 \\
&+ \frac{1}{2} \left(\frac{s^2}{(n_{\text{ref}}^{l-1})^2} \right) (-3k_1 + 2k_2 + 2k_3 - k_4) \\
&+ \frac{2}{3} \left(\frac{s^3}{(n_{\text{ref}}^{l-1})^3} \right) (k_1 - k_2 - k_3 + k_4) + \mathcal{O}((\Delta t^l)^4). \qquad (2.39) \\
\langle U \rangle^{(1)} =& \langle U \rangle^{(0)} + \left(\frac{s}{n_{\text{ref}}^{l-1}} + \frac{1}{2n_{\text{ref}}^{l-1}} \right) k_1 \\
&+ \frac{1}{2} \left(\frac{s^2}{(n_{\text{ref}}^{l-1})^2} + \frac{s}{(n_{\text{ref}}^{l-1})^2} \right) (-3k_1 + 2k_2 + 2k_3 - k_4) \\
&+ \frac{2}{3} \left(\frac{s^3}{(n_{\text{ref}}^{l-1})^3} + \frac{3s^2}{2(n_{\text{ref}}^{l-1})^3} \right) (k_1 - k_2 - k_3 + k_4) + \mathcal{O}((\Delta t^l)^4); \qquad (2.40) \\
\langle U \rangle^{(2)} =& \langle U \rangle^{(0)} + \left(\frac{s}{n_{\text{ref}}^{l-1}} + \frac{1}{2n_{\text{ref}}^{l-1}} \right) k_1 \\
&+ \frac{1}{2} \left(\frac{s^2}{(n_{\text{ref}}^{l-1})^2} + \frac{s}{(n_{\text{ref}}^{l-1})^2} + \frac{1}{2(n_{\text{ref}}^{l-1})^2} \right) (-3k_1 + 2k_2 + 2k_3 - k_4) \\
&+ \frac{2}{3} \left(\frac{s^3}{(n_{\text{ref}}^{l-1})^3} + \frac{3s^2}{2(n_{\text{ref}}^{l-1})^3} + \frac{3s}{2(n_{\text{ref}}^{l-1})^3} + \frac{3}{8(n_{\text{ref}}^{l-1})^3} \right) (k_1 - k_2 - k_3 + k_4) \\
&+ \frac{1}{4(n_{\text{ref}}^{l-1})^2} (k_2 - k_3) + \mathcal{O}((\Delta t^l)^4); \qquad (2.41) \\
\langle U \rangle^{(3)} =& \langle U \rangle^{(0)} + \left(\frac{s}{n_{\text{ref}}^{l-1}} + \frac{1}{n_{\text{ref}}^{l-1}} \right) k_1 \\
&+ \frac{1}{2} \left(\frac{s^2}{(n_{\text{ref}}^{l-1})^2} + \frac{2s}{(n_{\text{ref}}^{l-1})^2} + \frac{1}{(n_{\text{ref}}^{l-1})^2} \right) (-3k_1 + 2k_2 + 2k_3 - k_4) \\
&+ \frac{2}{3} \left(\frac{s^3}{(n_{\text{ref}}^{l-1})^3} + \frac{3s^2}{(n_{\text{ref}}^{l-1})^3} + \frac{3s}{(n_{\text{ref}}^{l-1})^3} + \frac{3}{4(n_{\text{ref}}^{l-1})^3} \right) (k_1 - k_2 - k_3 + k_4) \\
&+ \frac{1}{2(n_{\text{ref}}^{l-1})^2} (k_3 - k_2) + \mathcal{O}((\Delta t^l)^4). \qquad (2.42)
\end{aligned}$$

2.5.5 Interpolation in space

Consider the ghost cell layer just outside and the sponge layer just inside each grid in level $l > 0$. For each component u of our array of conserved variables U, we fill in ghost cells and compute damping terms by interpolating $\langle u \rangle^{l-1}$ (averages over coarse cells) to $\langle u \rangle^{l}$ (averages over fine cells) with cubic Taylor polynomials. We assume that $\langle u \rangle^{l-1}$ has already been interpolated to the correct time, if necessary.

For each coarse cell indexed by $\boldsymbol{i} \in \mathbb{Z}^{\mathbf{D}}$, we use the following notations [McCorquodale and Colella, 2011, Sec. 3.2]:

- $\mathcal{F}(\boldsymbol{i})$ is the set of fine cells contained within $\boldsymbol{i}$.
- $a_{\boldsymbol{i},\boldsymbol{p}}$ (for $\boldsymbol{p} \in \mathbb{N}^{\mathbf{D}}$ such that $\|\boldsymbol{p}\|_1 = \sum_d |p_d| \leq 3$) are the coefficients that will be used for interpolation to $\langle u \rangle^{l}_{\boldsymbol{k}}$ for all $\boldsymbol{k} \in \mathcal{F}(\boldsymbol{i})$. These will be the coefficients of the Taylor polynomial of degree 3 for u around the center of cell $\boldsymbol{i}$. The number of coefficients for each coarse cell in 2D is 10, and in 3D is 20. The coefficients will be computed from values of $\langle u \rangle^{l-1}$.
- $\mathcal{N}(\boldsymbol{i})$ is the set of coarse cells used as a stencil from which to take $\langle u \rangle^{l-1}$ in order to find the coefficients $a_{\boldsymbol{i},\boldsymbol{p}}$. See [McCorquodale and Colella, 2011, Sec. 3.2.2] for a more detailed description.

For $\boldsymbol{z} \in \mathbb{R}^{\mathbf{D}}$ and $\boldsymbol{p} \in \mathbb{N}^{\mathbf{D}}$, we write $\langle \boldsymbol{z}^{\boldsymbol{p}} \rangle^{l-1}_{\boldsymbol{j}}$ or $\langle \boldsymbol{z}^{\boldsymbol{p}} \rangle^{l}_{\boldsymbol{k}}$ to denote the average, respectively over coarse cell $\boldsymbol{j}$ or fine cell $\boldsymbol{k}$, of

$$\boldsymbol{z}^{\boldsymbol{p}} = \prod_d (z_d^{p_d} - K(p_d)), \tag{2.43}$$

where

$$K(q) = \begin{cases} \dfrac{2^{-q}}{q+1} & \text{if } q > 0 \text{ and } q \text{ is even;} \\ 0 & \text{otherwise.} \end{cases} \tag{2.44}$$

This constant is included to simplify numerical calculations; the average of $\boldsymbol{z}^{\boldsymbol{p}}$ on the cube $[-\frac{1}{2}, \frac{1}{2}]^{\mathbf{D}}$ is 1 if $\boldsymbol{p} = \boldsymbol{0}$, and 0 otherwise.

To obtain the coefficients $a_{\boldsymbol{i},\boldsymbol{p}}$ for coarse cell $\boldsymbol{i}$, we solve a constrained linear least-squares problem [Golub and Van Loan, 1996, pgs. 585–586] for the overdetermined system

$$\sum_{\substack{\boldsymbol{p} \in \mathbb{N}^{\mathbf{D}} \\ \|\boldsymbol{p}\|_1 \leq 3}} a_{\boldsymbol{i},\boldsymbol{p}} \langle (\boldsymbol{x} - \boldsymbol{x}_{\boldsymbol{i}})^{\boldsymbol{p}} \rangle^{l-1}_{\boldsymbol{j}} = \langle u \rangle^{l-1}_{\boldsymbol{j}} \quad \text{for all } \boldsymbol{j} \in \mathcal{N}(\boldsymbol{i}) - \{\boldsymbol{i}\}, \tag{2.45}$$

where $\mathcal{N}(\boldsymbol{i}) - \{\boldsymbol{i}\}$ is the set of all coarse cells in the stencil *except* for $\boldsymbol{i}$ itself. Eq. (2.45) is subject to the conservation constraint

$$\sum_{\substack{\boldsymbol{p} \in \mathbb{N}^{\mathbf{D}} \\ \|\boldsymbol{p}\|_1 \leq 3}} a_{\boldsymbol{i},\boldsymbol{p}} \langle (\boldsymbol{x} - \boldsymbol{x}_{\boldsymbol{i}})^{\boldsymbol{p}} \rangle^{l-1}_{\boldsymbol{i}} = \langle u \rangle^{l-1}_{\boldsymbol{i}} \tag{2.46}$$

where $\boldsymbol{x_i}$ is the center of cell $\boldsymbol{i}$. We then use the coefficients $a_{\boldsymbol{i},\boldsymbol{p}}$ to interpolate for each fine cell $\boldsymbol{k} \in \mathcal{F}(\boldsymbol{i})$:

$$\langle u \rangle_{\boldsymbol{k}}^{l} = \sum_{\substack{\boldsymbol{p} \in \mathbb{N}^{\mathbf{D}} \\ \|\boldsymbol{p}\|_1 \leq 3}} a_{\boldsymbol{i},\boldsymbol{p}} \langle (\boldsymbol{x} - \boldsymbol{x_i})^{\boldsymbol{p}} \rangle_{\boldsymbol{k}}^{l}. \tag{2.47}$$

The conservation constraint Eq. (2.46) follows from the general definition of averaging down:

$$\frac{1}{(n_{\text{ref}}^{l-1})^{\mathbf{D}}} \sum_{\boldsymbol{k} \in \mathcal{F}(\boldsymbol{i})} \langle u \rangle_{\boldsymbol{k}}^{l} = \langle u \rangle_{\boldsymbol{i}}^{l-1}. \tag{2.48}$$

This is true whether one computes the average as defined in Eq. (2.22) or Eq. (2.43). Substituting Eq. (2.47) into Eq. (2.48) yields

$$\frac{1}{(n_{\text{ref}}^{l-1})^{\mathbf{D}}} \sum_{\boldsymbol{k} \in \mathcal{F}(\boldsymbol{i})} \sum_{\substack{\boldsymbol{p} \in \mathbb{N}^{\mathbf{D}} \\ \|\boldsymbol{p}\|_1 \leq 3}} a_{\boldsymbol{i},\boldsymbol{p}} \langle (\boldsymbol{x} - \boldsymbol{x_i})^{\boldsymbol{p}} \rangle_{\boldsymbol{k}}^{l} = \langle u \rangle_{\boldsymbol{i}}^{l-1}. \tag{2.49}$$

But it also follows from Eq. (2.48) that for each $\boldsymbol{p}$,

$$\frac{1}{(n_{\text{ref}}^{l-1})^{\mathbf{D}}} \sum_{\boldsymbol{k} \in \mathcal{F}(\boldsymbol{i})} \langle (\boldsymbol{x} - \boldsymbol{x_i})^{\boldsymbol{p}} \rangle_{\boldsymbol{k}}^{l} = \langle (\boldsymbol{x} - \boldsymbol{x_i})^{\boldsymbol{p}} \rangle_{\boldsymbol{i}}^{l-1} \tag{2.50}$$

For $\mathbf{D} = 2$, Eq. (2.45) contains 10 variables (i.e. the $a_{\boldsymbol{i},\boldsymbol{p}}$ coefficients) and 11 or 12 equations. For $\mathbf{D} = 3$, we have 20 variables and 30 to 32 equations. Converting these constrained least-squares problems to unconstrained ones is described in [McCorquodale, 2012, pgs. 23–24], which, in turn, relies on formulas from [Golub and Van Loan, 1996].

2.6 Stability Analysis of the Faraday-Ampère-Maxwell System

2.6.1 Source-Free Faraday-Ampère-Maxwell System

For simplicity, let us first consider the undamped, source-free Maxwell Equations. These can be expressed as

$$\begin{aligned} \frac{\partial \mathbf{B}}{\partial t} &= -\boldsymbol{\nabla} \times \mathbf{E}, \\ \frac{\partial \mathbf{E}}{\partial t} &= c^2 \boldsymbol{\nabla} \times \mathbf{B}, \\ \boldsymbol{\nabla} \cdot \mathbf{E} &= 0, \\ \boldsymbol{\nabla} \cdot \mathbf{B} &= 0. \end{aligned} \tag{2.51}$$

In this case, $\mathbf{E} = \mathbf{E}_T$ and $\mathbf{E}_P = \mathbf{0}$, since $\boldsymbol{\nabla} \cdot \mathbf{E}_T = 0$ by definition, while $\boldsymbol{\nabla} \cdot \mathbf{E}_P = 0$ by virtue of an absence of charge density.

Equivalently, the undamped, source-free Faraday-Ampère-Maxwell (FAM) system may be written as

$$\frac{\partial U^l}{\partial t} + \nabla \cdot F(U^l) = 0, \tag{2.52}$$

or

$$\frac{\partial}{\partial t}\begin{pmatrix} \mathbf{B} \\ \mathbf{E} \end{pmatrix} + A_x \cdot \frac{\partial}{\partial x}\begin{pmatrix} \mathbf{B} \\ \mathbf{E} \end{pmatrix} + A_y \cdot \frac{\partial}{\partial y}\begin{pmatrix} \mathbf{B} \\ \mathbf{E} \end{pmatrix} + A_z \cdot \frac{\partial}{\partial z}\begin{pmatrix} \mathbf{B} \\ \mathbf{E} \end{pmatrix} = 0, \tag{2.53}$$

where

$$A_x = \begin{pmatrix} & & & 0 & 0 & 0 \\ & 0 & & 0 & 0 & -1 \\ & & & 0 & 1 & 0 \\ 0 & 0 & 0 & & & \\ 0 & 0 & c^2 & & 0 & \\ 0 & -c^2 & 0 & & & \end{pmatrix}, \tag{2.54}$$

$$A_y = \begin{pmatrix} & & & 0 & 0 & 1 \\ & 0 & & 0 & 0 & 0 \\ & & & -1 & 0 & 0 \\ 0 & 0 & -c^2 & & & \\ 0 & 0 & 0 & & 0 & \\ c^2 & 0 & 0 & & & \end{pmatrix}, \tag{2.55}$$

and

$$A_z = \begin{pmatrix} & & & 0 & -1 & 0 \\ & 0 & & 1 & 0 & 0 \\ & & & 0 & 0 & 0 \\ 0 & c^2 & 0 & & & \\ -c^2 & 0 & 0 & & 0 & \\ 0 & 0 & 0 & & & \end{pmatrix}. \tag{2.56}$$

The eigenvalues of each A_i matrix are $-c$, 0 and c, each with a multiplicity of 2. Each A_i has an associated matrix R_i with each column given by a right eigenvector of A_i. Conventionally, the columns are arranged such that the leftmost column in R_i is associated with the lowest eigenvalue in A_i, the second leftmost column is associated with the second lowest eigenvalue, and so on. Thus,

$$R_x = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ c^{-1} & 0 & 0 & 0 & -c^{-1} & 0 \\ 0 & -c^{-1} & 0 & 0 & 0 & c^{-1} \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}, \tag{2.57}$$

$$R_y = \begin{pmatrix} 0 & -c^{-1} & 0 & 0 & 0 & c^{-1} \\ 0 & 0 & 1 & 0 & 0 & 0 \\ c^{-1} & 0 & 0 & 0 & -c^{-1} & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \end{pmatrix}, \tag{2.58}$$

and

$$R_z = \begin{pmatrix} c^{-1} & 0 & 0 & 0 & -c^{-1} & 0 \\ 0 & -c^{-1} & 0 & 0 & 0 & c^{-1} \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}. \tag{2.59}$$

Likewise, each A_i has an associated matrix L_i, defined as the inverse of R_i. Each row in L_i is a left eigenvector of A_i. Following the convention by which the R_i matrices are constructed, the top row in L_i is associated with the lowest eigenvalue of A_i, the second row in is associated with the second lowest eigenvalue, and so on. Thus,

$$L_x = \begin{pmatrix} 0 & c/2 & 0 & 0 & 0 & 1/2 \\ 0 & 0 & -c/2 & 0 & 1/2 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & -c/2 & 0 & 0 & 0 & 1/2 \\ 0 & 0 & c/2 & 0 & 1/2 & 0 \end{pmatrix}, \tag{2.60}$$

$$L_y = \begin{pmatrix} 0 & 0 & c/2 & 1/2 & 0 & 0 \\ -c/2 & 0 & 0 & 0 & 0 & 1/2 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & -c/2 & 1/2 & 0 & 0 \\ c/2 & 0 & 0 & 0 & 0 & 1/2 \end{pmatrix}, \tag{2.61}$$

and

$$L_z = \begin{pmatrix} c/2 & 0 & 0 & 0 & 1/2 & 0 \\ 0 & -c/2 & 0 & 1/2 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ -c/2 & 0 & 0 & 0 & 1/2 & 0 \\ 0 & c/2 & 0 & 1/2 & 0 & 0 \end{pmatrix}. \tag{2.62}$$

For any A_i, R_i, and L_i (not just those defined for the Faraday-Ampère-Maxwell system), $L_i \cdot A_i \cdot R_i = \mathrm{diag}(\lambda_{i,m})$, where $\lambda_{i,m} \leq \lambda_{i,m+1} \; \forall \; m \in [1, \dim U - 1]$. Thus, for the Faraday-Ampère-Maxwell system,

$$L_i \cdot A_i \cdot R_i = \mathrm{diag}(-c, -c, 0, 0, c, c). \tag{2.63}$$

2.6.2 Damped, Source-Free Faraday-Ampère-Maxwell System

Now we consider systems of the form

$$\frac{\partial U^l}{\partial t} + \nabla \cdot F(U^l) = -\sigma \left(U^l - I(U^{l-1}) \right). \tag{2.64}$$

For Maxwell's Equations, the terms in Eq. (2.64) are as defined at the beginning of Sec. 2.4.

To assist in the analysis of the damped, source-free Maxwell system, we define

$$\mathcal{U} \equiv U^l - I(U^{l-1}) \tag{2.65}$$

for each component of U. Since F as defined in Eq. (2.16) is a linear operator of U, we have

$$\frac{\partial \mathcal{U}}{\partial t} + \nabla \cdot F(\mathcal{U}) = -\sigma \mathcal{U}, \tag{2.66}$$

which can also be expressed as

$$\frac{\partial}{\partial t} \begin{pmatrix} \mathcal{B}_x \\ \mathcal{B}_y \\ \mathcal{B}_z \\ \mathcal{E}_x \\ \mathcal{E}_y \\ \mathcal{E}_z \end{pmatrix} + \nabla \cdot \begin{pmatrix} 0 & -\mathcal{E}_z & \mathcal{E}_y & 0 & c^2\mathcal{B}_z & -c^2\mathcal{B}_y \\ \mathcal{E}_z & 0 & -\mathcal{E}_x & -c^2\mathcal{B}_z & 0 & c^2\mathcal{B}_x \\ -\mathcal{E}_y & \mathcal{E}_x & 0 & c^2\mathcal{B}_y & -c^2\mathcal{B}_x & 0 \end{pmatrix} = -\sigma \begin{pmatrix} \mathcal{B}_x \\ \mathcal{B}_y \\ \mathcal{B}_z \\ \mathcal{E}_x \\ \mathcal{E}_y \\ \mathcal{E}_z \end{pmatrix}, \tag{2.67}$$

or equivalently,

$$\begin{aligned} \frac{\partial}{\partial t} \begin{pmatrix} \mathcal{B}_x \\ \mathcal{B}_y \\ \mathcal{B}_z \\ \mathcal{E}_x \\ \mathcal{E}_y \\ \mathcal{E}_z \end{pmatrix} = &- \begin{pmatrix} & & & 0 & 0 & 0 \\ & \mathbf{0} & & 0 & 0 & -1 \\ & & & 0 & 1 & 0 \\ 0 & 0 & 0 & & & \\ 0 & 0 & c^2 & & \mathbf{0} & \\ 0 & -c^2 & 0 & & & \end{pmatrix} \cdot \frac{\partial}{\partial x} \begin{pmatrix} \mathcal{B}_x \\ \mathcal{B}_y \\ \mathcal{B}_z \\ \mathcal{E}_x \\ \mathcal{E}_y \\ \mathcal{E}_z \end{pmatrix} \\ &- \begin{pmatrix} & & & 0 & 0 & 1 \\ & \mathbf{0} & & 0 & 0 & 0 \\ & & & -1 & 0 & 0 \\ 0 & 0 & -c^2 & & & \\ 0 & 0 & 0 & & \mathbf{0} & \\ c^2 & 0 & 0 & & & \end{pmatrix} \cdot \frac{\partial}{\partial y} \begin{pmatrix} \mathcal{B}_x \\ \mathcal{B}_y \\ \mathcal{B}_z \\ \mathcal{E}_x \\ \mathcal{E}_y \\ \mathcal{E}_z \end{pmatrix} \\ &- \begin{pmatrix} & & & 0 & -1 & 0 \\ & \mathbf{0} & & 1 & 0 & 0 \\ & & & 0 & 0 & 0 \\ 0 & c^2 & 0 & & & \\ -c^2 & 0 & 0 & & \mathbf{0} & \\ 0 & 0 & 0 & & & \end{pmatrix} \cdot \frac{\partial}{\partial z} \begin{pmatrix} \mathcal{B}_x \\ \mathcal{B}_y \\ \mathcal{B}_z \\ \mathcal{E}_x \\ \mathcal{E}_y \\ \mathcal{E}_z \end{pmatrix} - \sigma \begin{pmatrix} \mathcal{B}_x \\ \mathcal{B}_y \\ \mathcal{B}_z \\ \mathcal{E}_x \\ \mathcal{E}_y \\ \mathcal{E}_z \end{pmatrix} \end{aligned} \tag{2.68}$$

Now, we express $\mathcal{U}$ in terms of discrete Fourier transforms (DFTs) as

$$\mathcal{U}_{(j_x,j_y,j_z)} = \sum_{k_x,k_y,k_z} \hat{\mathcal{U}}_{(k_x,k_y,k_z)} e^{2\pi i k_x j_x \Delta x/L_x} e^{2\pi i k_y j_y \Delta y/L_y} e^{2\pi i k_z j_z \Delta z/L_z}. \tag{2.69}$$

Assuming periodic boundary conditions, and using the fourth-order stencils for first derivatives described in Eq. (2.29), we have

$$\begin{aligned}
\left.\frac{\partial \mathcal{U}}{\partial x}\right|_{(j_x,j_y,j_z)} &= \sum_{k_x,k_y,k_z} \frac{\hat{\mathcal{U}}_{(k_x,k_y,k_z)}}{\Delta x} \left[\frac{8}{12}\left(e^{2\pi i k_x (j_x+1)\Delta x/L_x} - e^{2\pi i k_x (j_x-1)\Delta x/L_x}\right)\right. \\
&\qquad \left. -\frac{1}{12}\left(e^{2\pi i k_x (j_x+2)\Delta x/L_x} - e^{2\pi i k_x (j_x-2)\Delta x/L_x}\right)\right] \\
&\qquad \times e^{2\pi i k_y j_y \Delta y/L_y} e^{2\pi i k_z j_z \Delta z/L_z} \\
&= \sum_{k_x,k_y,k_z} \hat{\mathcal{U}}_{(k_x,k_y,k_z)} e^{2\pi i k_x j_x \Delta x/L_x} e^{2\pi i k_y j_y \Delta y/L_y} e^{2\pi i k_z j_z \Delta z/L_z} \\
&\qquad \times \frac{1}{\Delta x}\left[\frac{8}{12}\left(e^{2\pi i k_x \Delta x/L_x} - e^{-2\pi i k_x \Delta x/L_x}\right)\right. \\
&\qquad \left. -\frac{1}{12}\left(e^{4\pi i k_x \Delta x/L_x} - e^{-4\pi i k_x \Delta x/L_x}\right)\right] \\
&= \sum_{k_x,k_y,k_z} \hat{\mathcal{U}}_{(k_x,k_y,k_z)} e^{2\pi i k_x j_x \Delta x/L_x} e^{2\pi i k_y j_y \Delta y/L_y} e^{2\pi i k_z j_z \Delta z/L_z} \\
&\qquad \times \frac{i}{6\Delta x}\left[8\sin\left(\frac{2\pi k_x \Delta x}{L_x}\right) - \sin\left(\frac{4\pi k_x \Delta x}{L_x}\right)\right] \\
&= \sum_{k_x,k_y,k_z} i\Omega_x \hat{\mathcal{U}}_{(k_x,k_y,k_z)} e^{2\pi i k_x j_x \Delta x/L_x} e^{2\pi i k_y j_y \Delta y/L_y} e^{2\pi i k_z j_z \Delta z/L_z},
\end{aligned} \tag{2.70}$$

with

$$\Omega_x \equiv \frac{1}{6\Delta x}\left[8\sin\left(\frac{2\pi k_x \Delta x}{L_x}\right) - \sin\left(\frac{4\pi k_x \Delta x}{L_x}\right)\right]. \tag{2.71}$$

If we define Ω_y and Ω_z similarly, the DFTs of $\frac{\partial \mathcal{U}}{\partial y}$ and $\frac{\partial \mathcal{U}}{\partial z}$ are analogous to Eq. (2.70).

Substituting Eq. (2.70) into Eq. (2.68), we find that at each Fourier node (k_x, k_y, k_z), we must have

$$\frac{d}{dt}\begin{pmatrix} \hat{\mathcal{B}}_x \\ \hat{\mathcal{B}}_y \\ \hat{\mathcal{B}}_z \\ \hat{\mathcal{E}}_x \\ \hat{\mathcal{E}}_y \\ \hat{\mathcal{E}}_z \end{pmatrix} = -\begin{pmatrix} & \sigma\mathbb{I} & & 0 & -i\Omega_z & i\Omega_y \\ & & & i\Omega_z & 0 & -i\Omega_x \\ & & & -i\Omega_y & i\Omega_x & 0 \\ 0 & ic^2\Omega_z & -ic^2\Omega_y & & & \\ -ic^2\Omega_z & 0 & ic^2\Omega_x & & \sigma\mathbb{I} & \\ ic^2\Omega_y & -ic^2\Omega_x & 0 & & & \end{pmatrix} \cdot \begin{pmatrix} \hat{\mathcal{B}}_x \\ \hat{\mathcal{B}}_y \\ \hat{\mathcal{B}}_z \\ \hat{\mathcal{E}}_x \\ \hat{\mathcal{E}}_y \\ \hat{\mathcal{E}}_z \end{pmatrix}. \tag{2.72}$$

With the help of a computer algebra system (or through exceedingly tedious algebra by hand) we see that the matrix on the right side of Eq. (2.72) has the following eigenvalues, each with multiplicity 2: $-\sigma - ic\sqrt{\Omega_x^2 + \Omega_y^2 + \Omega_z^2},\ -\sigma,\ -\sigma + ic\sqrt{\Omega_x^2 + \Omega_y^2 + \Omega_z^2}$.

We see that Eq. (2.72) is of the form

$$\frac{dU}{dt} = M \cdot U, \tag{2.73}$$

where U and M are generic 1- and 2-D (square) arrays, respectively. If the components of M are constant in time, the exact solution to Eq. (2.73) is given by

$$U(t) = \exp(Mt) \cdot U(0), \tag{2.74}$$

or, if M is diagonalizable (which ours is),

$$U(t) = V \cdot \exp(\Lambda t) \cdot V^{-1} \cdot U(0), \tag{2.75}$$

where V is a matrix of right eigenvectors of M and Λ is a diagonal matrix of eigenvalues of M. Proving that Eq. (2.75) follows from Eq. (2.74) for diagonalizable matrices is a reader exercise in [Iserles, 1996, pg. 304]. It follows readily from the definition of the matrix exponential

$$\exp(M) \equiv I + \sum_{n=1}^{\infty} \frac{M^n}{n!} \tag{2.76}$$

and the fact that for any diagonalizable matrix $M = V\Lambda V^{-1}$ and integer $n \geq 1$,

$$\begin{aligned} M^n &= (V\Lambda V^{-1})(V\Lambda V^{-1}) \cdots (V\Lambda V^{-1}) \\ &= (V\Lambda V^{-1} V\Lambda V^{-1}) \cdots (V\Lambda V^{-1}) \\ &= (V\Lambda^2 V^{-1}) \cdots (V\Lambda V^{-1}) \\ &= V\Lambda^n V^{-1} \end{aligned} \tag{2.77}$$

If we advance Eq. (2.73) in time with a Runge-Kutta method, we find that U at the nth time step is approximately given by

$$U(n\Delta t) \approx U^n = V \cdot R(\Lambda\Delta t)^n \cdot V^{-1} \cdot U(0) \tag{2.78}$$

[Iserles, 1996, pg. 60], where

$$R(\Lambda\Delta t) = \begin{pmatrix} R(\lambda_1 \Delta t) & & 0 \\ & \ddots & \\ 0 & & R(\lambda_n \Delta t) \end{pmatrix}, \tag{2.79}$$

and $R(z)$ is the stability function [Hairer and Wanner, 1996, pg. 40] of the Runge-Kutta method used to advance Eq. (2.73). It is given by

$$R(z) = 1 + zb^T \cdot (I - zA)^{-1} \cdot e. \tag{2.80}$$

Here, z is the complex variable $x + iy$, the T superscript denotes a transposition, $e = (1, ..., 1)^T$, with $\dim e$ equal to the number of stages of the Runge-Kutta advance, and I is an identity matrix. A and b^T refer to the Butcher diagram of the advance [Butcher, 1964, Hairer et al., 1993], given by

$$\begin{array}{c|c} c & A \\ \hline & b^T \end{array} . \tag{2.81}$$

In the case of the RK4 advance, we have

$$\begin{aligned} A &= \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1/2 & 0 & 0 & 0 \\ 0 & 1/2 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \\ b^T &= (1/6, 2/6, 2/6, 1/6) \\ c &= (0, 1/2, 1/2, 1)^T. \end{aligned} \tag{2.82}$$

Straightforward algebra shows that for the RK4 advance,

$$R(z) = 1 + z + \frac{z^2}{2} + \frac{z^3}{6} + \frac{z^4}{24}. \tag{2.83}$$

For a general Runge-Kutta advance to be stable, we must have $|R(\lambda_m \Delta t)| < 1 \ \forall \ m \in [1, \dim U]$. For our problem, $-\sigma \Delta t \pm ic\sqrt{\Omega_x^2 + \Omega_y^2 + \Omega_z^2}\Delta t$ must lie in the region of absolute stability (RAS), the region in the complex plane in which $|R(z)| < 1$. Each Runge-Kutta scheme has an associated RAS. The RAS for RK4 is depicted in the figure below:

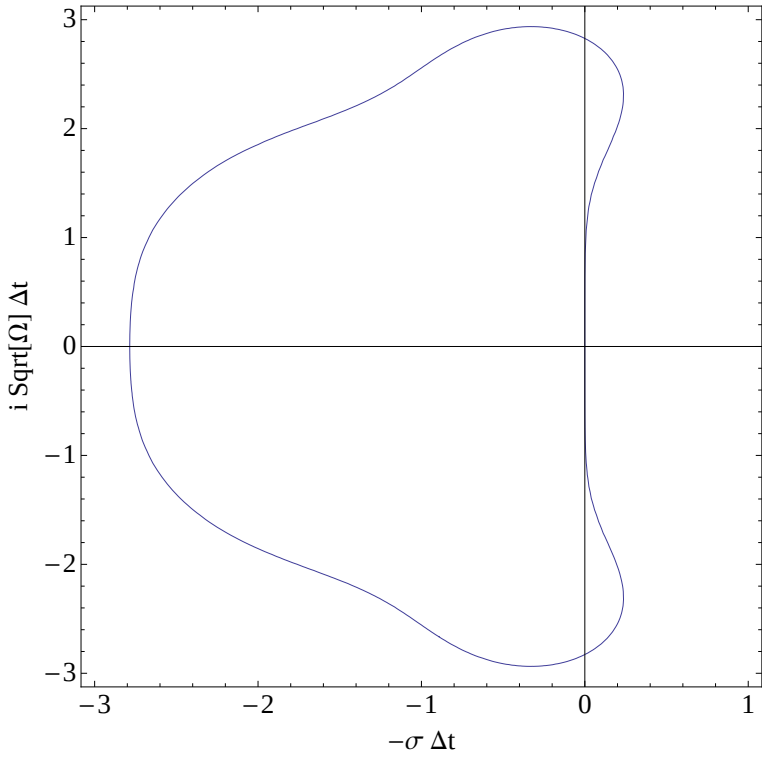

Figure 2.1: Plot of the region of absolute stability for the RK4 advance. Here, the argument of the stability function R is $z = x + iy = -\sigma\Delta t \pm ic\sqrt{\Omega_x^2 + \Omega_y^2 + \Omega_z^2}\Delta t$.

Let $\omega_x = 6\Delta x\Omega_x$. Define ω_y and ω_z similarly. Through straightforward application of calculus, algebra and trigonometry (in particular the double-angle identities) we find the maximum value of each ω_j is

$$\omega_j^{\max} = \sqrt{9 + 24\sqrt{6}} \approx 8.233. \tag{2.84}$$

The maximum height with respect to the x-axis of the region of absolute stability for RK4 is approximately 2.937, which implies that

$$\max\left(c\sqrt{\Omega_x^2 + \Omega_y^2 + \Omega_z^2}\Delta t\right) = \max\left(\frac{c\Delta t}{6\Delta x}\sqrt{\omega_x^2 + \omega_y^2 + \omega_z^2}\right) \approx 2.937, \tag{2.85}$$

assuming that $\Delta x = \Delta y = \Delta z$. Using Eq. (2.84), we find that advancing the source-free Maxwell system in space via the fourth-order stencils [to be given earlier] and in time via RK4 implies the approximate CFL condition

$$\frac{c\Delta t}{\Delta x} \lessapprox \frac{(2.937)(6)}{(8.233)\sqrt{3}} \approx 1.236. \tag{2.86}$$

In general, when advancing hyperbolic equations, $\frac{\Delta t}{\Delta x} \sim$ const. is necessary for resolving waves within the specified accuracy. This constant is typically the inverse of the maximum wave speed of the given system, so Eq. (2.86) does not constitute an excessively restrictive CFL condition for Maxwell's Equations [LeVeque, 2002, pgs. 68–71].

Chapter 3

Implementation

3.1 Chombo code library

Chombo is a set of C++ classes designed to support block-structured AMR applications. The Applied Numerical Algorithms Group (ANAG) at Lawrence Berkeley National Laboratory (LBNL) develops, maintains, and tests these classes [Colella et al., 2012]. While Chombo uses C++ for structure and code steering, most array operations are carried out in Fortran for performance reasons. Chombo provides an interface macro, ChomboFortran, that allows the user to write dimension-independent Fortran code, which itself can be called from C++. We will use the following libraries:

- BaseTools, for dimension-independent helper and data holder classes
- BoxTools, for rectangular array operations, $\mathbb{Z}^n$ set operations, defining data on unions of rectangles, and assigning such data to processors
- AMRTools, for operations common in AMR algorithms, such as interpolation, averaging, and refluxing
- AMRTimeDependent, for advancing hyperbolic PDEs, such as gas dynamics and the Faraday-Ampère-Maxwell system

The most general structure of an AMR code with Chombo classes is illustrated in Fig. 3.1 below:

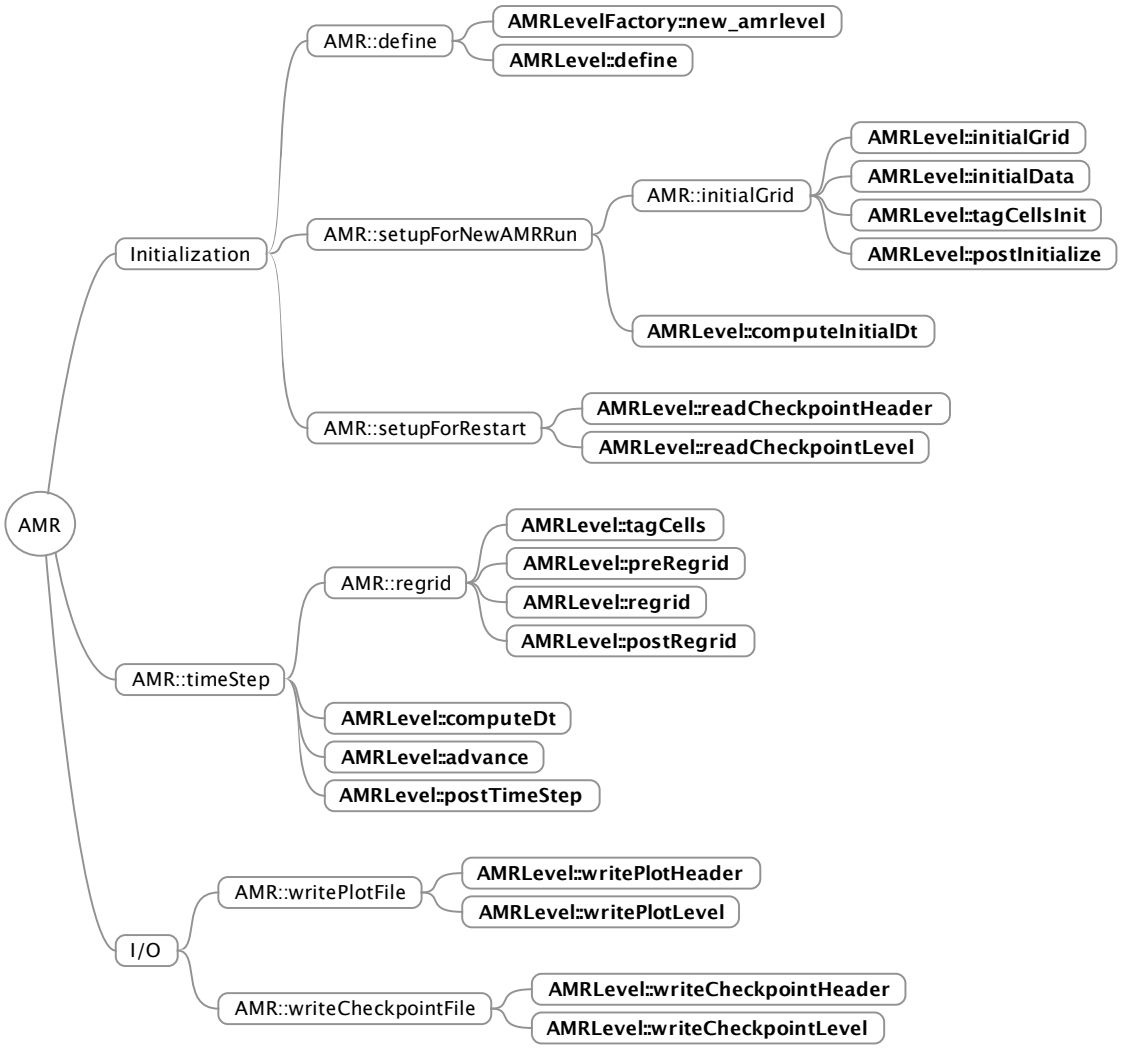

Figure 3.1: General structure of an AMR code. The user typically writes classes derived from `AMRLevel` for specific problems. Reprinted from [Claridge, 2011] with permission from the author.

The primary classes for solving the Faraday-Ampère-Maxwell system, and their basic relationships, are illustrated in Fig. 3.2 below:

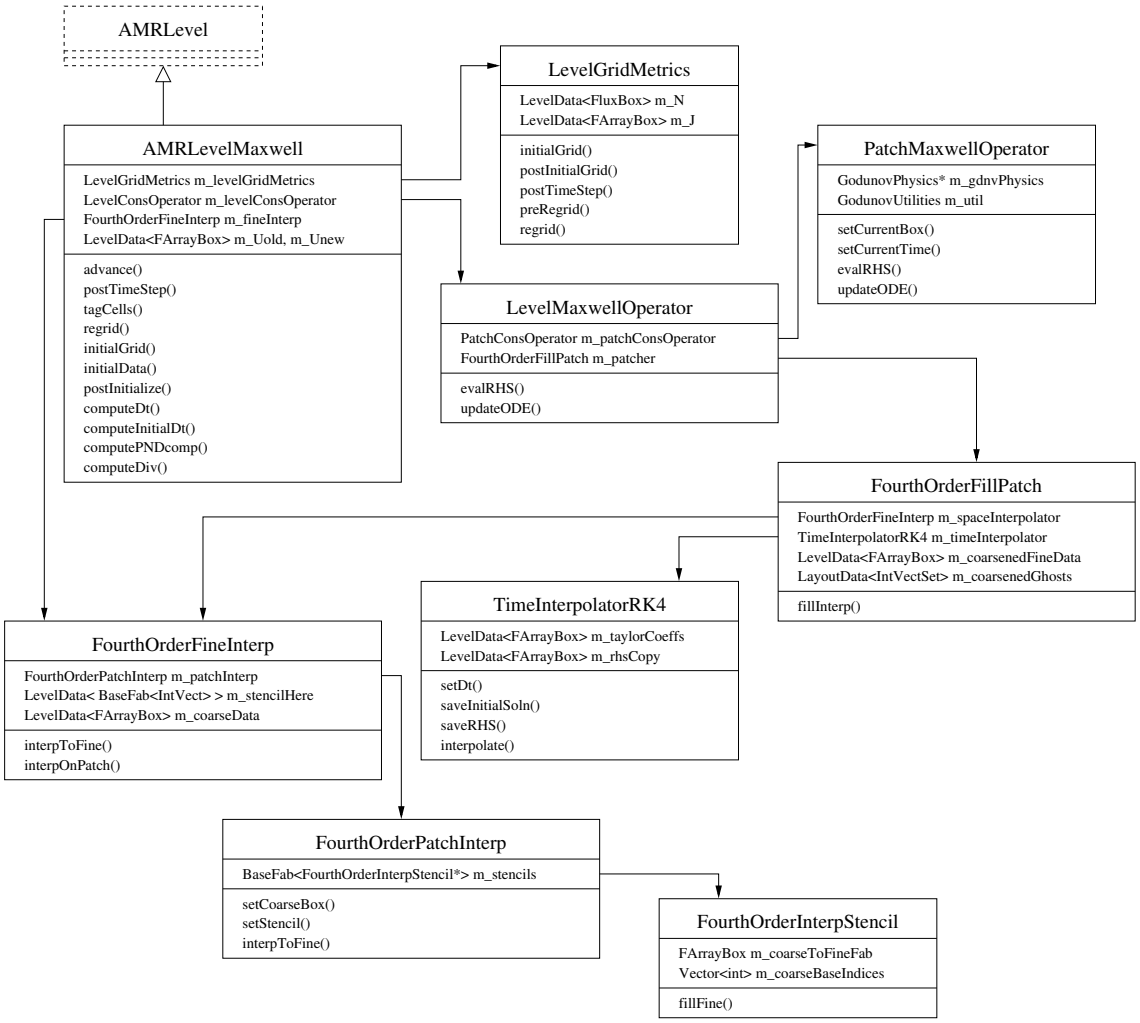

Figure 3.2: Software configuration for the Maxwell code. Modified from Fig. 6 in [McCorquodale, 2012] with permission from the author.

3.1.1 Important Chombo tools

A rectangular problem domain in Chombo is covered by disjoint unions of rectangles at different resolutions. Each such union at a particular level of resolution l is termed Ω^l, which is given by

$$\Omega^l = \bigcup_k \Omega^{l,k}, \tag{3.1}$$

where $\Omega^{l,k}$ is a rectangle at the lth level of resolution. Each $\Omega^{l,k}$, in turn, is divided into square control volumes, or cells. Each cell is identified with a D-dimensional array of integer components $(i_0, ..., i_{\mathbf{D}-1}) = \boldsymbol{i} \in \mathbb{Z}^{\mathbf{D}}$. The cell identified with $\boldsymbol{i}$ occupies the physical space $[\boldsymbol{x}_0 + \boldsymbol{i}h, \boldsymbol{x}_0 + (\boldsymbol{i} + \boldsymbol{u})h]$, where $\boldsymbol{x}_0 \in \mathbb{R}^{\mathbf{D}}$ represents some sort of coordinate origin or offset,

$\boldsymbol{u} = (1, ..., 1) \in \mathbb{Z}^{\mathbf{D}}$, and h is the mesh spacing. The bracket notation lists the coordinates of the lower left and upper right corners of (in this case) the cell, i.e. [lower left corner, upper right corner].

The integer array $\boldsymbol{i}$ is represented by the class `IntVect` in the `BoxTools` library.

Each rectangular grid at a given resolution level l is termed a grid or a box, and is expressed as Γ, B, or $\Omega^{l,k}$, depending on context. A box B is typically expressed in the abbreviated bracket notation given above as $[\boldsymbol{i}_{\text{lower left}}, \boldsymbol{i}_{\text{upper right}}]$, where each $\boldsymbol{i}$ is an `IntVect`. B is represented by the class `Box` in the `BoxTools` library.

Ω^l, in turn, is represented by the class `DisjointBoxLayout`, which also handles the assignment of the `Box`es in Ω^l to different processors.

At each time step, we calculate an m-component array U at each cell in each box on each level. In the context of this project, $m = 6$: 3 components each for $\mathbf{B}$ and $\mathbf{E}$. Our array can thus be described as $U : \mathbb{R}^{\mathbf{D}} \to \mathbb{R}^m$, or more accurately as a mapping from a subset of $\mathbb{R}^{\mathbf{D}}$ to $\mathbb{R}^m$. Though it is expressed in terms of `IntVect` cell indices rather than physical coordinates, each `Box` B is a subset of $\mathbb{R}^{\mathbf{D}}$. U over such subsets, i.e. $U : B \to \mathbb{R}^m$ or $U : \Omega^{l,k} \to \mathbb{R}^m$, is represented by the class `FArrayBox`. In turn, $U : \Omega^l \to \mathbb{R}^m$ is represented by the class `LevelData<FArrayBox>`, essentially a union of `FArrayBox`es [Colella et al., 2012].

3.2 Subcycling in time

Subcycling consists of first advancing the coarsest level ($l = 0$) from time t to $t + \Delta t^0$ (i.e. t plus the coarsest time step), then the next coarsest level ($l = 1$) from t to $t + \Delta t^1$, and so on until we reach the final refinement level $l_{\max}$. Then we keep advancing level $l_{\max}$ from t to $t + n_{\text{ref}}^{l_{\max}-1} \Delta t^{l_{\max}} = t + \Delta t^{l_{\max}-1}$, since $n_{\text{ref}}^{l_{\max}-1} \equiv \Delta t^{l_{\max}-1} / \Delta t^{l_{\max}}$. We then amend the level $l_{\max} - 1$ solution at time $t + \Delta t^{l_{\max}-1}$ with data from level $l_{\max}$ using so-called flux registers. We then repeat the process of advancing level $l_{\max}$ and amending level $l_{\max} - 1$ until we reach time $t + \Delta t^{l_{\max}-2}$, at which point we amend the data on level $l_{\max} - 2$ as well. We then work our way down the levels recursively until all are advanced to $t + \Delta t^0$ and corrected with the appropriate flux registers. This process is illustrated in all the advancement algorithms described in Chapter 2, for example in Alg. 3, and in Fig. 3.3 below.

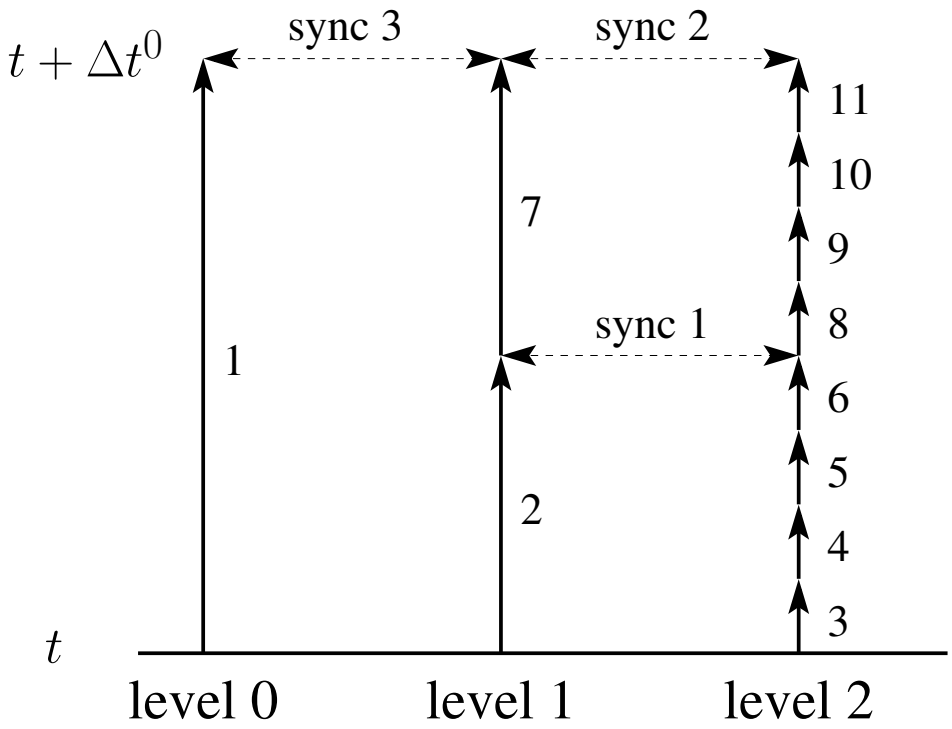

Figure 3.3: Illustration of subcycling in time with a refinement ratio of 2 between levels 0 and 1 and a refinement ratio of 4 between levels 1 and 2. The vertical arrows indicate advancement in time by Δt^l. The horizontal, dotted arrows indicate updates of coarse data via averaging and flux registers. The numbers indicate the order in which the advancements and updates take place.

3.2.1 Averaging down fine solutions to coarse solutions

Now we need to clarify what is meant by amending a coarse solution and flux registers. First, we define a coarsening operator

$$\mathcal{C}_r(\boldsymbol{i}) = \left(\left\lfloor \frac{i_0}{r} \right\rfloor, ..., \left\lfloor \frac{i_{\mathbf{D}-1}}{r} \right\rfloor \right), \tag{3.2}$$

where $\lfloor x \rfloor$ is the largest integer less than or equal to x. We can also apply $\mathcal{C}_r$ to `IntVectSet`s, `Box`es, and `DisjointBoxLayout`s. In particular, $\mathcal{C}_{n_{\text{ref}}^{l-1}}(\Omega^l) \subset \Omega^{l-1}$, i.e. $\mathcal{C}_{n_{\text{ref}}^{l-1}}(\Omega^l)$ is the portion of Ω^{l-1} that Ω^l covers. Similarly, we have an inverse coarsening operator, defined such that $\mathcal{C}^{-1}_{n_{\text{ref}}^{l-1}}\left(\mathcal{C}_{n_{\text{ref}}^{l-1}}(\Omega^l)\right) = \Omega^l$.

After computing $\langle U \rangle^{l_{\max}-1}(t + \Delta t^{l_{\max}-1})$ and $\langle U \rangle^{l_{\max}}(t + n_{\text{ref}}^{l_{\max}-1}\Delta t^{l_{\max}}) = \langle U \rangle^{l_{\max}}(t + \Delta t^{l_{\max}-1})$, we first replace $\langle U \rangle^{l_{\max}-1}(t+\Delta t^{l_{\max}-1})$ with an averaged-down version of $\langle U \rangle^{l_{\max}}(t+$

$\Delta t^{l_{\max}-1}$) in accordance with Eq. (2.48), i.e.

$$\langle U\rangle_{\boldsymbol{i}}^{l_{\max}-1}(t+\Delta t^{l_{\max}-1}) = \frac{1}{(n_{\text{ref}}^{l_{\max}-1})^{\mathbf{D}}}\sum_{\boldsymbol{k}\in\mathcal{F}(\boldsymbol{i})}\langle U\rangle_{\boldsymbol{k}}^{l_{\max}}(t+\Delta t^{l_{\max}-1}) \quad \forall\ \boldsymbol{i}\in\mathcal{C}_{n_{\text{ref}}^{l_{\max}-1}}(\Omega^{l_{\max}}). \tag{3.3}$$

3.2.2 Flux Registers

Now, because $D\cdot F$ is computed with face averages, when we replace $\langle U\rangle^{l_{\max}-1}(t+\Delta t^{l_{\max}-1})$ with its averaged-down equivalent on $\mathcal{C}_{n_{\text{ref}}^{l_{\max}-1}}(\Omega^{l_{\max}})$, we also need to replace $\langle F^d\rangle^{l_{\max}-1}$ on the d-faces of the cells bounding $\mathcal{C}_{n_{\text{ref}}^{l_{\max}-1}}(\Omega^{l_{\max}})$ with values averaged down from $\langle F^d\rangle^{l_{\max}}$ on those boundary faces. We define the sets of d-faces in $\Omega^{l_{\max}-1}$ adjoining $\Omega^{l_{\max}}$ as

$$\zeta_{d,\pm}^{l_{\max}} = \{\boldsymbol{i}\pm\tfrac{1}{2}\boldsymbol{e}_d : \boldsymbol{i}\pm\boldsymbol{e}_d\notin\mathcal{C}_{n_{\text{ref}}^{l_{\max}-1}}(\Omega^{l_{\max}}),\ \boldsymbol{i}\in\mathcal{C}_{n_{\text{ref}}^{l_{\max}-1}}(\Omega^{l_{\max}})\}. \tag{3.4}$$

$\zeta_{d,\pm}^{l_{\max}}$ are, respectively, the sets of high and low d-faces of $\Omega^{l_{\max}}$ expressed in indices of $\Omega^{l_{\max}-1}$. For each $\boldsymbol{i}\pm\frac{1}{2}\boldsymbol{e}_d\in\zeta_{d,+}^{l_{\max}}\cup\zeta_{d,-}^{l_{\max}}$, and each integer $s\in[1,...,n_{\text{ref}}^{l_{\max}-1}]$, we compute

$$\overline{\langle F^d\rangle}_{\boldsymbol{i}\pm\frac{1}{2}\boldsymbol{e}_d}^{l_{\max}}(t+s\Delta t^{l_{\max}}) = \frac{1}{(n_{\text{ref}}^{l_{\max}-1})^{\mathbf{D}-1}}\sum_{\boldsymbol{k}+\frac{1}{2}\boldsymbol{e}_d\in\mathcal{F}^d}\langle F^d\rangle_{\boldsymbol{k}+\frac{1}{2}\boldsymbol{e}_d}^{l_{\max}}(t+s\Delta t^{l_{\max}}). \tag{3.5}$$

The sum is over the set of all boundary d-faces of $\Omega^{l_{\max}}$ which coincide with the d-face $\boldsymbol{i}\pm\frac{1}{2}\boldsymbol{e}_d$ in $\Omega^{l_{\max}-1}$, i.e. $\mathcal{F}^d = [\boldsymbol{i}n_{\text{ref}}^{l_{\max}-1}+\frac{1}{2}\boldsymbol{e}_d, (\boldsymbol{i}+\boldsymbol{u}-\boldsymbol{e}_d)n_{\text{ref}}^{l_{\max}-1}+\frac{1}{2}\boldsymbol{e}_d]$

On $\zeta_{d,+}^{l_{\max}}\cup\zeta_{d,-}^{l_{\max}}$, we define the flux register $\delta F_d^{l_{\max}}$ as

$$\begin{aligned}\delta F_d^{l_{\max}}(t+\Delta t^{l_{\max}-1}) \equiv \frac{1}{n_{\text{ref}}^{l_{\max}-1}}&\sum_{s=1}^{n_{\text{ref}}^{l_{\max}-1}}\overline{\langle F^d\rangle}^{l_{\max}}(t+s\Delta t^{l_{\max}})\\ &-\langle F^d\rangle^{l_{\max}-1}(t+\Delta t^{l_{\max}-1})\end{aligned} \tag{3.6}$$

and the reflux divergence operator D_R as

$$D_R(\delta F^{l_{\max}})_{\boldsymbol{i}} = \frac{1}{h^{l_{\max}-1}}\sum_{d=0}^{\mathbf{D}-1}\sum_{\substack{\pm=+,-:\\ \boldsymbol{i}\pm\frac{1}{2}\boldsymbol{e}_d\in\zeta_{d,\mp}^{l_{\max}}}}\pm\delta F_{d,\boldsymbol{i}\pm\frac{1}{2}\boldsymbol{e}_d}^{f} \tag{3.7}$$

The flux register and the reflux divergence are defined to be zero everywhere else.

After updating $\langle U\rangle^{l_{\max}-1}(t+\Delta t^{l_{\max}-1})$ on $\mathcal{C}_{n_{\text{ref}}^{l_{\max}-1}}(\Omega^{l_{\max}})$ as in Eq. (3.3), we increment it with $-\Delta t^{l_{\max}-1}D_R(\delta F^{l_{\max}})$.

Averaging down and refluxing are carried out recursively down through the refinement levels, whenever $\langle U\rangle^l(t+n_{\text{ref}}^{l-1}\Delta t^l) = \langle U\rangle^l(t+\Delta t^{l-1})$ has been computed and amended appropriately. We simply replace all references to level $l_{\max}$ in the equations and definitions in Sec. 3.2.1 and Sec. 3.2.2 with level l.

The notation in this section differs from that of [Colella et al., 2012, Sec. 3.1.2.3]. This is an attempt to maintain consistency with Sec. 2.5 and the earlier sections of this chapter, not to mention to avoid using $\langle\cdot\rangle$ to denote two different types of averages.

Tagging cells for further refinement, storing and advancing data on multiple levels, and subcycling in time require more complex algorithms and data structures than traditional single-level schemes. Moreover, these algorithms are much more difficult (if not impossible) to implement with procedural rather than object-oriented programming. However, despite the complexity of tagging cells for refinement and of subcycling in time, employing AMR to obtain fine grids only where they are necessary and using coarser grids everywhere else offers tremendous savings in time and computational cost (storage, memory, etc.).

3.3 Damping Coefficient

In choosing the optimum damping coefficient σ, we would ideally specify the various Fourier nodes k_j, space steps Δx_j and domain lengths L_j, from which we would compute the Ω_j values, which we would then use to determine the σ that minimizes $|R(z)|$, with $z = -\sigma\Delta t \pm c\Delta t\sqrt{\Omega_x^2 + \Omega_y^2 + \Omega_z^2}$. However, because our code does not employ spatial Fourier transforms in advancing PDEs, implementing such a function, we choose as our damping coefficient the value of σ corresponding to the tallest part of the RAS. For an RK4 advance of the source-free Maxwell system (for that matter, of any non-dispersive, non-dissipative wave equation) this yields $\sigma \approx 0.329/\Delta t$. This choice allows the greatest possible range of Fourier nodes while remaining inside the RAS.

3.4 Construction of the Sponge Layer

In implementing the damping scheme, we will solve Eq. (2.52) on most of the problem domain. We will only solve Eq. (2.64) in the complement to the proper nesting domain in each refined box B. The proper nesting domain B_{pnd} of each box B is defined as the set of all points contained within B that lie no *fewer* than p units away from any edge of the disjoint union of refined rectangles Ω^l, *not including edges of B or any rectangle contained in Ω^l abutting the problem domain boundary.* The complement to the proper nesting domain of each box B, therefore, is the set of all points contained within B that lie no *more* than p units away from any edge of the disjoint union of refined rectangles Ω^l, not including the

problem domain boundary. In set theoretic notation,

$$\begin{aligned} B_{\text{pnd}} &= \bigcap_{v\in[-pu,pu]} \left((\Omega^l_{\text{mod}} + v) \cap B_{\text{mod}}\right) \\ &= \bigcap_{v\in[-pu,pu]} \left((\cup_k \Omega^{l,k}_{\text{mod}} + v) \cap B_{\text{mod}}\right) \\ &= \bigcap_{v\in[-pu,pu]} \cup_k \left((\Omega^{l,k}_{\text{mod}} + v) \cap B_{\text{mod}}\right), \end{aligned} \tag{3.8}$$

where l is the refinement level of the disjoint union of boxes of which B is a part, $\Omega^{l,k}$ denotes an arbitrary box within Ω^l, v is an offset vector, u is a vector with components all equal to ± 1, and $v \in [-pu, pu]$ is an abbreviated notation meaning $v \in p(\pm 1, \pm 1, ..., \pm 1)$. B_{mod} is defined by the following algorithm

Algorithm 1 $B_{\text{mod}}(B, \Omega)$: Growing a box B so that B_{pnd} abuts the problem domain Ω if B abuts Ω

$B_{\text{mod}} := B$
$\boldsymbol{i}_{\text{lo}}$ = cell index of lower left corner of B_{mod}
$\boldsymbol{i}_{\text{hi}}$ = cell index of upper right corner of B_{mod}
$\boldsymbol{j}_{\text{lo}}$ = cell index of lower left corner of problem domain
$\boldsymbol{j}_{\text{hi}}$ = cell index of upper right corner of problem domain
for $d \in [0, \mathbf{D} - 1]$ **do**
  if $\boldsymbol{i}_{\text{lo}} \cdot \boldsymbol{e}_d = \boldsymbol{j}_{\text{lo}} \cdot \boldsymbol{e}_d$ **then**
    $\boldsymbol{i}_{\text{lo}} := \boldsymbol{i}_{\text{lo}} - \boldsymbol{e}_d$
  end if
  if $\boldsymbol{i}_{\text{hi}} \cdot \boldsymbol{e}_d = \boldsymbol{j}_{\text{hi}} \cdot \boldsymbol{e}_d$ **then**
    $\boldsymbol{i}_{\text{hi}} := \boldsymbol{i}_{\text{hi}} + \boldsymbol{e}_d$
  end if
end for

with each box $\Omega^{l,k}_{\text{mod}}$ within Ω^l_{mod} defined similarly.
The complement to B_{pnd}, B^c_{pnd}, is given by

$$B^c_{\text{pnd}} = (B_{\text{mod}} - B_{\text{pnd}}) \cap B. \tag{3.9}$$

Fig. 3.4 illustrates the relationships between B, B_{mod}, B_{pnd}, B^c_{pnd}, Ω^l, and Ω^l_{mod} for an arbitrary B and Ω^l.

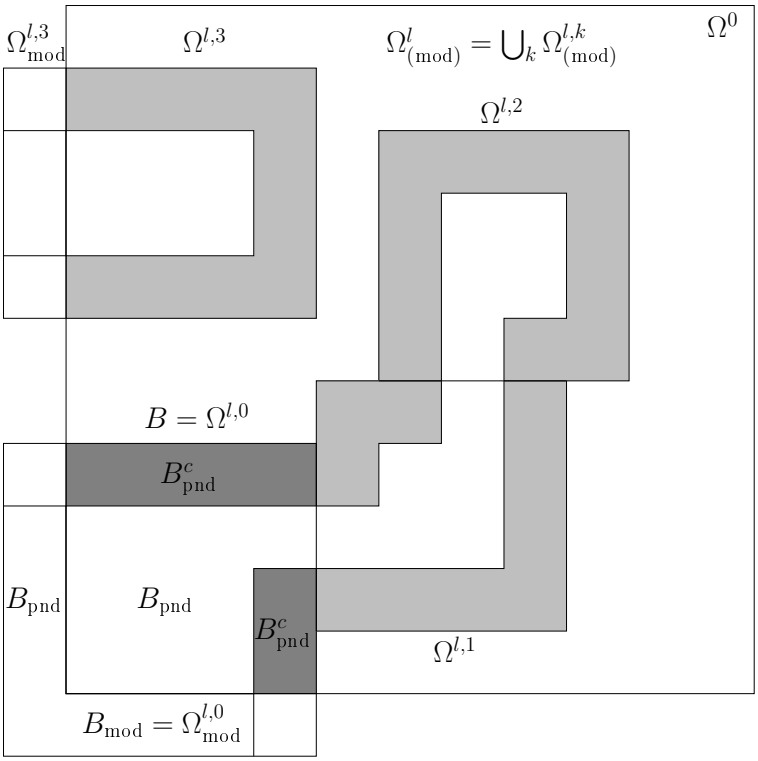

Figure 3.4: An arbitrary box $B = \Omega^{l,0}$ abutting the boundary of the problem domain Ω^0, the disjoint box layout Ω^l to which the box belongs, the box's proper nesting domain B_{pnd}, and the complement to the proper nesting domain B^c_{pnd}, a.k.a. the sponge layer.

The sponge layer is constructed in the `levelSetup()` function of the class `AMRLevelMaxwell`. As sponge layers and grids are intrinsically related, the two are created simultaneously. We have added the following two functions to the `AMRLevelMaxwell` class to assist in the construction of the sponge layer:

- `IntVectSet shiftVectSet(const int& a_sf)`

  Construct a complete set of `IntVect`s with all components equal to $\pm$`a_sf`. Corresponds to the set of vectors v by which we shift the box $\Omega^{l,k}$ in Eq. (3.8).

  - `a_sf`: The absolute value of each component of each vector in the set of shifting vectors. Corresponds to p in Eq. (3.8).

- ```
  IntVectSet computePNDcomp(const Box&              a_box,
                            const int&              a_shift,
                            const DisjointBoxLayout& a_dbl)
  ```

  Compute the PND complement of `a_box` as in Alg. 1, Eq. (3.8), and Eq. (3.9).
  ```

- `a_box`: Box whose PND complement we are computing. Corresponds to B in Eq. (3.8).
- `a_shift`: Integer specifying the magnitude of each component of the shifting vectors. Corresponds to p in Eq. (3.8).
- `a_dbl`: Set of all boxes at the level l where this function is called. Corresponds to Ω^l in Eq. (3.8).

Numerical experiments with the scalar wave equation (and [Harris and Durran, 2010]) seem to suggest that ramping up σ from the inside of the sponge layer (the layer of cells directly adjacent to the proper nesting domain of the fine region) to the outside (the layer of cells adjacent to the coarse-fine grid boundary) yields better damping than would be obtained from a single σ in the entire sponge layer. We have tested several formulas for ramping up σ in a sponge layer N cells thick. To simplify our damping coefficient formulas, let us define n as an integer $\in [1, N]$, $\xi \equiv n/N$, and $\sigma_0 \equiv 0.329/\Delta t$. In this case, $n = 1$ corresponds to the layer of coarse cells adjacent to the proper nesting domain and $n = N$ is the layer of coarse cells just on the fine side of the coarse-fine grid boundary.

Harris and Durran ramp up σ linearly [Harris and Durran, 2010]:

$$\sigma(\xi) = \sigma_0 \xi. \tag{3.10}$$

As such, this was the first damping coefficient ramp-up scheme we implemented.

However, we then began to experiment with ramp-up schemes smoother at $n = 1$ and $n = N$. To that end, we tested ramp-up formulas of the form

$$\sigma(\xi) = \sigma_0 \sin^p \left(\frac{\pi \xi}{2} \right), \tag{3.11}$$

where p is a positive integer.

We determined through more numerical experimentation that the sinusoidal ramp-up schemes resulted in greater errors and erroneous growth in E_y in the case of a plane wave propagating along the x-axis from a finely resolved to a coarsely resolved region, with the wave marginally resolved on the fine grid and clearly underresolved on the coarse grid. We also implemented several ramp-up schemes of the form

$$\sigma(\xi) = \sigma_0 C \int (\xi(1-\xi))^p d\xi, \tag{3.12}$$

with no integration constant and C chosen such that $C \int_0^1 (\xi(1-\xi))^p d\xi = 1$. In theory, this results in as much cumulative damping as the linear scheme in Eq. (3.10), since

$$\int_0^1 \xi d\xi = \frac{1}{2} = \int_0^1 C \int_0^\xi (\xi'(1-\xi'))^p d\xi', \tag{3.13}$$

assuming C is chosen as described above. Furthermore, since $(\xi(1-\xi))^p$ is symmetric about $\xi = 1/2$, σ chosen according to Eq. (3.12) will equal $1/2$ at $\xi = 1/2$, i.e. halfway into $B^c_{\rm pnd}$.

We construct the sponge layer with varying σ via the following pseudocode:

Algorithm 2 Construction of the sponge layer at level l

for $n \in [1, N]$ **do**
  $p = (N - n + 1)\Delta x^{l-1}/\Delta x^l$
  for each `Box` B in `DisjointBoxLayout` Ω^l **do**
    $B_{\rm pnd} = \bigcap_{v\in[-pu,pu]} \cup_k \left((\Omega^{l,k}_{\rm mod} + v) \cap B_{\rm mod}\right)$
    $B^c_{\rm pnd} = (B_{\rm mod} - B_{\rm pnd}) \cap B$
    $\sigma = \sigma(\xi)$ in $B^c_{\rm pnd}$
  end for
end for

3.5 Interpolation

The basic algorithm for constructing the right hand side of Eq. (2.64) in Chombo is:

- Set up coarsened versions of the fine grids, with the appropriate number of coarse ghost cells. The first part of this is carried out with the `coarsen` modification function defined in the `DisjointBoxLayout` class. The coarse level ghost vectors and coarsened version of the fine grids are then used in defining a `LevelData<FArrayBox>` named `coarsenedFineU`.

- Interpolate coarse data on coarsened versions of fine grids, defined at t^{l-1} (i.e., the time on the next coarser level from which we're advancing) to coarse data on coarsened versions of fine grids, defined at $t^l + \Delta t^l$ (i.e., the time on the current level to which we're advancing). This is carried out when the `interpolate` function of the `TimeInterpolatorRK4` class.

- Interpolate coarsened fine data at $t^l + \Delta t^l$ to fine data at $t^l + \Delta t^l$. This is carried out `FArrayBox` by `FArrayBox`, as follows:

- **for** each `FArrayBox` in U^l

  - Get the associated coarsened fine `Box`
  - Grow that `Box` by the number of coarse ghost cells corresponding to the fine ghost cells that need to be filled
  - Convert the ghosted, coarsened fine `Box` to an `IntVectSet`.
  - Interpolate coarsened data on the ghosted, coarsened fine `Box`

- **endfor**
- **for** each `FArrayBox` in U^l
  - Negate interpolated coarse data ($I(U^{l-1})$) with the `*=` operator.
  - Add fine data (U^l) with `+=` operator.
  - Multiply by $1/\Delta t$ with `*=` operator.
  - Multiply by $\sigma \Delta t$, which is non-zero in the sponge layer and zero everywhere else.
- **endfor**

3.6 Advancing the basic, source-free hyperbolic system

We will advance Eq. (2.52) as described in [McCorquodale and Colella, 2011] and [Colella et al., 2012, Ch. 5]. As indicated in [Colella et al., 2012], the advancement algorithm for this system is given by Alg. 3:

Algorithm 3 advance(l): Advancing a simple hyperbolic conservation system

$U^l(t^l + \Delta t^l) := \text{RK4}\left(U^l(t^l), t^l, \Delta t^l\right)$, with $\frac{\partial U^l}{\partial t} = -\nabla \cdot F(U^l)$
if $l < l_{\max}$ **then**
  $\delta F^l_d = -F^l_d$ on $\zeta^{l+1}_{+,d} \cup \zeta^{l+1}_{-,d}$, $\forall\ d \in [0, \mathbf{D}-1]$
end if
if $l > 0$ **then**
  $\delta F^l_d := \frac{1}{n^{l-1}_{\text{ref}}} \overline{F^d}^l$ on $\zeta^l_{+,d} \cup \zeta^l_{-,d}$, $\forall\ d \in [0, \mathbf{D}-1]$
end if
for $q \in [0, n^l_{\text{ref}} - 1]$ **do**
  advance($l+1$)
end for
$U^l(t^l + \Delta t^l) = \text{average}(U^{l+1}(t^l + \Delta t^l), n^l_{\text{ref}})$ on $\mathcal{C}_{n^l_{\text{ref}}}(\Omega^{l+1})$
$U^l(t^l + \Delta t^l) := U^l(t^l + \Delta t^l) - \Delta t^l D_R(\delta F^{l+1})$
$t^l := t^l + \Delta t^l$
$n^l_{\text{step}} := n^l_{\text{step}} + 1$
if $(n^l_{\text{step}} = 0 \mod n_{\text{regrid}})$ and $(n^{l-1}_{\text{step}} \neq 0 \mod n_{\text{regrid}})$ **then**
  regrid(l)
end if

3.7 Advancing the damped, source-free system

Advancing Eq. (2.64) is similar to advancing Eq. (2.52). We merely add the steps outlined in Sec. 3.5 to Alg. 3, as indicated below:

Algorithm 4 advance(l): Advancing a damped hyperbolic conservation system

$U^l(t^l + \Delta t^l) :=$ RK4 $\left(U^l(t^l), t^l, \Delta t^l\right)$, with $\frac{\partial U^l}{\partial t} = -\nabla \cdot F(U^l)$
if $l > 0$ **then**
  copy $U^{l-1}(t^{l-1})$ on Ω^{l-1} to $\bar{U}^{l-1}(t^{l-1})$ on $\mathcal{C}_{n_{\text{ref}}^{l-1}}(\Omega^l)$
  $\bar{U}^{l-1}(t^l + \Delta t^l) :=$ timeInterp $\left(\bar{U}^{l-1}(t^{l-1}),\ t^l + \Delta t^l\right)$
  $I(U^{l-1})(t^l + \Delta t^l) :=$ spaceInterp $\left(\bar{U}^{l-1}(t^l + \Delta t^l)\right)$
  $U^l(t^l + \Delta t^l) := U^l(t^l + \Delta t^l) - \sigma \Delta t^l \left[U^l(t^l + \Delta t^l) - I(U^{l-1})(t^l + \Delta t^l)\right]$
end if
if $l < l_{\max}$ **then**
  $\delta F_d^l = -F_d^l$ on $\zeta_{+,d}^{l+1} \cup \zeta_{-,d}^{l+1}$, $\forall\ d \in [0, \mathbf{D} - 1]$
end if
if $l > 0$ **then**
  $\delta F_d^l := \frac{1}{n_{\text{ref}}^{l-1}} \overline{F^d}^l$ on $\zeta_{+,d}^l \cup \zeta_{-,d}^l$, $\forall\ d \in [0, \mathbf{D} - 1]$
end if
for $q \in [0, n_{\text{ref}}^l - 1]$ **do**
  advance($l + 1$)
end for
$U^l(t^l + \Delta t^l) =$ average$(U^{l+1}(t^l + \Delta t^l), n_{\text{ref}}^l)$ on $\mathcal{C}_{n_{\text{ref}}^l}(\Omega^{l+1})$
$U^l(t^l + \Delta t^l) := U^l(t^l + \Delta t^l) - \Delta t^l D_R(\delta F^{l+1})$
$t^l := t^l + \Delta t^l$
$n_{\text{step}}^l := n_{\text{step}}^l + 1$
if $(n_{\text{step}}^l = 0 \mod n_{\text{regrid}})$ and $(n_{\text{step}}^{l-1} \neq 0 \mod n_{\text{regrid}})$ **then**
  regrid(l)
end if

3.8 Advancing the damped system with sources

Assuming that the current density $\mathbf{J}$ is specified and that we can compute its divergence-free projection $\mathbb{P}(\mathbf{J})$ we need only make slight modifications to Alg. 4, as indicated below:

Algorithm 5 advance(l): Advancing a damped hyperbolic system with sources

$U^l(t^l + \Delta t^l) := \text{RK4}\left(U^l(t^l), t^l, \Delta t^l\right)$, with $\frac{\partial U^l}{\partial t} = S^l - \nabla \cdot F(U^l)$
if $l > 0$ **then**
  copy $U^{l-1}(t^{l-1})$ on Ω^{l-1} to $\bar{U}^{l-1}(t^{l-1})$ on $\mathcal{C}_{n^{l-1}_{\text{ref}}}(\Omega^l)$
  $\bar{U}^{l-1}(t^l + \Delta t^l) := \text{timeInterp}\left(\bar{U}^{l-1}(t^{l-1}),\ t^l + \Delta t^l\right)$
  $I(U^{l-1})(t^l + \Delta t^l) := \text{spaceInterp}\left(\bar{U}^{l-1}(t^l + \Delta t^l)\right)$
  $U^l(t^l + \Delta t^l) := U^l(t^l + \Delta t^l) - \sigma \Delta t^l \left[U^l(t^l + \Delta t^l) - I(U^{l-1})(t^l + \Delta t^l)\right]$
end if
if $l < l_{\max}$ **then**
  $\delta F^l_d = -F^l_d$ on $\zeta^{l+1}_{+,d} \cup \zeta^{l+1}_{-,d}$, $\forall\ d \in [0, \mathbf{D} - 1]$
end if
if $l > 0$ **then**
  $\delta F^l_d := \frac{1}{n^{l-1}_{\text{ref}}} \overline{F^d}^l$ on $\zeta^l_{+,d} \cup \zeta^l_{-,d}$, $\forall\ d \in [0, \mathbf{D} - 1]$
end if
for $q \in [0, n^l_{\text{ref}} - 1]$ **do**
  advance($l + 1$)
end for
$U^l(t^l + \Delta t^l) = \text{average}(U^{l+1}(t^l + \Delta t^l), n^l_{\text{ref}})$ on $\mathcal{C}_{n^l_{\text{ref}}}(\Omega^{l+1})$
$U^l(t^l + \Delta t^l) := U^l(t^l + \Delta t^l) - \Delta t^l D_R(\delta F^{l+1})$
$t^l := t^l + \Delta t^l$
$n^l_{\text{step}} := n^l_{\text{step}} + 1$
if $(n^l_{\text{step}} = 0 \mod n_{\text{regrid}})$ and $(n^{l-1}_{\text{step}} \neq 0 \mod n_{\text{regrid}})$ **then**
  regrid(l)
end if

Chapter 4

Source-Free Maxwell's Equations: Test Problems and Results

4.1 2D test problems

For the remainder of this section, we will assume that no quantity depends on z. We can think of this as a two-dimensional system, where B_z and E_z are scalar-valued. In that case, Eq. (2.52) decouples into

$$\frac{\partial}{\partial t}\begin{pmatrix} B_x \\ B_y \\ E_z \end{pmatrix} + \nabla \cdot \begin{pmatrix} 0 & -E_z & -c^2 B_y \\ E_z & 0 & c^2 B_x \end{pmatrix} = 0 \tag{4.1}$$

and

$$\frac{\partial}{\partial t}\begin{pmatrix} E_x \\ E_y \\ B_z \end{pmatrix} + \nabla \cdot \begin{pmatrix} 0 & c^2 B_z & E_y \\ -c^2 B_z & 0 & -E_x \end{pmatrix} = 0. \tag{4.2}$$

Let us now restrict ourselves to systems in which B_x, B_y and E_z are initially equal to 0 and thus, according to Eq. (4.1), equal to 0 for all time. The simplest possible such solutions to Eq. (4.2) and Gauss' Laws are systems of the form

$$\begin{aligned} E_x(x,t) &= 0, \\ E_y(x,t) &= \mathcal{E} f(k_x(x-ct)), \\ B_z(x,t) &= \frac{\mathcal{E}}{c} f(k_x(x-ct)), \end{aligned} \tag{4.3}$$

where $f(k_x(x-ct))$ is an arbitrary, dimensionless function and $\mathcal{E}$ is a constant with units of electric field (kg m C^{-1} s^{-2} in SI units). Note that E_y and B_z are solutions to wave equations: $E_y(x,0) \propto B_z(x,0) \propto f(k_x x)$ and E_y and B_z maintain their forms while propagating along the x-axis at speed c.

A more general system of waves propagating in any direction in the xy plane can be expressed as

$$\begin{aligned} E_x(x,y,t) &= \mathcal{E}_x f(k_x x + k_y y - \omega t), \\ E_y(x,y,t) &= \mathcal{E}_y f(k_x x + k_y y - \omega t), \\ B_z(x,y,t) &= \mathcal{B}_z f(k_x x + k_y y - \omega t). \end{aligned} \tag{4.4}$$

Gauss' Law of Magnetism is automatically satisfied.

From Gauss' Law of Electricity, we have

$$k_x \mathcal{E}_x + k_y \mathcal{E}_y = 0. \tag{4.5}$$

If $\mathcal{E}_x \neq 0$, then

$$\mathcal{E}_y = -\frac{k_x \mathcal{E}_x}{k_y}, \tag{4.6}$$

while if $\mathcal{E}_x = 0$, $k_y = 0$.

Faraday's Law yields

$$\mathcal{B}_z = \frac{k_x \mathcal{E}_y - k_y \mathcal{E}_x}{\omega}, \tag{4.7}$$

or, if $\mathcal{E}_x \neq 0$,

$$\mathcal{B}_z = \frac{-(k_x^2 + k_y^2)\mathcal{E}_x}{\omega k_y}. \tag{4.8}$$

The Ampère-Maxwell Law yields

$$-\omega \mathcal{E}_x - c^2 k_y \mathcal{B}_z = 0 \tag{4.9}$$

and

$$-\omega \mathcal{E}_y + c^2 k_x \mathcal{B}_z = 0. \tag{4.10}$$

Substituting Eq. (4.9) and Eq. (4.10) into Eq. (4.7) yields

$$\mathcal{B}_z = \frac{c^2(k_x^2 + k_y^2)}{\omega^2}\mathcal{B}_z, \tag{4.11}$$

which in turn implies that

$$\omega^2 = c^2(k_x^2 + k_y^2). \tag{4.12}$$

Eqs. (4.3) are a special case of Eqs. (4.4), with $\mathcal{E}_x = 0$, $\mathcal{E}_y = \mathcal{E}$, $k_y = 0$, with all other equivalences following from these.

Test problems we programmed and ran include:

$$f(k_x(x - ct)) = \cos(2\pi K_x(x - ct)), \tag{4.13}$$

with K_x, the number of periods per unit length, equal to 1 rad^{-1} m^{-1} ($k_x = 2\pi K_x$ is the wave number);

$$f(k_x(x - ct)) = \exp\left(-4 \ln 2 \frac{(x - ct)^2}{w^2}\right), \tag{4.14}$$

where w (equal to 1 m in our simulations) is the full width at half maximum of the Gaussian pulse, related to the standard deviation σ by $w = 2\sigma\sqrt{2\ln 2}$ (note that the standard deviation of a Gaussian σ is completely unrelated to the damping coefficient σ);

$$f(k_x x + k_y y - \omega t) = \cos\left(2\pi\left[K_x x + K_y y - \sqrt{K_x^2 + K_y^2}\, ct\right]\right), \tag{4.15}$$

with $(K_x, K_y) = (1,1)$, $(1,2)$, and $(2,1)$ rad^{-1} m^{-1};

$$f(k_x x + k_y y - \omega t) = \cos\left(2\pi K[(\cos\alpha)x + (\sin\alpha)y - ct]\right), \tag{4.16}$$

with $K = 1$ rad^{-1} m^{-1} and $\alpha = \pi/6,\ \pi/4$, and $\pi/3$ rad; and

$$f(k_x x + k_y y - \omega t) = \exp\left(-4\ln 2\frac{[(\cos\alpha)x + (\sin\alpha)y - ct]^2}{w^2}\right), \tag{4.17}$$

with $w = 1$ m and $\alpha = \pi/6,\ \pi/4$, and $\pi/3$ rad.

The plane wave problems were advanced on a 32 m X 4 m coarse grid, with an $l = 1$ refined patch of 4 m X 4 m on the left side of the problem domain. To check convergence, the angled plane wave problems, i.e. those with solutions corresponding to Eqs. (4.15) and (4.16), were also advanced on this grid, albeit without the refined patch. When possible, periodic boundary conditions were employed. This was not possible with the refined patch on the left side of the problem domain, as the left and right x-boundaries were resolved differently. Likewise, the periodicity of Eq. (4.16) did not match that of the problem domain. In those cases, the actual solutions were imposed as boundary conditions.

The Gaussian pulse problems were advanced on a 32 m X 8 m coarse grid, with an $l = 1$ refined patch of 4 m X 4 m in the center of the problem domain. To test convergence, Eq. (4.17) was also advanced on this grid, without the refined patch. For both Eqs. (4.14) and (4.17), the actual solution was imposed on the x-boundaries. We also had to impose the actual solution on the y-boundaries for Eq. (4.17), but were able to employ periodic boundary conditions in y for Eq. (4.14).

4.2 3D test problems

Let us now consider the source-free Maxwell Equations in 3 dimensions. Since Eqs. (2.51) can be expressed as wave equations in $\mathbf{B}$ and $\mathbf{E}$, traveling wave solutions are worth investigating.

A fairly general solution to the scalar wave equation (not the most general, but general enough for our purposes) can be written as $f(k_x x + k_y y + k_z z - \omega t + \alpha) = f(\boldsymbol{k}\cdot\boldsymbol{x} - \omega t + \alpha)$, provided that $\omega^2 = c^2k^2 = c^2(k_x^2 + k_y^2 + k_z^2)$. Thus, it seems reasonable to consider electric fields of the form

$$\mathbf{E}(\boldsymbol{x}, t) = \mathcal{E} f(\boldsymbol{k}\cdot\boldsymbol{x} - \omega t + \alpha)\boldsymbol{e}_a, \tag{4.18}$$

where $\boldsymbol{e}_a$ is an arbitrary unit vector [Jackson, 1998]. Clearly,

$$\boldsymbol{\nabla}\cdot\mathbf{E} = \mathcal{E} f'\left(\boldsymbol{k}\cdot\boldsymbol{x} - \omega t + \alpha\right)\boldsymbol{k}\cdot\boldsymbol{e}_a. \tag{4.19}$$

If $\boldsymbol{k} \perp \boldsymbol{e}_a$, i.e. $\boldsymbol{k}\cdot\boldsymbol{e}_a = 0$, $\boldsymbol{\nabla}\cdot\mathbf{E} = 0$ and the source-free version of Gauss' Law of Electricity is satisfied. However, for any $\boldsymbol{k}$ in 3-dimensional space, there exists an entire plane of vectors perpendicular to $\boldsymbol{k}$. Thus, we may choose unit vectors $\boldsymbol{e}_1$ and $\boldsymbol{e}_2$ that are perpendicular to each other and to $\boldsymbol{k}$. If we define $\boldsymbol{n} \equiv \boldsymbol{k}/|\boldsymbol{k}|$ (borrowing Jackson's notation), we may further specify $\boldsymbol{e}_1$ and $\boldsymbol{e}_2$ by requiring

$$\begin{aligned} \boldsymbol{e}_1 \times \boldsymbol{e}_2 &= \boldsymbol{n}, \\ \boldsymbol{e}_2 \times \boldsymbol{n} &= \boldsymbol{e}_1, \\ \boldsymbol{n} \times \boldsymbol{e}_1 &= \boldsymbol{e}_2. \end{aligned} \tag{4.20}$$

Since Maxwell's Equations and the wave equation are linear, a more general traveling electric field can be written as

$$\mathbf{E}(\boldsymbol{x}, t) = \mathcal{E}_1 f_1(\boldsymbol{k}\cdot\boldsymbol{x} - \omega t + \alpha_1)\boldsymbol{e}_1 + \mathcal{E}_2 f_2(\boldsymbol{k}\cdot\boldsymbol{x} - \omega t + \alpha_2)\boldsymbol{e}_2, \tag{4.21}$$

where f_1 and f_2 both solve the scalar wave equation, but may otherwise be completely unrelated to each other. Moreover, the phase angles α_1 and α_2 need not be equal, and $\mathcal{E}_1$ need not equal $\mathcal{E}_2$.

Now suppose that

$$\mathbf{B}(\boldsymbol{x}, t) = -\frac{\mathcal{E}_2}{c} f_2(\boldsymbol{k}\cdot\boldsymbol{x} - \omega t + \alpha_2)\boldsymbol{e}_1 + \frac{\mathcal{E}_1}{c} f_1(\boldsymbol{k}\cdot\boldsymbol{x} - \omega t + \alpha_1)\boldsymbol{e}_2. \tag{4.22}$$

We will see shortly that $\mathbf{E}$ and $\mathbf{B}$ defined thus satisfy the source-free Maxwell Equations. Gauss' Laws are automatically satisfied, as we know from Eq. (4.19) and our requirements that $\boldsymbol{k}\cdot\boldsymbol{e}_1 = 0 = \boldsymbol{k}\cdot\boldsymbol{e}_2$.

Faraday's Law becomes

$$\frac{\omega}{c}\mathcal{E}_i f_i'(\boldsymbol{k}\cdot\boldsymbol{x} - \omega t + \alpha_i) = k\mathcal{E}_i f_i'(\boldsymbol{k}\cdot\boldsymbol{x} - \omega t + \alpha_i), \tag{4.23}$$

where the i subscript stands for either 1 or 2. Assuming that our waves travel only along $\boldsymbol{k}$ and not $-\boldsymbol{k}$, $\omega = ck$ and Eq. (4.23) is an identity.

Similarly, the Ampère-Maxwell Law becomes

$$\omega\mathcal{E}_i f_i'(\boldsymbol{k}\cdot\boldsymbol{x} - \omega t + \alpha_i) = ck\mathcal{E}_i f_i'(\boldsymbol{k}\cdot\boldsymbol{x} - \omega t + \alpha_i), \tag{4.24}$$

which is also an identity.

Thus, $\mathbf{E}$ and $\mathbf{B}$ as defined by Eq. (4.21) and Eq. (4.22) form a broad class of traveling wave solutions to the source-free Maxwell Equations. Indeed, Eqs. (4.4) are a special case of Eqs. (4.21) and (4.22), with

$$\begin{aligned} \boldsymbol{k} &= k_x\boldsymbol{e}_x + k_y\boldsymbol{e}_y, \\ \boldsymbol{e}_1 &= \frac{-k_y\boldsymbol{e}_x + k_x\boldsymbol{e}_y}{k}, \\ \boldsymbol{e}_2 &= \boldsymbol{e}_z, \\ \mathcal{E}_1 &= \frac{k_x\mathcal{E}_y - k_y\mathcal{E}_x}{k}, \\ \mathcal{E}_2 &= 0. \end{aligned} \tag{4.25}$$

We also need Eq. (4.5) for Eqs. (4.25) to hold.

4.3 Results

For plane waves and Gaussian pulses propagating solely along the x-axis, i.e. solutions proportional to Eqs. (4.13) and (4.14), we observed 4th-order convergence (assuming sufficient resolution), as illustrated below:

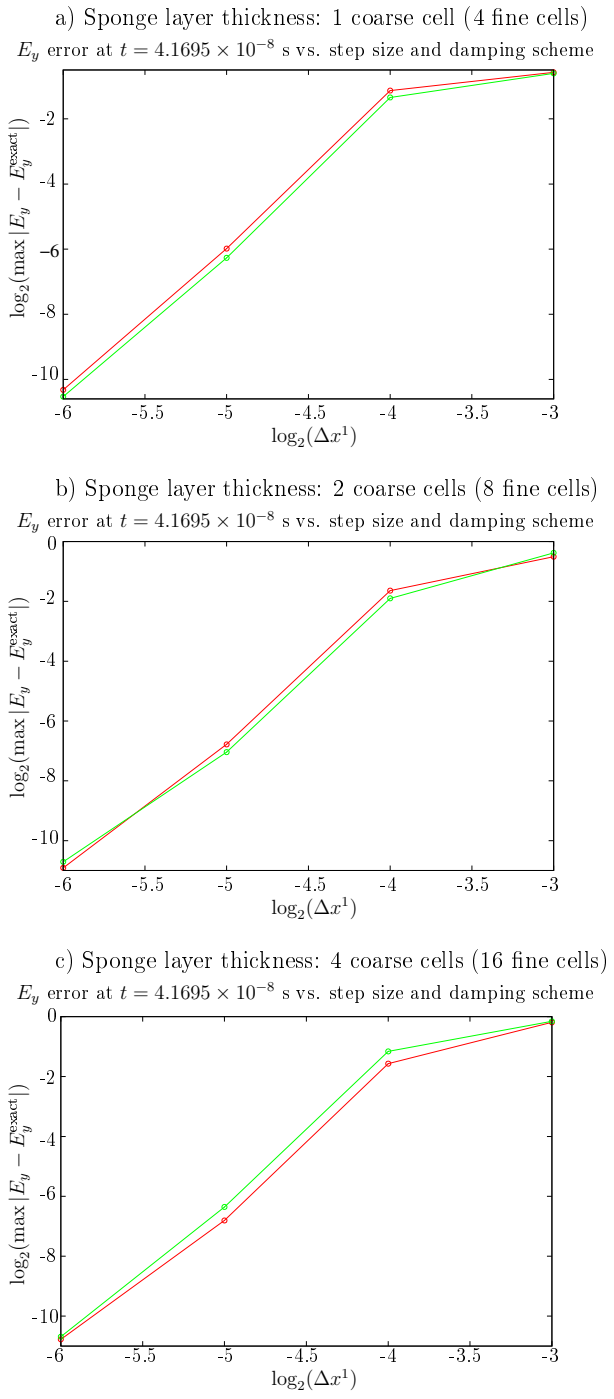

Figure 4.1: Log-log plots of error in E_y vs. Level 1 step size and damping scheme for a plane wave propagating along the x-axis, i.e. Eq. (4.13). The red curves represent linear ramp-up in σ, while the green curves represent sinusoidal ramp-up.

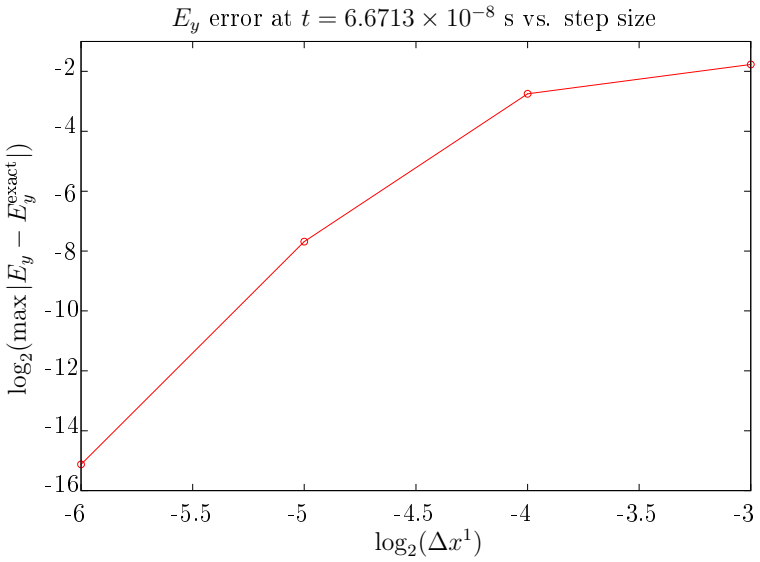

Figure 4.2: Log-log plots of error in E_y vs. Level 1 step size for a Gaussian wave propagating along the x-axis, i.e. Eq. (4.14). In this case, we have a sponge layer 2 coarse cells thick and linear ramp-up in σ.

In advancing Eq. (4.15) with $K_x = K_y = 1$ rad^{-1} m^{-1} on an unrefined grid, and with periodic boundary conditions, we observed (nearly) 4th-order accuracy, given proper resolution:

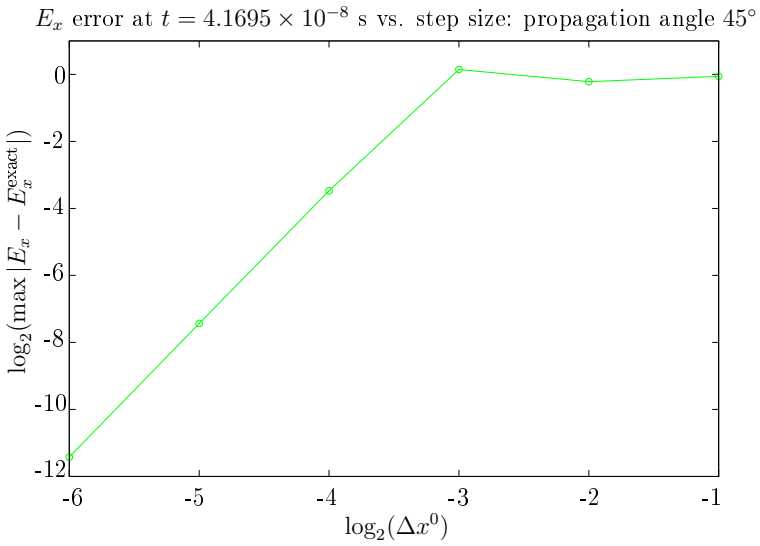

Figure 4.3: Log-log plot of error in E_x vs. step size on an unrefined grid and with periodic BCs for Eq. (4.15) with $K_x = K_y = 1$ rad^{-1} m^{-1}.

However, we observed decidedly slower convergence for both plane waves and Gaussian pulses when refined grids and sponge layers were involved, and when we needed to impose exact solutions as boundary conditions, as Fig. 4.4 illustrates. At present, the cause of this loss of accuracy remains uncertain. Perhaps there is an error in imposing the exact solutions as boundary conditions.

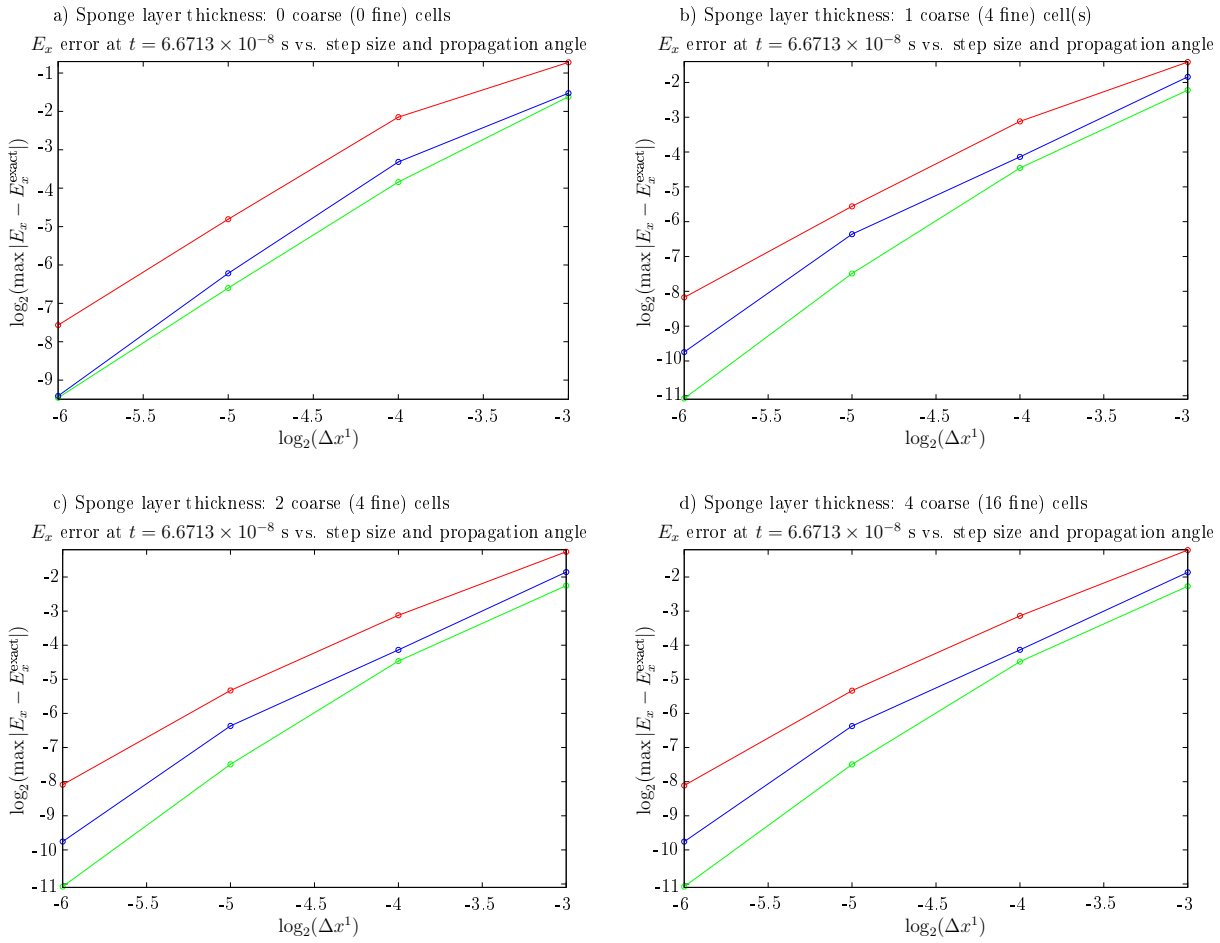

Figure 4.4: Log-log plot of error in E_x vs. step size and propagation angle, with several sponge layer thicknesses, for Eq. (4.17). Propagation angles $\alpha = \pi/6,\ \pi/4$, and $\pi/3$ rad are represented respectively by the red, green, and blue curves.

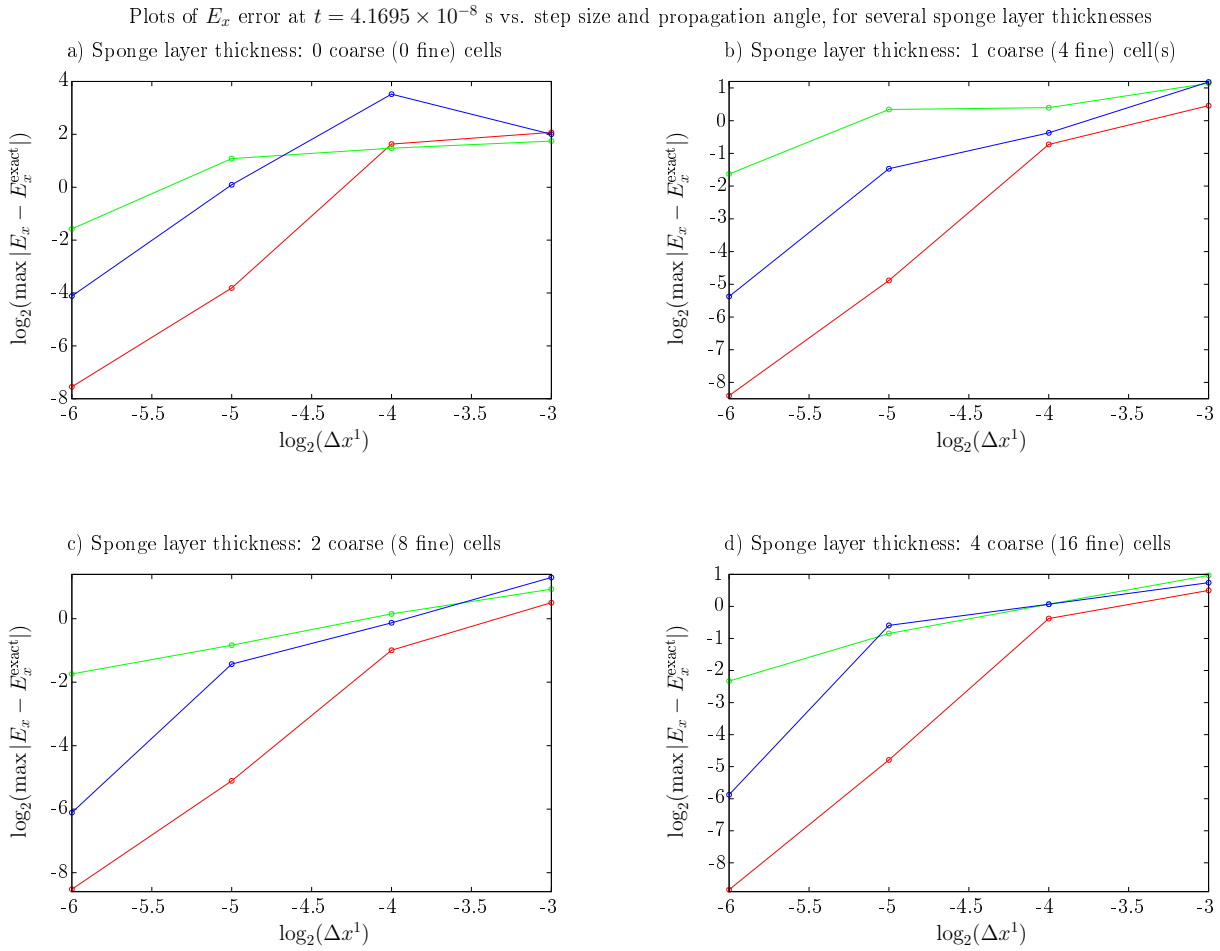

Figure 4.5: Log-log plot of error in E_x vs. step size and propagation angle, with several sponge layer thicknesses, for Eq. (4.15). Wave number pairs $(K_x, K_y) = (1, 1)$, $(1, 2)$, and $(2, 1)$ rad^{-1} m^{-1} are represented respectively by the red, green, and blue curves.

Chapter 5

A divergence-free, cylindrical current source with a non-zero magnetic dipole moment

5.1 Definitions

We consider a cylinder with radius a and height d centered at $(x, y, z) = (x_1, y_1, 0)$ with a current density $\mathbf{J}$ running through it. We wish to design a $\mathbf{J}$ such that $\boldsymbol{\nabla} \cdot \mathbf{J} = 0$ everywhere. Since $\mathbf{J} = 0$ outside the cylinder, it must also equal 0 and be smooth on the cylinder boundaries.

Let us ignore time dependence for the moment, and define

$$\psi_\alpha(x, y, z) = \frac{J_\alpha a}{3\pi} \cos^6\left(\frac{\pi\sqrt{(x - x_1)^2 + (y - y_1)^2}}{2a}\right) \cos^6\left(\frac{\pi z}{d}\right). \tag{5.1}$$

α is an integer index, which will become convenient when we consider sums of currents. J_α is a constant with units of current density. The factor of $a/(3\pi)$ is arbitrary, but it ensures that $\mathbf{J}_\alpha$ has the proper units and results in nice cancellations.

Note that $\cos^6(0) = 1$, $\cos^6(\pm\pi/2) = 0$, and $\cos^6$ is very smooth at its extrema. Thus, given our cylinder, defined by $(x - x_1)^2 + (y - y_1)^2 \leq a^2$ and $z \in [-d/2, d/2]$, $\psi = 0$ and is smooth on the cylinder boundaries.

For our current density $\mathbf{J}_\alpha$ resulting from ψ_α, let us take

$$\begin{aligned} \mathbf{J}_\alpha(x, y, z) &= \boldsymbol{\nabla} \times \psi_\alpha(x, y, z)\boldsymbol{e}_z \\ &= \frac{\partial \psi_\alpha}{\partial y}\boldsymbol{e}_x - \frac{\partial \psi_\alpha}{\partial x}\boldsymbol{e}_y \\ &\equiv J_{\alpha x}(x, y, z)\boldsymbol{e}_x + J_{\alpha y}(x, y, z)\boldsymbol{e}_y \end{aligned} \tag{5.2}$$

inside our cylinder and 0 outside the cylinder. Thus, $\mathbf{J}_\alpha$ is manifestly divergence- free everywhere except perhaps the cylinder boundaries. If, however, $\mathbf{J}_\alpha = 0$ and is smooth

enough on the cylinder boundaries, $\boldsymbol{\nabla} \cdot \mathbf{J}_\alpha = 0$ everywhere. Since $\psi_\alpha = 0$ and is smooth on the cylinder boundaries, we expect the same of $\mathbf{J}_\alpha$, but we should verify it.

Through straightforward (though perhaps tedious) application of the chain rule, we see that

$$J_{\alpha x}(x, y, z) = -J_\alpha \frac{(y - y_1)}{r} \sin\left(\frac{\pi r}{2a}\right) \cos^5\left(\frac{\pi r}{2a}\right) \cos^6\left(\frac{\pi z}{d}\right) \tag{5.3}$$

and

$$J_{\alpha y}(x, y, z) = J_\alpha \frac{(x - x_1)}{r} \sin\left(\frac{\pi r}{2a}\right) \cos^5\left(\frac{\pi r}{2a}\right) \cos^6\left(\frac{\pi z}{d}\right), \tag{5.4}$$

with

$$r \equiv \sqrt{(x - x_1)^2 + (y - y_1)^2}. \tag{5.5}$$

For notational convenience, and ease in computing the magnetic dipole moment later, let us make the substitutions

$$\begin{aligned} x &= x_1 + r\cos\phi \\ y &= y_1 + r\sin\phi, \end{aligned} \tag{5.6}$$

with r still defined by Eq. (5.5). Then

$$J_{\alpha x}(r, \phi, z) = -J_\alpha \sin\phi \sin\left(\frac{\pi r}{2a}\right) \cos^5\left(\frac{\pi r}{2a}\right) \cos^6\left(\frac{\pi z}{d}\right) \tag{5.7}$$

and

$$J_{\alpha y}(r, \phi, z) = J_\alpha \cos\phi \sin\left(\frac{\pi r}{2a}\right) \cos^5\left(\frac{\pi r}{2a}\right) \cos^6\left(\frac{\pi z}{d}\right). \tag{5.8}$$

In these coordinates, the cylinder boundaries are $r = a$, $z = \pm d/2$. We see immediately that

$$\begin{aligned} J_{\alpha x}(r, \phi, z = \pm d/2) &= 0 \\ J_{\alpha y}(r, \phi, z = \pm d/2) &= 0 \\ J_{\alpha x}(r = a, \phi, z) &= 0 \\ J_{\alpha y}(r = a, \phi, z) &= 0 \end{aligned} \tag{5.9}$$

and can verify easily enough that all spatial derivatives are 0 at the cylinder boundaries as well. Thus, $\boldsymbol{\nabla} \cdot \mathbf{J}_\alpha = 0$ everywhere.

We also see that $J_{\alpha x}(r = 0, \phi, z, t) = J_{\alpha y}(r = 0, \phi, z, t) = 0$. In Cartesian coordinates, the equivalent limit (with $(x, y) \to (x_1, y_1)$) is not nearly so straightforward.

If we express $\mathbf{J}_\alpha$ in terms of cylindrical unit vectors $\boldsymbol{e}_r$, $\boldsymbol{e}_\phi$, and $\boldsymbol{e}_z$, we find that

$$J_{\alpha r}(r, \phi, z) = J_{\alpha x}(r, \phi, z)\cos\phi + J_{\alpha y}(r, \phi, z)\sin\phi = 0 \tag{5.10}$$

and

$$\begin{aligned} J_{\alpha\phi}(r, \phi, z) &= -J_{\alpha x}(r, \phi, z)\sin\phi + J_{\alpha y}(r, \phi, z)\cos\phi \\ &= J_\alpha \sin\left(\frac{\pi r}{2a}\right) \cos^5\left(\frac{\pi r}{2a}\right) \cos^6\left(\frac{\pi z}{d}\right). \end{aligned} \tag{5.11}$$

This will be useful in analyzing simulation results later.

5.2 Computing the magnetic dipole moment

For this section, let us drop the index α for convenience. Since our (time-independent) $\mathbf{J}$ is divergence-free, it does not yield an electric dipole moment. We thus expect the strongest radiation term from this current density to be a magnetic dipole term, assuming it exists. We will show that the $\mathbf{J}$ defined in the previous section does, in fact, have a non-zero magnetic dipole moment.

The magnetic dipole moment is given by

$$\boldsymbol{m} = \frac{1}{2}\int_V (\boldsymbol{x} \times \mathbf{J}(\boldsymbol{x}))\; d^3x. \tag{5.12}$$

Note that this is a definite integral over the entire problem volume V. $\boldsymbol{m}$ is a single vector, not a vector field. Since our $\mathbf{J}$ only takes on non-zero values inside the cylinder, we may use the cylinder as our volume of integration in computing $\boldsymbol{m}$.

We see immediately that the $x-$ and $y-$components of $\boldsymbol{m}$, namely m_x and m_y, both equal 0. This is because

$$m_x,\; m_y \propto \int_{-d/2}^{d/2} z\cos^6\left(\frac{\pi z}{d}\right) dz. \tag{5.13}$$

The integrand is odd in z, while the range of integration is symmetric in z. Thus, $m_x = m_y = 0$.

By definition,

$$m_z = \frac{1}{2}\int_V (xJ_y - yJ_x)\, d^3x. \tag{5.14}$$

From Eqs. (5.6) – (5.8), we see that

$$\begin{aligned} m_z =& \frac{J_1}{2}\int_0^{2\pi}\int_0^a (r + x_1\cos\phi + y_1\sin\phi)\sin\left(\frac{\pi r}{2a}\right)\cos^5\left(\frac{\pi r}{2a}\right) r\; dr d\phi \\ &\times \int_{-d/2}^{d/2}\cos^6\left(\frac{\pi z}{d}\right) dz. \end{aligned} \tag{5.15}$$

The terms in Eq. (5.15) proportional to $\cos\phi$ and $\sin\phi$ do not contribute to m_z, since $\int_0^{2\pi}\cos\phi\; d\phi = \int_0^{2\pi}\sin\phi\; d\phi = 0$. Thus,

$$\begin{aligned} m_z &= \frac{J_1}{2}\cos(2\pi\nu t)\int_0^{2\pi} d\phi\int_0^a r^2\sin\left(\frac{\pi r}{2a}\right)\cos^5\left(\frac{\pi r}{2a}\right) dr\int_{-d/2}^{d/2}\cos^6\left(\frac{\pi z}{d}\right) dz \\ &= \frac{J_1}{2}2\pi\frac{(45\pi^2 - 272)a^3}{432\pi^3}\frac{5d}{16} \\ &= \frac{5(45\pi^2 - 272)}{6912\pi^2}J_1 a^3 d \\ &\approx 0.0126\; J_1 a^3 d, \end{aligned} \tag{5.16}$$

according to Mathematica.

Note that the magnetic dipole moment is independent of the coordinates of the center of the cylinder.

5.3 Current density in two dimensions

To reduce computational time and aid in visualization, we also consider a 2D analogue of the current density described in the previous sections. Instead of a cylinder, our current runs through a circle of radius r centered at $(x, y) = (x_1, y_1)$. If we define

$$\psi_\alpha(x, y) = \frac{J_\alpha a}{3\pi} \cos^6\left(\frac{\pi\sqrt{(x - x_1)^2 + (y - y_1)^2}}{2a}\right) \tag{5.17}$$

and insert Eq. (5.17) into Eq. (5.2) (treating $\boldsymbol{e}_z$ as a scalar) we obtain

$$J_{\alpha x}(x, y) = -J_\alpha \frac{(y - y_1)}{r} \sin\left(\frac{\pi r}{2a}\right) \cos^5\left(\frac{\pi r}{2a}\right) \tag{5.18}$$

and

$$J_{\alpha y}(x, y) = J_\alpha \frac{(x - x_1)}{r} \sin\left(\frac{\pi r}{2a}\right) \cos^5\left(\frac{\pi r}{2a}\right), \tag{5.19}$$

with r given by Eq. (5.5). Equivalently, we have

$$J_{\alpha x}(r, \phi) = -J_\alpha \sin\phi \sin\left(\frac{\pi r}{2a}\right) \cos^5\left(\frac{\pi r}{2a}\right) \tag{5.20}$$

and

$$J_{\alpha y}(r, \phi) = J_\alpha \cos\phi \sin\left(\frac{\pi r}{2a}\right) \cos^5\left(\frac{\pi r}{2a}\right) \tag{5.21}$$

for $r \in [0, a]$. Eqs. (5.20) and (5.21) can in turn be compressed into

$$J_{\alpha\phi}(r, \phi) = J_\alpha \sin\left(\frac{\pi r}{2a}\right) \cos^5\left(\frac{\pi r}{2a}\right). \tag{5.22}$$

5.4 Artificial dissipation

Consider $\mathbf{J}(x, y, t) = \mathbf{J}_1(x, y) \cos(2\pi\nu t)$ for $t > 0$. In advancing Maxwell's Equations with this $\mathbf{J}$, given that $\nu = 0$ and all EM field components equal 0 at $t = 0$, we observe a non-physical standing wave phenomenon at the center of the source. The maximum standing wave amplitude appears to grow linearly in time and is roughly proportional to the fourth power of the grid spacing. We attempt to mitigate this problem by adding an artificial dissipation term to Maxwell's Equations.

Recall that we are treating the Faraday-Ampère-Maxwell system as a damped hyperbolic conservation system with a source term, i.e.

$$\frac{\partial U^l}{\partial t} + \nabla \cdot F(U^l) = S^l - \sigma \left(U^l - I(U^{l-1}) \right). \tag{5.23}$$

Here,

$$U^l = \begin{pmatrix} \mathbf{B}^l \\ \mathbf{E}_T^l \end{pmatrix}, \tag{5.24}$$

$$F(U^l) = \begin{pmatrix} 0 & -E_{Tz}^l & E_{Ty}^l & 0 & c^2 B_z^l & -c^2 B_y^l \\ E_{Tz}^l & 0 & -E_{Tx}^l & -c^2 B_z^l & 0 & c^2 B_x^l \\ -E_{Ty}^l & E_{Tx}^l & 0 & c^2 B_y^l & -c^2 B_x^l & 0 \end{pmatrix}, \tag{5.25}$$

and

$$S^l = \begin{pmatrix} \mathbf{0} \\ -\dfrac{\mathbb{P}(\mathbf{J}^l)}{\epsilon_0} \end{pmatrix} \tag{5.26}$$

We will concern ourselves with dissipation terms of the form

$$a_3 c h^3 \sum_d \frac{\partial^4 U}{\partial x_d^4} \tag{5.27}$$

and

$$a_5 c h^5 \sum_d \frac{\partial^6 U}{\partial x_d^6}. \tag{5.28}$$

In theory, adding either Eq. (5.27) or Eq. (5.28) to Eq. (5.23) should damp our non-physical standing waves. Here, a_3 and a_5 are dimensionless constants, and the factors of c are inserted for dimensionality reasons. These will become clearer shortly.

5.4.1 Adding artificial dissipation terms to flux

Recall from Sec. 2.5.2 that we approximate $\nabla \cdot F$ as

$$(D \cdot F)_{\boldsymbol{i}} = \frac{1}{h} \sum_d \left(\langle F^d \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}_d} - \langle F^d \rangle_{\boldsymbol{i}-\frac{1}{2}\boldsymbol{e}_d} \right) \tag{5.29}$$

Thus, the replacements

$$\langle F^d \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}_d} \to \langle F^d \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}_d} + a_3 c h^3 \left\langle \frac{\partial^3 U}{\partial x_d^3} \right\rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}_d} \tag{5.30}$$

and

$$\langle F^d \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}_d} \to \langle F^d \rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}_d} + a_5 c h^5 \left\langle \frac{\partial^5 U}{\partial x_d^5} \right\rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}_d} \tag{5.31}$$

are equivalent to adding either Eq. (5.27) or Eq. (5.28) to Eq. (5.23), respectively.

To calculate the artificial dissipation terms, we recall that face-averaged quantities can be defined from cell-averaged quantities as follows:

$$h^{n+1}\left\langle \frac{\partial^{n+1}U}{\partial x_d^{n+1}}\right\rangle_{\boldsymbol{i}} = h^n\left\langle \frac{\partial^n U}{\partial x_d^n}\right\rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}_d} - h^n\left\langle \frac{\partial^n U}{\partial x_d^n}\right\rangle_{\boldsymbol{i}-\frac{1}{2}\boldsymbol{e}_d}, \tag{5.32}$$

and that

$$Af(x+ah) = A\sum_{n=0}^{\infty}\frac{f^{(n)}(x)}{n!}h^n a^n. \tag{5.33}$$

Employing the properties of even symmetry, we see that

$$\frac{d^4 f(x)}{dx^4}h^4 = Af(x) + B\left(f(x+h)+f(x-h)\right) + C\left(f(x+2h)+f(x-2h)\right) + \mathcal{O}(h^6). \tag{5.34}$$

From Eq. (5.33), we must have

$$\begin{pmatrix} 1 & 2 & 2 \\ 0 & 1 & 4 \\ 0 & 1 & 16 \end{pmatrix}\begin{pmatrix} A \\ B \\ C \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 12 \end{pmatrix} \tag{5.35}$$

in order for Eq. (5.35) to hold. This is solved easily for $A = 6$, $B = -4$, $C = 1$. Thus, to the desired accuracy,

$$h^4\left\langle \frac{\partial^4 U}{\partial x_d^4}\right\rangle_{\boldsymbol{i}} = \langle U\rangle_{\boldsymbol{i}+2\boldsymbol{e}_d} - 4\langle U\rangle_{\boldsymbol{i}+\boldsymbol{e}_d} + 6\langle U\rangle_{\boldsymbol{i}} - 4\langle U\rangle_{\boldsymbol{i}-\boldsymbol{e}_d} + \langle U\rangle_{\boldsymbol{i}-2\boldsymbol{e}_d}. \tag{5.36}$$

Eqs. (5.36) and (5.32) imply that

$$h^3\left\langle \frac{\partial^3 U}{\partial x_d^3}\right\rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}_d} = \langle U\rangle_{\boldsymbol{i}+2\boldsymbol{e}_d} - 3\langle U\rangle_{\boldsymbol{i}+\boldsymbol{e}_d} + 3\langle U\rangle_{\boldsymbol{i}} - \langle U\rangle_{\boldsymbol{i}-\boldsymbol{e}_d}. \tag{5.37}$$

Using an evenly symmetric 7-point stencil to approximate $\frac{d^6 f(x)}{dx^6}h^6$, we see that

$$\begin{pmatrix} 1 & 2 & 2 & 2 \\ 0 & 1 & 4 & 9 \\ 0 & 1 & 16 & 81 \\ 0 & 1 & 64 & 729 \end{pmatrix}\begin{pmatrix} A \\ B \\ C \\ D \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 360 \end{pmatrix}. \tag{5.38}$$

Here, A, B, and C are defined as in Eq. (5.34) (though they will not take on the same values) and D is the coefficient of $(f(x+3h)+f(x-3h))$. Eq. (5.38) is solved easily for $A = -20$, $B = 15$, $C = -6$, $D = 1$. Thus,

$$\begin{aligned} h^6\left\langle \frac{\partial^6 U}{\partial x_d^6}\right\rangle_{\boldsymbol{i}} = \quad & \langle U\rangle_{\boldsymbol{i}+3\boldsymbol{e}_d} - 6\langle U\rangle_{\boldsymbol{i}+2\boldsymbol{e}_d} + 15\langle U\rangle_{\boldsymbol{i}+\boldsymbol{e}_d} - 20\langle U\rangle_{\boldsymbol{i}} \\ & + \langle U\rangle_{\boldsymbol{i}-3\boldsymbol{e}_d} - 6\langle U\rangle_{\boldsymbol{i}-2\boldsymbol{e}_d} + 15\langle U\rangle_{\boldsymbol{i}-\boldsymbol{e}_d} \end{aligned} \tag{5.39}$$

and

$$h^5 \left\langle \frac{\partial^5 U}{\partial x_d^5} \right\rangle_{\boldsymbol{i}+\frac{1}{2}\boldsymbol{e}_d} = \langle U\rangle_{\boldsymbol{i}+3\boldsymbol{e}_d} - 5\langle U\rangle_{\boldsymbol{i}+2\boldsymbol{e}_d} + 10\langle U\rangle_{\boldsymbol{i}+\boldsymbol{e}_d} - \langle U\rangle_{\boldsymbol{i}-2\boldsymbol{e}_d} + 5\langle U\rangle_{\boldsymbol{i}-\boldsymbol{e}_d} - 10\langle U\rangle_{\boldsymbol{i}}. \tag{5.40}$$

5.4.2 Characteristic and stability analysis of artificial dissipation terms

If we again define $\mathcal{U}$ as in Eq. (2.65), i.e.

$$\mathcal{U} \equiv U^l - I(U^{l-1}) \tag{5.41}$$

then the source-free FAM system with damping and fourth-derivative artificial dissipation can be expressed as

$$\frac{\partial \mathcal{U}}{\partial t} + \nabla \cdot F(\mathcal{U}) + \sigma \mathcal{U} + a_3 c h^3 \sum_d \frac{\partial^4 \mathcal{U}}{\partial x_d^4} = 0, \tag{5.42}$$

while the equivalent system with sixth-derivative artificial dissipation can be expressed as

$$\frac{\partial \mathcal{U}}{\partial t} + \nabla \cdot F(\mathcal{U}) + \sigma \mathcal{U} + a_5 c h^5 \sum_d \frac{\partial^6 \mathcal{U}}{\partial x_d^6} = 0. \tag{5.43}$$

Using Eq. (2.69) and Eq. (5.36), and assuming periodic boundary conditions, we have

$$\begin{aligned}
\left.\frac{\partial^4 \mathcal{U}}{\partial x^4}\right|_{(j_x,j_y,j_z)} &= \sum_{k_x,k_y,k_z} \frac{\hat{\mathcal{U}}_{(k_x,k_y,k_z)}}{\Delta x^4}\, e^{2\pi i k_y j_y \Delta y/L_y}\, e^{2\pi i k_z j_z \Delta z/L_z} \\
&\qquad \times \big[6e^{2\pi i k_x j_x \Delta x/L_x} \\
&\qquad\quad -4\left(e^{2\pi i k_x (j_x+1)\Delta x/L_x} + e^{2\pi i k_x (j_x-1)\Delta x/L_x}\right) \\
&\qquad\quad +\ \left(e^{2\pi i k_x (j_x+2)\Delta x/L_x} + e^{2\pi i k_x (j_x-2)\Delta x/L_x}\right)\big] \\
&= \sum_{k_x,k_y,k_z} \hat{\mathcal{U}}_{(k_x,k_y,k_z)} e^{2\pi i k_x j_x \Delta x/L_x} e^{2\pi i k_y j_y \Delta y/L_y} e^{2\pi i k_z j_z \Delta z/L_z} \\
&\qquad \times \frac{1}{\Delta x^4}\left[6 - 8\cos\left(\frac{2\pi k_x \Delta x}{L_x}\right) + 2\cos\left(\frac{4\pi k_x \Delta x}{L_x}\right)\right] \\
&= \sum_{k_x,k_y,k_z} \Omega_x^{(4)} \hat{\mathcal{U}}_{(k_x,k_y,k_z)} e^{2\pi i k_x j_x \Delta x/L_x} e^{2\pi i k_y j_y \Delta y/L_y} e^{2\pi i k_z j_z \Delta z/L_z},
\end{aligned} \tag{5.44}$$

with

$$\Omega_x^{(4)} \equiv \frac{1}{\Delta x^4}\left[6 - 8\cos\left(\frac{2\pi k_x \Delta x}{L_x}\right) + 2\cos\left(\frac{4\pi k_x \Delta x}{L_x}\right)\right]. \tag{5.45}$$

If we define $\Omega_y^{(4)}$ and $\Omega_z^{(4)}$ similarly, the DFTs of $\frac{\partial^4 \mathcal{U}}{\partial y^4}$ and $\frac{\partial^4 \mathcal{U}}{\partial z^4}$ are analogous to Eq. (5.44).

Combining Eq. (2.69) and Eq. (5.39) with the analysis in Eq. (5.44) shows that

$$\left.\frac{\partial^6 \mathcal{U}}{\partial x^6}\right|_{(j_x,j_y,j_z)} = \sum_{k_x,k_y,k_z} \Omega_x^{(6)} \hat{\mathcal{U}}_{(k_x,k_y,k_z)} e^{2\pi i k_x j_x \Delta x/L_x} e^{2\pi i k_y j_y \Delta y/L_y} e^{2\pi i k_z j_z \Delta z/L_z}, \tag{5.46}$$

with

$$\Omega_x^{(6)} \equiv \frac{1}{\Delta x^4}\left[-20 + 30\cos\left(\frac{2\pi k_x \Delta x}{L_x}\right) - 12\cos\left(\frac{4\pi k_x \Delta x}{L_x}\right) + 2\cos\left(\frac{6\pi k_x \Delta x}{L_x}\right)\right]. \tag{5.47}$$

If we define $\Omega_y^{(6)}$ and $\Omega_z^{(6)}$ similarly, the DFTs of $\frac{\partial^6 \mathcal{U}}{\partial y^6}$ and $\frac{\partial^6 \mathcal{U}}{\partial z^6}$ are analogous to Eq. (5.46).

Using the 4th-order discretizations established in this section and earlier, the DFT of the source-free Faraday-Ampère-Maxwell system with damping and dissipation can be expressed as

$$\frac{d}{dt}\begin{pmatrix} \hat{\mathcal{B}}_x \\ \hat{\mathcal{B}}_y \\ \hat{\mathcal{B}}_z \\ \hat{\mathcal{E}}_x \\ \hat{\mathcal{E}}_y \\ \hat{\mathcal{E}}_z \end{pmatrix} = -\begin{pmatrix} \sigma_*^{(4,6)}\mathbb{I} & \begin{matrix} 0 & -i\Omega_z & i\Omega_y \\ i\Omega_z & 0 & -i\Omega_x \\ -i\Omega_y & i\Omega_x & 0 \end{matrix} \\ \begin{matrix} 0 & ic^2\Omega_z & -ic^2\Omega_y \\ -ic^2\Omega_z & 0 & ic^2\Omega_x \\ ic^2\Omega_y & -ic^2\Omega_x & 0 \end{matrix} & \sigma_*^{(4,6)}\mathbb{I} \end{pmatrix} \cdot \begin{pmatrix} \hat{\mathcal{B}}_x \\ \hat{\mathcal{B}}_y \\ \hat{\mathcal{B}}_z \\ \hat{\mathcal{E}}_x \\ \hat{\mathcal{E}}_y \\ \hat{\mathcal{E}}_z \end{pmatrix}. \tag{5.48}$$

Here, $\sigma_*^{(4,6)}$ is an abbreviated notation meaning either $\sigma_*^{(4)}$ or $\sigma_*^{(6)}$, depending on the type of artificial dissipation employed. In turn,

$$\sigma_*^{(4)} \equiv \sigma + a_3 c h^3 \left(\Omega_x^{(4)} + \Omega_y^{(4)} + \Omega_z^{(4)}\right) \tag{5.49}$$

and

$$\sigma_*^{(6)} \equiv \sigma + a_5 c h^5 \left(\Omega_x^{(6)} + \Omega_y^{(6)} + \Omega_z^{(6)}\right), \tag{5.50}$$

assuming that $h = \Delta x = \Delta y = \Delta z$, as is standard in Chombo simulations.

Following the logic of Sec. 2.6.2, we see that the matrix on the right side of Eq. (5.48) has the following eigenvalues, each with multiplicity 2: $-\sigma_*^{(4,6)} - ic\sqrt{\Omega_x^2 + \Omega_y^2 + \Omega_z^2}$, $-\sigma_*^{(4,6)}$, $-\sigma_*^{(4,6)} + ic\sqrt{\Omega_x^2 + \Omega_y^2 + \Omega_z^2}$.

5.4.3 Choosing artificial dissipation coefficients

Ideally, we would choose a_3 and/or a_5 so as to minimize the modulus of the RK4 stability function R, subject to grid spacings, time step, problem domain dimensions, and Fourier nodes. However, since Chombo does not provide an easy means of spectral decomposition, we are forced to use more *ad hoc* means. Ignoring the imaginary axis in the stability plot of RK4, and assuming no damping, we semi-arbitrarily chose $\left.\sigma_*^{(4,6)}\Delta t\right|_{\max} = 2$. Then, using the facts that, for $n \in \mathbb{Z}$, $\cos(2n\pi) = 1$, while $\cos((2n+1)\pi) = -1$, we obtained

$$\left.\sigma_*^{(4)}\Delta t\right|_{\max} = a_3 c h^3 D \frac{16}{h^4}\Delta t = 16(\mathrm{CFL})\mathbf{D} a_3, \tag{5.51}$$

where $\mathbf{D}$ is the number of spatial dimensions and the CFL number is given by $c\Delta t/h$. Similarly,

$$\sigma_*^{(6)}\Delta t\Big|_{\max} = -a_5 c h^5 \mathbf{D}\frac{64}{h^6}\Delta t = -64(\mathrm{CFL})\mathbf{D}a_5. \tag{5.52}$$

Since we have chosen $\sigma_*^{(4,6)}\Delta t\Big|_{\max} = 2$, Eqs. (5.51) and (5.52) yield

$$a_3 = \frac{2}{16(\mathrm{CFL})\mathbf{D}} \tag{5.53}$$

and

$$a_5 = -\frac{2}{64(\mathrm{CFL})\mathbf{D}}. \tag{5.54}$$

These are the artificial dissipation coefficients used in the test problems described in the subsequent sections.

5.5 Low-frequency simulation results in 2D

For this battery of simulations, we consider

$$\mathbf{J}(x, y, t) = \mathbf{J}_1(x, y)\sin(2\pi\nu_1 t) \tag{5.55}$$

for $t \geq 0$ and all field components equal to 0 at $t = 0$.

The relevant parameters are summarized below:

$$\begin{aligned}
J_1 &= 1\ \mathrm{Cm}^{-2}\mathrm{s}^{-1}\ (\mathrm{V/m}) \\
a &= \frac{1}{2}\ \mathrm{m} \\
x_1 &= 0\ \mathrm{m} \\
y_1 &= 0\ \mathrm{m} \\
\nu_1 &= 149896229\ \mathrm{s}^{-1}\ (\mathrm{Hz}) \\
h^0 &= \frac{1}{4},\ \frac{1}{8},\ \frac{1}{16},\ \frac{1}{32},\ \frac{1}{64},\ \frac{1}{128}\ \mathrm{m} \\
h^1 &= \frac{1}{16},\ \frac{1}{32},\ \frac{1}{64},\ \frac{1}{128},\ \frac{1}{256},\ \frac{1}{512}\ \mathrm{m} \\
\mathrm{CFL} &= \frac{1}{2} \\
\Omega^0 &= [-10, 10]\ \mathrm{m} \\
\Omega^1 &= [-2, 2]\ \mathrm{m} \\
\text{nesting depth} &= 0,\ 2 \text{ coarse cells} \\
\text{ramping depth} &= 0,\ 2 \text{ coarse cells}
\end{aligned} \tag{5.56}$$

These simulations are run both with and without 6th-derivative artificial dissipation.

ν_1 was specifically chosen as $\nu_{\max}/16$ for $h^0 = 1/4$ m, where

$$\nu_{\max} \equiv \frac{c}{(\text{CFL})h^0}, \tag{5.57}$$

and, as usual, $\text{CFL} = \frac{c\Delta t^l}{\Delta x^l}$. As such, we expect the oscillations in $\mathbf{J}$ to be quite well resolved at all levels of refinement considered in Eq. (5.56), and, since CFL = 1/2, that the spatial period of the oscillations in $\mathbf{J}$ will equal $8h^0$ for $h^0 = 1/4$ m, i.e. the period will equal 2 m.

5.5.1 Two-level, 4th-order simulations: field components

For low resolutions ($h^0 = 1/16$ m and higher) the simulations converge much more slowly than the expected, desired 4th order. At these resolutions, the differences between simulation results at a coarse h^0 and the next finer h^0 are due to high-frequency modes, whose amplitudes decrease as h^0 decreases. Both damping and artificial dissipation do a decent job at damping the high-frequency modes on the fine patch Ω^1, with artificial dissipation performing somewhat better in this regard. However, for $h^0 = 1/32$ m and lower, the high-frequency modes have all but disappeared, and the code begins to converge as desired.

The convergence properties for B_z and E_x resulting from two-level simulations are summarized in the tables below. The differences (given in columns with labels of the form "$1/2^n$ to $1/2^{n+1}$") are between data with $h^0 = 1/2^n$ m and data with $h^0 = 1/2^{n+1}$ m which has been averaged down to $h^0 = 1/2^n$ m. In other words, we have Richardson estimates of the errors in B_z and E_x at all but the finest resolution, since we don't know the exact solutions.

Table 5.1: Lower-resolution convergence properties of B_z after two periods of $\mathbf{J}$ in two-level simulations without damping and without artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in Tesla (T) = kg s^{-1} C^{-1}.

Norm	1/4 to 1/8	Rate	1/8 to 1/16	Rate	1/16 to 1/32
L_1	2.7333×10^{-8}	1.3956	1.0389×10^{-8}	2.8447	1.4462×10^{-9}
L_2	4.4103×10^{-9}	1.2933	1.7995×10^{-9}	2.4054	3.3966×10^{-10}
L_∞	2.1404×10^{-9}	1.5311	7.4059×10^{-10}	1.9639	1.8984×10^{-10}

Table 5.2: Higher-resolution convergence properties of B_z after two periods of $\mathbf{J}$ in two-level simulations without damping and without artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in Tesla (T) = kg s^{-1} C^{-1}.

Norm	1/16 to 1/32	Rate	1/32 to /64	Rate	1/64 to 1/128
L_1	1.4462×10^{-9}	3.9939	9.0771×10^{-11}	3.9668	5.8052×10^{-12}
L_2	3.3966×10^{-10}	3.7785	2.4752×10^{-11}	3.9595	1.5910×10^{-12}
L_∞	1.8984×10^{-10}	3.5385	1.6338×10^{-11}	3.9691	1.0432×10^{-12}

Table 5.3: Lower-resolution convergence properties of E_x after two periods of **J** in two-level simulations without damping and without artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in V m^{-1} = kg m s^{-2} C^{-1}.

Norm	1/4 to 1/8	Rate	1/8 to 1/16	Rate	1/16 to 1/32
L_1	5.0749×10^{0}	1.3321	2.0157×10^{0}	2.9311	2.6428×10^{-1}
L_2	8.9164×10^{-1}	1.2059	3.8653×10^{-1}	2.4245	7.2001×10^{-2}
L_∞	5.3050×10^{-1}	1.2651	2.2073×10^{-1}	1.9624	5.6639×10^{-2}

Table 5.4: Higher-resolution convergence properties of E_x after two periods of **J** in two-level simulations without damping and without artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in V m^{-1} = kg m s^{-2} C^{-1}.

Norm	1/16 to 1/32	Rate	1/32 to /64	Rate	1/64 to 1/128
L_1	2.6428×10^{-1}	3.9914	1.6616×10^{-2}	3.9679	1.0619×10^{-3}
L_2	7.2001×10^{-2}	3.7784	5.2473×10^{-3}	3.9596	3.3727×10^{-4}
L_∞	5.6639×10^{-2}	3.5317	4.8975×10^{-3}	3.9701	3.1251×10^{-4}

Table 5.5: Lower-resolution convergence properties of B_z after two periods of **J** in two-level simulations without damping and with artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in Tesla (T) = kg s^{-1} C^{-1}.

Norm	1/4 to 1/8	Rate	1/8 to 1/16	Rate	1/16 to 1/32
L_1	1.6720×10^{-8}	1.3486	6.5655×10^{-9}	2.3670	1.2727×10^{-9}
L_2	2.7653×10^{-9}	1.0544	1.3315×10^{-9}	2.0717	3.1674×10^{-10}
L_∞	8.9701×10^{-10}	0.65842	5.6832×10^{-10}	1.6878	1.7641×10^{-10}

Table 5.6: Higher-resolution convergence properties of B_z after two periods of **J** in two-level simulations without damping and with artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in Tesla (T) = kg s^{-1} C^{-1}.

Norm	1/16 to 1/32	Rate	1/32 to /64	Rate	1/64 to 1/128
L_1	1.2727×10^{-9}	3.7589	9.4015×10^{-11}	4.0366	5.7286×10^{-12}
L_2	3.1674×10^{-10}	3.5851	2.6393×10^{-11}	4.0242	1.6221×10^{-12}
L_∞	1.7641×10^{-10}	3.3429	1.7387×10^{-11}	4.0256	1.0676×10^{-12}

Table 5.7: Lower-resolution convergence properties of E_x after two periods of **J** in two-level simulations without damping and with artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in V m^{-1} = kg m s^{-2} C^{-1}.

Norm	1/4 to 1/8	Rate	1/8 to 1/16	Rate	1/16 to 1/32
L_1	3.1450×10^{0}	1.3670	1.2193×10^{0}	2.3882	2.3291×10^{-1}
L_2	5.8691×10^{-1}	1.0561	2.8227×10^{-1}	2.0717	6.7146×10^{-2}
L_∞	2.6906×10^{-1}	0.65968	1.7032×10^{-1}	1.6860	5.2935×10^{-2}

Table 5.8: Higher-resolution convergence properties of E_x after two periods of **J** in two-level simulations without damping and with artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in V m^{-1} = kg m s^{-2} C^{-1}.

Norm	1/16 to 1/32	Rate	1/32 to /64	Rate	1/64 to 1/128
L_1	2.3291×10^{-1}	3.7686	1.7089×10^{-2}	4.0366	1.0413×10^{-3}
L_2	6.7146×10^{-2}	3.5850	5.5952×10^{-3}	4.0242	3.4388×10^{-4}
L_∞	5.2935×10^{-2}	3.3439	5.2135×10^{-3}	4.0268	3.1984×10^{-4}

Table 5.9: Lower-resolution convergence properties of B_z after two periods of **J** in two-level simulations with damping and without artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in Tesla (T) = kg s^{-1} C^{-1}.

Norm	1/4 to 1/8	Rate	1/8 to 1/16	Rate	1/16 to 1/32
L_1	2.1488×10^{-8}	1.2019	9.3412×10^{-9}	2.7325	1.4055×10^{-9}
L_2	3.7453×10^{-9}	1.0880	1.7618×10^{-9}	2.3655	3.4188×10^{-10}
L_∞	1.3404×10^{-9}	0.84343	7.4703×10^{-10}	1.9659	1.9122×10^{-10}

Table 5.10: Higher-resolution convergence properties of B_z after two periods of **J** in two-level simulations with damping and without artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in Tesla (T) = kg s^{-1} C^{-1}.

Norm	1/16 to 1/32	Rate	1/32 to /64	Rate	1/64 to 1/128
L_1	1.4055×10^{-9}	3.9879	8.8583×10^{-11}	3.9679	5.6610×10^{-12}
L_2	3.4188×10^{-10}	3.7835	2.4828×10^{-11}	3.9621	1.5930×10^{-12}
L_∞	1.9122×10^{-10}	3.5435	1.6400×10^{-11}	3.9721	1.0450×10^{-12}

Table 5.11: Lower-resolution convergence properties of E_x after two periods of **J** in two-level simulations with damping and without artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in V m^{-1} = kg m s^{-2} C^{-1}.

Norm	1/4 to 1/8	Rate	1/8 to 1/16	Rate	1/16 to 1/32
L_1	3.9822×10^{0}	1.2034	1.7293×10^{0}	2.7593	2.5541×10^{-1}
L_2	7.9387×10^{-1}	1.0882	3.7340×10^{-1}	2.3651	7.2476×10^{-2}
L_∞	3.9927×10^{-1}	0.84362	2.2249×10^{-1}	1.9636	5.7045×10^{-2}

Table 5.12: Higher-resolution convergence properties of E_x after two periods of **J** in two-level simulations with damping and without artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in V m^{-1} = kg m s^{-2} C^{-1}.

Norm	1/16 to 1/32	Rate	1/32 to /64	Rate	1/64 to 1/128
L_1	2.5541×10^{-1}	3.9846	1.6134×10^{-2}	3.9684	1.0307×10^{-3}
L_2	7.2476×10^{-2}	3.7834	5.2634×10^{-3}	3.9621	3.3771×10^{-4}
L_∞	5.7045×10^{-2}	3.5367	4.9156×10^{-3}	3.9730	3.1303×10^{-4}

Table 5.13: Lower-resolution convergence properties of B_z after two periods of **J** in two-level simulations with damping and with artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in Tesla (T) = kg s^{-1} C^{-1}.

Norm	1/4 to 1/8	Rate	1/8 to 1/16	Rate	1/16 to 1/32
L_1	1.8072×10^{-8}	1.4081	6.8096×10^{-9}	2.3896	1.2995×10^{-9}
L_2	2.9638×10^{-9}	1.1109	1.3723×10^{-9}	2.0920	3.2189×10^{-10}
L_∞	9.3103×10^{-10}	0.68679	5.7839×10^{-10}	1.6956	1.7856×10^{-10}

Table 5.14: Higher-resolution convergence properties of B_z after two periods of **J** in two-level simulations with damping and with artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in Tesla (T) = kg s^{-1} C^{-1}.

Norm	1/16 to 1/32	Rate	1/32 to /64	Rate	1/64 to 1/128
L_1	1.2995×10^{-9}	3.7775	9.4762×10^{-11}	4.0433	5.7474×10^{-12}
L_2	3.2189×10^{-10}	3.5990	2.6565×10^{-11}	4.0300	1.6262×10^{-12}
L_∞	1.7856×10^{-10}	3.3518	1.7490×10^{-11}	4.0308	1.0700×10^{-12}

Table 5.15: Lower-resolution convergence properties of E_x after two periods of **J** in two-level simulations with damping and with artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in V m^{-1} = kg m s^{-2} C^{-1}.

Norm	1/4 to 1/8	Rate	1/8 to 1/16	Rate	1/16 to 1/32
L_1	3.3921×10^{0}	1.4224	1.2656×10^{0}	2.4110	2.3796×10^{-1}
L_2	6.2896×10^{-1}	1.1123	2.9092×10^{-1}	2.0920	6.8239×10^{-2}
L_∞	2.7943×10^{-1}	0.68897	1.7333×10^{-1}	1.6937	5.3582×10^{-2}

Table 5.16: Higher-resolution convergence properties of E_x after two periods of **J** in two-level simulations with damping and with artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in V m^{-1} = kg m s^{-2} C^{-1}.

Norm	1/16 to 1/32	Rate	1/32 to /64	Rate	1/64 to 1/128
L_1	2.3796×10^{-1}	3.7879	1.7228×10^{-2}	4.0435	1.0448×10^{-3}
L_2	6.8239×10^{-2}	3.5990	5.6316×10^{-3}	4.0300	3.4474×10^{-4}
L_∞	5.3582×10^{-2}	3.3529	5.2445×10^{-3}	4.0321	3.2056×10^{-4}

The convergence properties of B_z and E_x are further illustrated in Fig. 5.1 below. Red curves represent data from simulations with neither damping nor (6th-derivative) artificial dissipation; green curves, no damping but with artificial dissipation; blue curves, damping without dissipation; magenta, both damping and dissipation. The cyan lines depict ideal, 4th-order convergence.

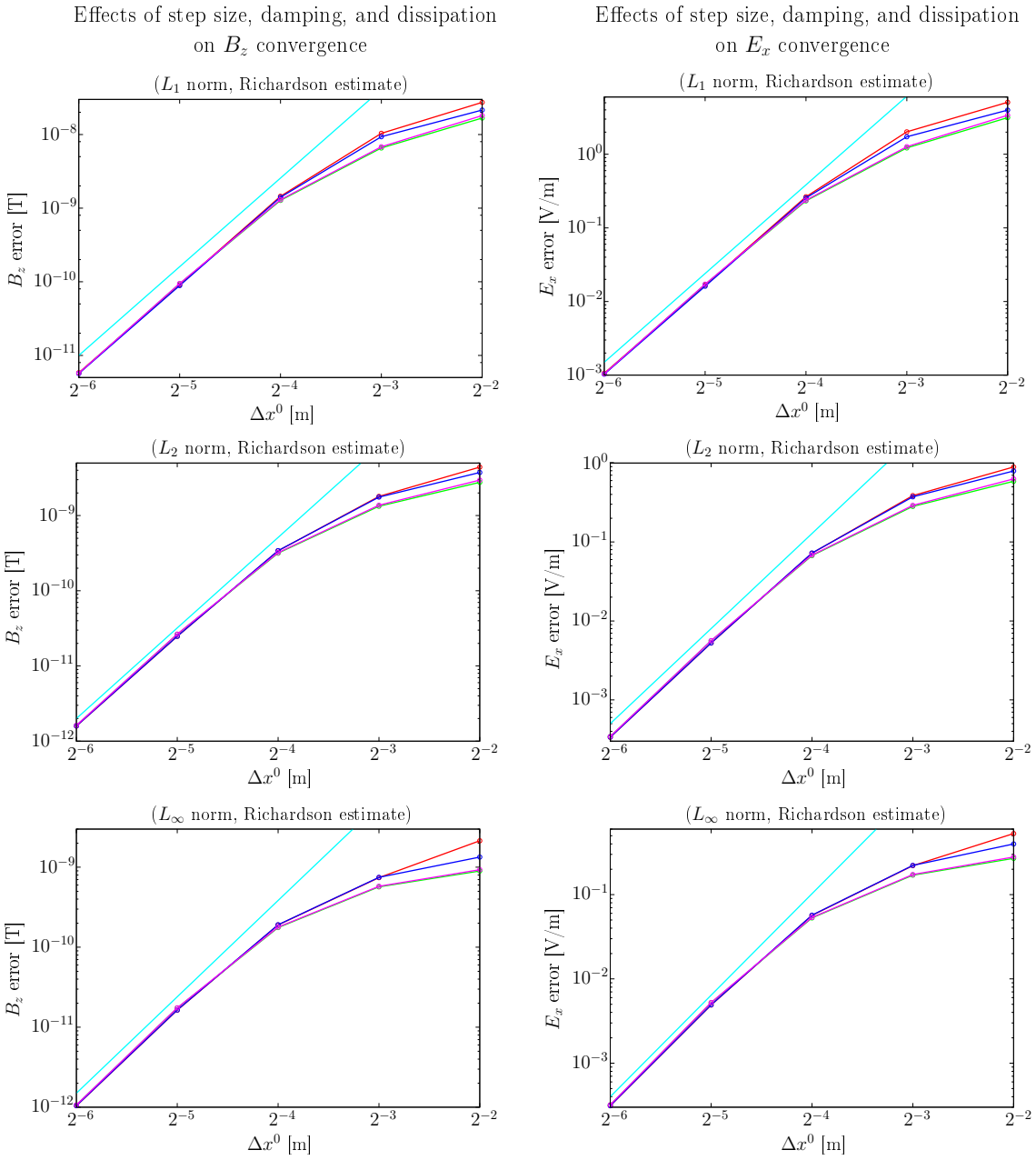

Figure 5.1: Illustration of convergence properties of B_z and E_x for a 4th-order simulation of the Faraday-Ampère-Maxwell system, with $\mathbf{J}$ given by Eq. (5.55) and parameters listed in Eq. (5.56). Red curves represent data from simulations with neither damping nor (6th-derivative) artificial dissipation; green curves, no damping but with artificial dissipation; blue curves, damping without dissipation; magenta, both damping and dissipation. The cyan lines depict ideal, 4th-order convergence.

Given a high enough resolution, we see that our simulations converge with 4th-order

accuracy. Damping and 6th-derivative artificial dissipation both eliminate trapped waves (albeit not perfectly) with artificial dissipation damping the physical, outgoing waves a bit more than the sponge layer terms. Moreover, damping and dissipation decrease the solution error, but also decrease the convergence rate at lower resolutions.

5.5.2 Two-level, 4th-order simulations: $\boldsymbol{\nabla \cdot \mathbf{E}}$

The convergence properties for $\boldsymbol{\nabla \cdot \mathbf{E}}$ errors resulting from two-level simulations are summarized in the tables below. Since our $\mathbf{J}$ is divergence-free, $\boldsymbol{\nabla \cdot \mathbf{E}}$ should equal 0. Thus, we do not need Richardson extrapolation to estimate the convergence of $\boldsymbol{\nabla \cdot \mathbf{E}}$ errors.

Table 5.17: Lower-resolution convergence properties of $\boldsymbol{\nabla \cdot \mathbf{E}}$ after two periods of $\mathbf{J}$ in two-level simulations with no damping and no artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in V m^{-2} = kg s^{-2} C^{-1}.

Norm	1/4	Rate	1/8	Rate	1/16	Rate
L_1	3.2080×10^{-1}	1.6817	9.9996×10^{-2}	5.1031	2.9094×10^{-3}	3.3330
L_2	2.0428×10^{-1}	1.2950	8.3254×10^{-2}	3.9847	5.2588×10^{-3}	2.5452
L_∞	4.7109×10^{-1}	0.38404	3.6099×10^{-1}	3.9210	2.3832×10^{-2}	2.1509

Table 5.18: Higher-resolution convergence properties of $\boldsymbol{\nabla \cdot \mathbf{E}}$ after two periods of $\mathbf{J}$ in two-level simulations with no damping and no artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in V m^{-2} = kg s^{-2} C^{-1}.

Norm	1/32	Rate	1/64	Rate	1/128
L_1	2.8872×10^{-4}	3.0673	3.4444×10^{-5}	3.0263	4.2277×10^{-6}
L_2	9.0094×10^{-4}	2.5096	1.5821×10^{-4}	2.5050	2.7872×10^{-5}
L_∞	5.3664×10^{-3}	2.0039	1.3380×10^{-3}	2.0026	3.3389×10^{-4}

Table 5.19: Lower-resolution convergence properties of $\boldsymbol{\nabla \cdot \mathbf{E}}$ after two periods of $\mathbf{J}$ in two-level simulations with no damping and with artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in V m^{-2} = kg s^{-2} C^{-1}.

Norm	1/4	Rate	1/8	Rate	1/16	Rate
L_1	6.6017×10^{-1}	5.0922	1.9353×10^{-2}	4.9035	6.4661×10^{-4}	4.0156
L_2	6.2952×10^{-1}	5.7523	1.1679×10^{-2}	4.8808	3.9640×10^{-4}	3.6328
L_∞	1.5869×10^{0}	6.0129	2.4574×10^{-2}	4.2879	1.2580×10^{-3}	3.0814

Table 5.20: Higher-resolution convergence properties of $\boldsymbol{\nabla} \cdot \mathbf{E}$ after two periods of $\mathbf{J}$ in two-level simulations with no damping and with artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in V m^{-2} = kg s^{-2} C^{-1}.

Norm	1/32	Rate	1/64	Rate	1/128
L_1	3.9978×10^{-5}	3.8727	2.7292×10^{-6}	3.7431	2.0382×10^{-7}
L_2	3.1956×10^{-5}	3.4730	2.8779×10^{-6}	3.4046	2.7176×10^{-7}
L_∞	1.4862×10^{-4}	3.0146	1.8391×10^{-5}	3.0451	2.2281×10^{-6}

Table 5.21: Lower-resolution convergence properties of $\boldsymbol{\nabla} \cdot \mathbf{E}$ after two periods of $\mathbf{J}$ in two-level simulations with damping and without artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in V m^{-2} = kg s^{-2} C^{-1}.

Norm	1/4	Rate	1/8	Rate	1/16	Rate
L_1	5.3879×10^{-1}	2.3371	1.0663×10^{-1}	5.9166	1.7653×10^{-3}	4.0986
L_2	2.9351×10^{-1}	1.6797	9.1618×10^{-2}	5.9146	1.5188×10^{-3}	3.5786
L_∞	7.4954×10^{-1}	0.99389	3.7636×10^{-1}	6.2746	4.8613×10^{-3}	3.0866

Table 5.22: Higher-resolution convergence properties of $\boldsymbol{\nabla} \cdot \mathbf{E}$ after two periods of $\mathbf{J}$ in two-level simulations with damping and without artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in V m^{-2} = kg s^{-2} C^{-1}.

Norm	1/32	Rate	1/64	Rate	1/128
L_1	1.0304×10^{-4}	4.0249	6.3299×10^{-6}	3.9213	4.1779×10^{-7}
L_2	1.2713×10^{-4}	3.5456	1.0887×10^{-5}	3.4627	9.8751×10^{-7}
L_∞	5.7228×10^{-4}	3.0115	7.0969×10^{-5}	2.4491	1.2996×10^{-5}

Table 5.23: Lower-resolution convergence properties of $\boldsymbol{\nabla} \cdot \mathbf{E}$ after two periods of $\mathbf{J}$ in two-level simulations with damping and with artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in V m^{-2} = kg s^{-2} C^{-1}.

Norm	1/4	Rate	1/8	Rate	1/16	Rate
L_1	7.1129×10^{-1}	5.0637	2.1267×10^{-2}	4.5844	8.8650×10^{-4}	4.1000
L_2	6.3055×10^{-1}	5.7919	1.1381×10^{-2}	4.3202	5.6975×10^{-4}	3.5713
L_∞	1.5869×10^{0}	6.1259	2.2723×10^{-2}	3.7806	1.6534×10^{-3}	2.9632

Table 5.24: Higher-resolution convergence properties of $\boldsymbol{\nabla} \cdot \mathbf{E}$ after two periods of $\mathbf{J}$ in two-level simulations with damping and with artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in V m^{-2} = kg s^{-2} C^{-1}.

Norm	1/32	Rate	1/64	Rate	1/128
L_1	5.1696×10^{-5}	3.9040	3.4534×10^{-6}	3.6906	2.6747×10^{-7}
L_2	4.7930×10^{-5}	3.4810	4.2927×10^{-6}	3.4168	4.0194×10^{-7}
L_∞	2.1202×10^{-4}	2.9397	2.7633×10^{-5}	2.9372	3.6078×10^{-6}

The convergence properties of $\boldsymbol{\nabla} \cdot \mathbf{E}$ are further illustrated in Fig. 5.2 and Fig. 5.3 below. Except for the pink line in the bottom graph of Fig. 5.3 (which shows hypothetical 3rd-order convergence), the colors have the same meanings in Fig. 5.3 as in Fig. 5.1. However, in Fig. 5.2, the red curve depicts the L_1 norm of the error data; the green curve, the L_2 norm; and the blue curve, the L_∞ norm. Plotting Fig. 5.2 in this manner demonstrates that errors in $\boldsymbol{\nabla} \cdot \mathbf{E}$ do not converge to 0 with 4th-order accuracy in the asymptotic regime in simulations that employ neither damping nor dissipation.

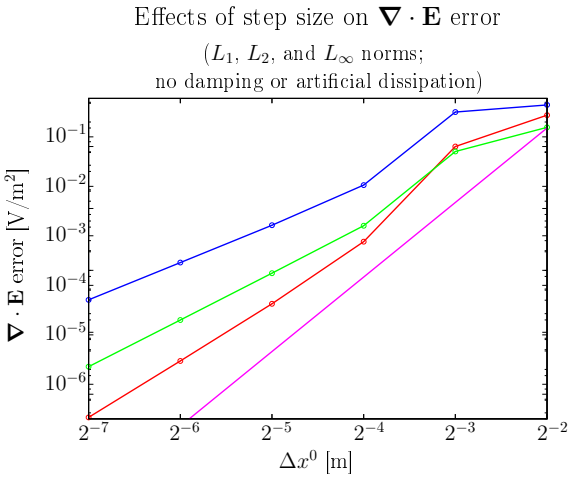

Figure 5.2: Illustration of convergence properties of $\boldsymbol{\nabla} \cdot \mathbf{E}$ errors for a 4th-order simulation of the Faraday-Ampère-Maxwell system, with $\mathbf{J}$ given by Eq. (5.55) and parameters listed in Eq. (5.56). The data plotted here results from simulations with no damping and no artificial dissipation. The red curve depicts the L_1 norm of the error data; the green curve, the L_2 norm; and the blue curve, the L_∞ norm. The magenta line represents ideal 4th-order convergence. Note that the errors in the asymptotic regime do not converge to 0 with 4th-order accuracy.

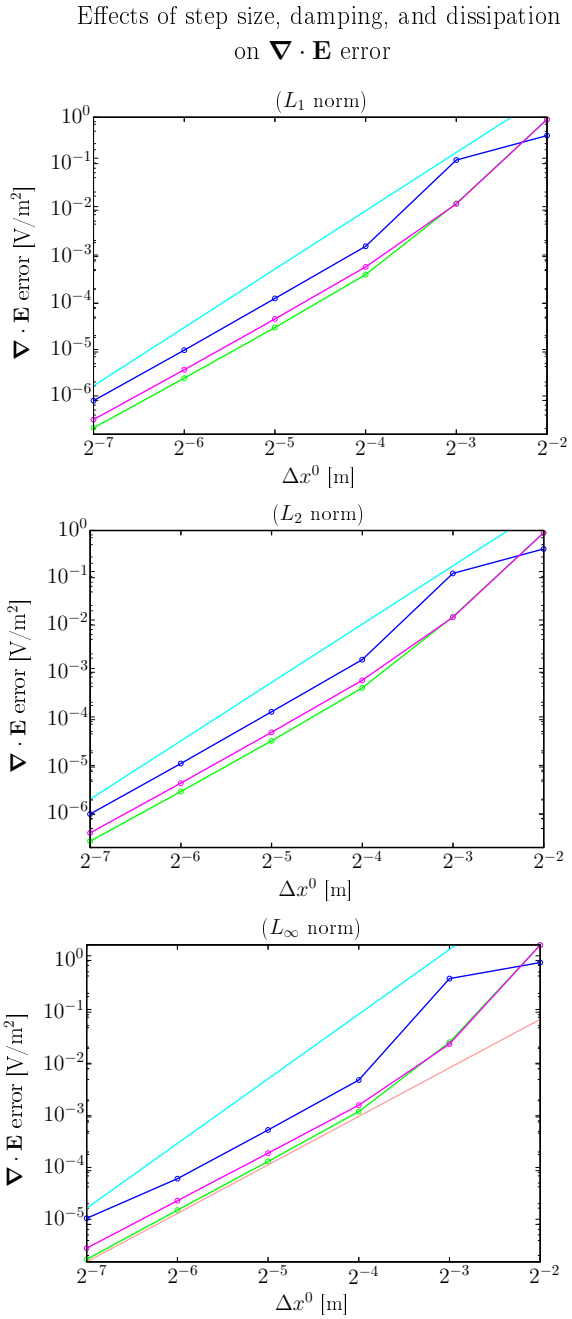

Figure 5.3: Illustration of convergence properties of $\boldsymbol{\nabla} \cdot \mathbf{E}$ errors for a 4th-order simulation of the Faraday-Ampère-Maxwell system, with $\mathbf{J}$ given by Eq. (5.55) and parameters listed in Eq. (5.56). Green curves represent data from simulations with no damping but with artificial dissipation; blue curves, with damping but without dissipation; magenta, with both damping and dissipation. The cyan lines depict ideal, 4th-order convergence. The pink line in the bottom graph depicts 3rd-order convergence.

The error in $\boldsymbol{\nabla} \cdot \mathbf{E}$ does converge to 0, but not quite as well as the solution errors in the field components (see Sec. 5.5.1). For simulations with no damping or dissipation, we do not observe 4th-order convergence in L_1, L_2, or L_∞ norms. For the other types of simulations, we observe roughly 4th-order convergence in L_1 and L_2 norms, but 3rd-order convergence in L_∞.

5.5.3 Two-level, 2nd-order simulations: field components

To gauge the effects of damping and artificial dissipation more precisely, we ran a battery of simulations in which we replaced 4th-order cell averages with their 2nd-order equivalents, effectively downgrading the algorithm from 4th-order to 2nd-order. When adding artificial dissipation, we used 4th-derivative rather than 6th-derivative terms.

The convergence properties for B_z and E_x resulting from two-level simulations are summarized in the tables below. The labels in these tables have the same meanings as in Tabs. 5.1 – 5.16.

Table 5.25: Lower-resolution convergence properties of B_z after two periods of $\mathbf{J}$ in two-level simulations without damping and without artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in Tesla (T) = kg s^{-1} C^{-1}.

Norm	1/4 to 1/8	Rate	1/8 to 1/16	Rate	1/16 to 1/32
L_1	5.1964×10^{-8}	0.96652	2.6592×10^{-8}	1.37481	1.0254×10^{-8}
L_2	8.3336×10^{-9}	0.85480	4.6080×10^{-9}	1.13912	2.0922×10^{-9}
L_∞	3.8382×10^{-9}	1.06261	1.8376×10^{-9}	0.79498	1.0591×10^{-9}

Table 5.26: Higher-resolution convergence properties of B_z after two periods of $\mathbf{J}$ in two-level simulations without damping and without artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in Tesla (T) = kg s^{-1} C^{-1}.

Norm	1/16 to 1/32	Rate	1/32 to 1/64	Rate	1/64 to 1/128
L_1	1.0254×10^{-8}	2.03659	2.4993×10^{-9}	2.03907	6.0813×10^{-10}
L_2	2.0922×10^{-9}	1.74222	6.2538×10^{-10}	1.99103	1.5732×10^{-10}
L_∞	1.0591×10^{-9}	1.61075	3.4678×10^{-10}	1.86710	9.5061×10^{-11}

Table 5.27: Lower-resolution convergence properties of E_x after two periods of $\mathbf{J}$ in two-level simulations without damping and without artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in V m^{-1} = kg m s^{-2} C^{-1}.

Norm	1/4 to 1/8	Rate	1/8 to 1/16	Rate	1/16 to 1/32
L_1	9.9670×10^{0}	0.97233	5.0800×10^{0}	1.40922	1.9127×10^{0}
L_2	1.7681×10^{0}	0.85446	9.7789×10^{-1}	1.14129	4.4333×10^{-1}
L_∞	8.1588×10^{-1}	0.56796	5.5037×10^{-1}	0.79751	3.1665×10^{-1}

Table 5.28: Higher-resolution convergence properties of E_x after two periods of **J** in two-level simulations without damping and without artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in V m^{-1} = kg m s^{-2} C^{-1}.

Norm	1/16 to 1/32	Rate	1/32 to 1/64	Rate	1/64 to 1/128
L_1	1.9127×10^{0}	2.05240	4.6112×10^{-1}	2.03599	1.1244×10^{-1}
L_2	4.4333×10^{-1}	1.74239	1.3250×10^{-1}	1.99075	3.3338×10^{-2}
L_∞	3.1665×10^{-1}	1.61661	1.0326×10^{-1}	1.85214	2.8601×10^{-2}

Table 5.29: Lower-resolution convergence properties of B_z after two periods of **J** in two-level simulations without damping and with artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in Tesla (T) = kg s^{-1} C^{-1}.

Norm	1/4 to 1/8	Rate	1/8 to 1/16	Rate	1/16 to 1/32
L_1	2.6667×10^{-8}	0.81279	1.5181×10^{-8}	1.15100	6.8362×10^{-9}
L_2	4.2055×10^{-9}	0.56725	2.8383×10^{-9}	0.87714	1.5453×10^{-9}
L_∞	1.5554×10^{-9}	0.48076	1.1146×10^{-9}	0.65813	7.0632×10^{-10}

Table 5.30: Higher-resolution convergence properties of B_z after two periods of **J** in two-level simulations without damping and with artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in Tesla (T) = kg s^{-1} C^{-1}.

Norm	1/16 to 1/32	Rate	1/32 to 1/64	Rate	1/64 to 1/128
L_1	6.8362×10^{-9}	1.60071	2.2540×10^{-9}	1.95496	5.8137×10^{-10}
L_2	1.5453×10^{-9}	1.42824	5.7421×10^{-10}	1.88513	1.5545×10^{-10}
L_∞	7.0632×10^{-10}	1.23377	3.0033×10^{-10}	1.68850	9.3177×10^{-11}

Table 5.31: Lower-resolution convergence properties of E_x after two periods of **J** in two-level simulations without damping and with artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in V m^{-1} = kg m s^{-2} C^{-1}.

Norm	1/4 to 1/8	Rate	1/8 to 1/16	Rate	1/16 to 1/32
L_1	5.0892×10^{0}	0.83952	2.8440×10^{0}	1.16162	1.2713×10^{0}
L_2	9.0414×10^{-1}	0.58556	6.0251×10^{-1}	0.87879	3.2766×10^{-1}
L_∞	4.0424×10^{-1}	0.27965	3.3301×10^{-1}	0.66518	2.1000×10^{-1}

Table 5.32: Higher-resolution convergence properties of E_x after two periods of **J** in two-level simulations without damping and with artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in V m^{-1} = kg m s^{-2} C^{-1}.

Norm	1/16 to 1/32	Rate	1/32 to 1/64	Rate	1/64 to 1/128
L_1	1.2713×10^{0}	1.60905	4.1675×10^{-1}	1.95511	1.0748×10^{-1}
L_2	3.2766×10^{-1}	1.42840	1.2174×10^{-1}	1.88515	3.2957×10^{-2}
L_∞	2.1000×10^{-1}	1.21432	9.0505×10^{-2}	1.69031	2.8044×10^{-2}

Table 5.33: Lower-resolution convergence properties of B_z after two periods of **J** in two-level simulations with damping and without artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in Tesla (T) = kg s^{-1} C^{-1}.

Norm	1/4 to 1/8	Rate	1/8 to 1/16	Rate	1/16 to 1/32
L_1	4.3668×10^{-8}	0.87568	2.3799×10^{-8}	1.27275	9.8497×10^{-9}
L_2	7.8739×10^{-9}	0.80018	4.5218×10^{-9}	1.10438	2.1031×10^{-9}
L_∞	2.5055×10^{-9}	0.41759	1.8758×10^{-9}	0.81719	1.0646×10^{-9}

Table 5.34: Higher-resolution convergence properties of B_z after two periods of **J** in two-level simulations with damping and without artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in Tesla (T) = kg s^{-1} C^{-1}.

Norm	1/16 to 1/32	Rate	1/32 to 1/64	Rate	1/64 to 1/128
L_1	9.8497×10^{-9}	2.03069	2.4106×10^{-9}	2.03929	5.8646×10^{-10}
L_2	2.1031×10^{-9}	1.74817	6.2605×10^{-10}	1.99258	1.5732×10^{-10}
L_∞	1.0646×10^{-9}	1.61809	3.4681×10^{-10}	1.86131	9.5451×10^{-11}

Table 5.35: Lower-resolution convergence properties of E_x after two periods of **J** in two-level simulations with damping and without artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in V m^{-1} = kg m s^{-2} C^{-1}.

Norm	1/4 to 1/8	Rate	1/8 to 1/16	Rate	1/16 to 1/32
L_1	8.2549×10^{0}	0.88769	4.4616×10^{0}	1.29061	1.8238×10^{0}
L_2	1.6795×10^{0}	0.80637	9.6037×10^{-1}	1.10684	4.4591×10^{-1}
L_∞	7.2648×10^{-1}	0.36922	5.6244×10^{-1}	0.82249	3.1804×10^{-1}

Table 5.36: Higher-resolution convergence properties of E_x after two periods of **J** in two-level simulations with damping and without artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in V m^{-1} = kg m s^{-2} C^{-1}.

Norm	1/16 to 1/32	Rate	1/32 to 1/64	Rate	1/64 to 1/128
L_1	1.8238×10^{0}	2.03397	4.4534×10^{-1}	2.03681	1.0853×10^{-1}
L_2	4.4591×10^{-1}	1.74826	1.3273×10^{-1}	1.99256	3.3354×10^{-2}
L_∞	3.1804×10^{-1}	1.62349	1.0322×10^{-1}	1.84645	2.8703×10^{-2}

Table 5.37: Lower-resolution convergence properties of B_z after two periods of **J** in two-level simulations with damping and with artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in Tesla (T) = kg s^{-1} C^{-1}.

Norm	1/4 to 1/8	Rate	1/8 to 1/16	Rate	1/16 to 1/32
L_1	3.0882×10^{-8}	0.94078	1.6088×10^{-8}	1.19355	7.0341×10^{-9}
L_2	4.9408×10^{-9}	0.72272	2.9939×10^{-9}	0.91946	1.5829×10^{-9}
L_∞	1.5702×10^{-9}	0.42950	1.1659×10^{-9}	0.69271	7.2133×10^{-10}

Table 5.38: Higher-resolution convergence properties of B_z after two periods of **J** in two-level simulations with damping and with artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in Tesla (T) = kg s^{-1} C^{-1}.

Norm	1/16 to 1/32	Rate	1/32 to 1/64	Rate	1/64 to 1/128
L_1	7.0341×10^{-9}	1.62211	2.2851×10^{-9}	1.96538	5.8515×10^{-10}
L_2	1.5829×10^{-9}	1.44591	5.8102×10^{-10}	1.89409	1.5632×10^{-10}
L_∞	7.2133×10^{-10}	1.25296	3.0266×10^{-10}	1.69363	9.3567×10^{-11}

Table 5.39: Lower-resolution convergence properties of E_x after two periods of **J** in two-level simulations with damping and with artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in V m^{-1} = kg m s^{-2} C^{-1}.

Norm	1/4 to 1/8	Rate	1/8 to 1/16	Rate	1/16 to 1/32
L_1	5.9007×10^{0}	0.96978	3.0128×10^{0}	1.20209	1.3095×10^{0}
L_2	1.0575×10^{0}	0.73476	6.3547×10^{-1}	0.92095	3.3563×10^{-1}
L_∞	4.4275×10^{-1}	0.34620	3.4829×10^{-1}	0.70234	2.1405×10^{-1}

Table 5.40: Higher-resolution convergence properties of E_x after two periods of $\mathbf{J}$ in two-level simulations with damping and with artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in V m^{-1} = kg m s^{-2} C^{-1}.

Norm	1/16 to 1/32	Rate	1/32 to 1/64	Rate	1/64 to 1/128
L_1	1.3095×10^{0}	1.63158	4.2262×10^{-1}	1.96579	1.0819×10^{-1}
L_2	3.3563×10^{-1}	1.44610	1.2318×10^{-1}	1.89408	3.3141×10^{-2}
L_∞	2.1405×10^{-1}	1.23068	9.1210×10^{-2}	1.69544	2.8162×10^{-2}

The convergence properties of B_z and E_x are further illustrated in Fig. 5.4 below. The cyan lines depict 2nd-order convergence. The other curves have the same meanings as in Fig. 5.1

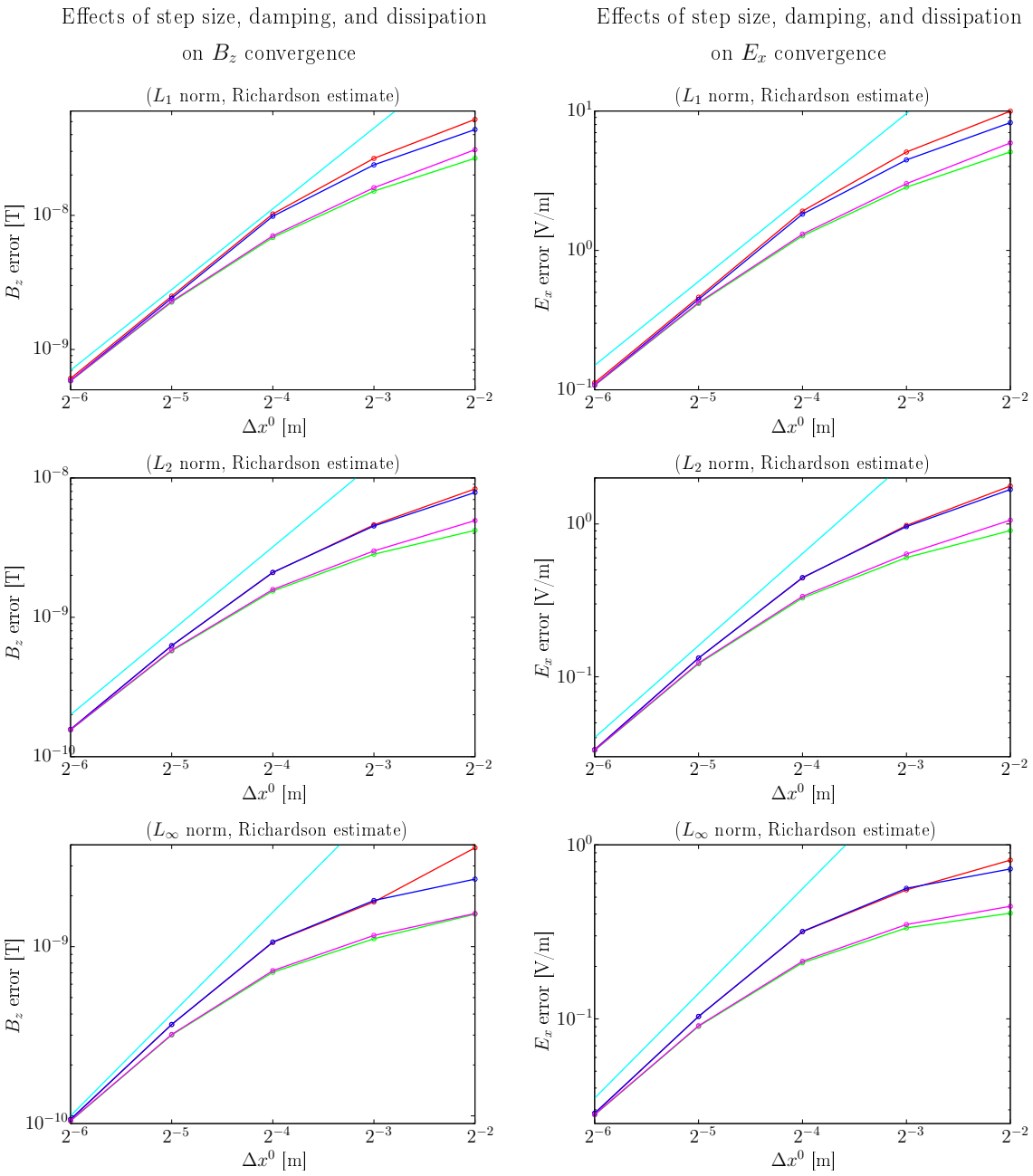

Figure 5.4: Illustration of convergence properties of B_z and E_x for a 2nd-order simulation of the Faraday-Ampère-Maxwell system, with $\mathbf{J}$ given by Eq. (5.55) and parameters listed in Eq. (5.56). Red curves represent data from simulations with neither damping nor (6th-derivative) artificial dissipation; green curves, no damping but with artificial dissipation; blue curves, damping without dissipation; magenta, both damping and dissipation. The cyan lines depict ideal, 2nd-order convergence.

Given a high enough resolution, we see that our simulations converge with 2nd-order accuracy. Damping and 4th-derivative artificial dissipation both eliminate trapped waves (albeit not perfectly) with artificial dissipation damping the physical, outgoing waves a bit more than the sponge layer terms. Moreover, damping and dissipation decrease the solution error, but also decrease the convergence rate at lower resolutions.

5.5.4 Two-level, 2nd-order simulations: $\boldsymbol{\nabla} \cdot \mathbf{E}$

The convergence properties for $\boldsymbol{\nabla} \cdot \mathbf{E}$ errors resulting from 2nd-order, two-level simulations are summarized in the tables below.

Table 5.41: Lower-resolution convergence properties of $\boldsymbol{\nabla} \cdot \mathbf{E}$ after two periods of $\mathbf{J}$ in two-level simulations without damping and without artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in V m^{-2} = kg s^{-2} C^{-1}.

Norm	1/4	Rate	1/8	Rate	1/16	Rate
L_1	6.4867×10^0	0.80342	3.7168×10^0	1.7383	1.1140×10^0	2.7322
L_2	1.7109×10^0	0.95314	8.8369×10^{-1}	2.0958	2.0673×10^{-1}	2.4940
L_∞	1.8998×10^0	0.20216	1.6514×10^0	1.8952	4.4395×10^{-1}	2.2126

Table 5.42: Higher-resolution convergence properties of $\boldsymbol{\nabla} \cdot \mathbf{E}$ after two periods of $\mathbf{J}$ in two-level simulations without damping and without artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in V m^{-2} = kg s^{-2} C^{-1}.

Norm	1/32	Rate	1/64	Rate	1/128
L_1	1.6765×10^{-1}	2.0843	3.9535×10^{-2}	2.0063	9.8410×10^{-3}
L_2	3.6698×10^{-2}	1.9475	9.5144×10^{-3}	1.7213	2.8854×10^{-3}
L_∞	9.5780×10^{-2}	-0.057116	9.9648×10^{-2}	-0.029123	1.0168×10^{-1}

Table 5.43: Lower-resolution convergence properties of $\boldsymbol{\nabla} \cdot \mathbf{E}$ after two periods of $\mathbf{J}$ in two-level simulations without damping and with artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in V m^{-2} = kg s^{-2} C^{-1}.

Norm	1/4	Rate	1/8	Rate	1/16	Rate
L_1	3.8994×10^0	1.8373	1.0912×10^0	1.6751	3.4171×10^{-1}	1.6049
L_2	2.7020×10^0	2.4409	4.9761×10^{-1}	2.7668	7.3113×10^{-2}	1.4814
L_∞	6.2609×10^0	2.2251	1.3391×10^0	3.8936	9.0100×10^{-2}	2.1535

Table 5.44: Higher-resolution convergence properties of $\boldsymbol{\nabla} \cdot \mathbf{E}$ after two periods of $\mathbf{J}$ in two-level simulations without damping and with artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in V m^{-2} = kg s^{-2} C^{-1}.

Norm	1/32	Rate	1/64	Rate	1/128
L_1	1.1234×10^{-1}	1.9129	2.9832×10^{-2}	1.9895	7.5123×10^{-3}
L_2	2.6185×10^{-2}	1.8127	7.4536×10^{-3}	1.9457	1.9349×10^{-3}
L_∞	2.0252×10^{-2}	1.1921	8.8634×10^{-3}	1.0838	4.1817×10^{-3}

Table 5.45: Lower-resolution convergence properties of $\boldsymbol{\nabla} \cdot \mathbf{E}$ after two periods of $\mathbf{J}$ in two-level simulations with damping and without artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in V m^{-2} = kg s^{-2} C^{-1}.

Norm	1/4	Rate	1/8	Rate	1/16	Rate
L_1	4.1158×10^{0}	1.0240	2.0240×10^{0}	1.4828	7.2419×10^{-1}	2.4313
L_2	1.4607×10^{0}	1.1824	6.4361×10^{-1}	2.1575	1.4426×10^{-1}	2.0702
L_∞	4.2073×10^{0}	0.27364	3.4804×10^{0}	3.7836	2.5273×10^{-1}	0.61644

Table 5.46: Higher-resolution convergence properties of $\boldsymbol{\nabla} \cdot \mathbf{E}$ after two periods of $\mathbf{J}$ in two-level simulations with damping and without artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in V m^{-2} = kg s^{-2} C^{-1}.

Norm	1/32	Rate	1/64	Rate	1/128
L_1	1.3426×10^{-1}	2.0552	3.2305×10^{-2}	2.0063	8.0412×10^{-3}
L_2	3.4352×10^{-2}	1.7929	9.9140×10^{-3}	1.5901	3.2930×10^{-3}
L_∞	1.6485×10^{-1}	0.15878	1.4767×10^{-1}	0.13306	1.3466×10^{-1}

Table 5.47: Lower-resolution convergence properties of $\boldsymbol{\nabla} \cdot \mathbf{E}$ after two periods of $\mathbf{J}$ in two-level simulations with damping and with artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in V m^{-2} = kg s^{-2} C^{-1}.

Norm	1/4	Rate	1/8	Rate	1/16	Rate
L_1	3.8845×10^{0}	1.8004	1.1152×10^{0}	1.6779	3.4853×10^{-1}	1.6141
L_2	2.6982×10^{0}	2.4429	4.9622×10^{-1}	2.7630	7.3102×10^{-2}	1.4790
L_∞	6.2609×10^{0}	2.2251	1.3391×10^{0}	3.8936	9.0099×10^{-2}	2.4836

Table 5.48: Higher-resolution convergence properties of $\boldsymbol{\nabla} \cdot \mathbf{E}$ after two periods of $\mathbf{J}$ in two-level simulations with damping and with artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in V m^{-2} = kg s^{-2} C^{-1}.

Norm	1/32	Rate	1/64	Rate	1/128
L_1	1.1385×10^{-1}	1.9184	3.0119×10^{-2}	1.9948	7.5572×10^{-3}
L_2	2.6225×10^{-2}	1.8158	7.4493×10^{-3}	1.9528	1.9243×10^{-3}
L_∞	1.6110×10^{-2}	1.2521	6.7636×10^{-3}	0.98557	3.4158×10^{-3}

The convergence properties of $\boldsymbol{\nabla} \cdot \mathbf{E}$ are further illustrated in Fig. 5.5 below. The cyan and pink lines represent 2nd- and 1st-order convergence, respectively. Otherwise, the curves have the same meanings as in Fig. 5.3.

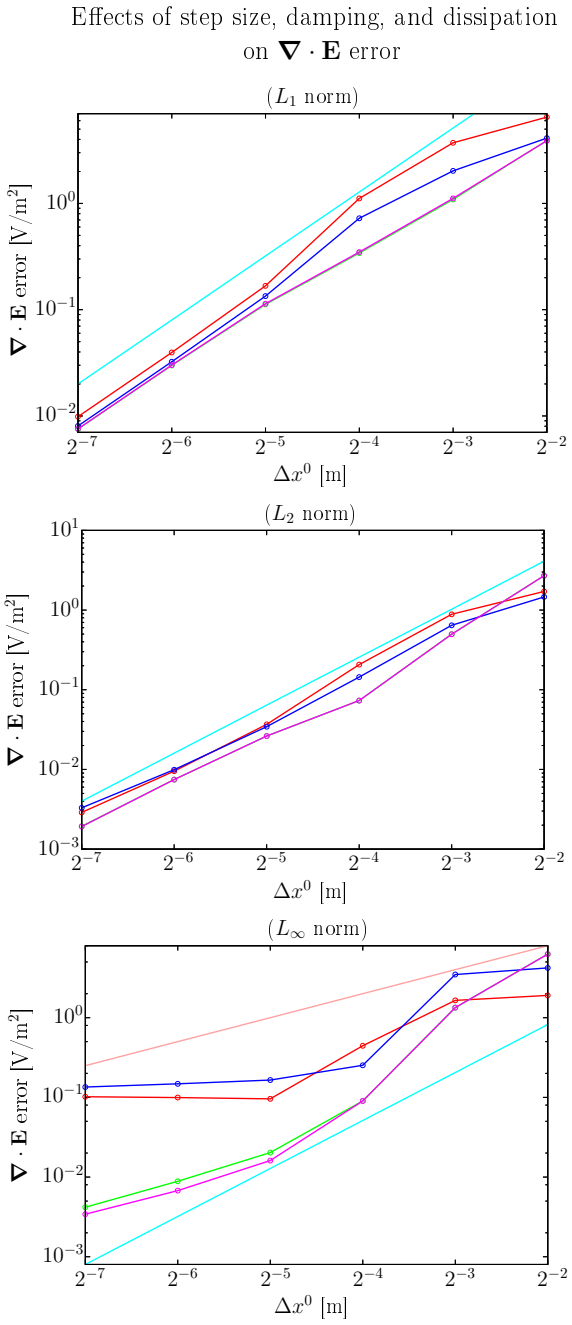

Figure 5.5: Illustration of convergence properties of $\boldsymbol{\nabla} \cdot \mathbf{E}$ errors for a 2nd-order simulation of the Faraday-Ampère-Maxwell system, with $\mathbf{J}$ given by Eq. (5.55) and parameters listed in Eq. (5.56). Red curves represent data from simulations with neither damping nor (6th-derivative) artificial dissipation; green curves, no damping but with artificial dissipation; blue curves, damping without dissipation; magenta, both damping and dissipation. The cyan lines depict ideal, 2nd-order convergence. The pink line in the bottom graph depicts 1st-order convergence.

For L_1 and L_2 norms, we see that the errors in $\boldsymbol{\nabla}\cdot\mathbf{E}$ converge to 0 with 2nd-order accuracy, and that damping and dissipation make little difference. However, for the L_∞ norm, we do not observe 2nd-order convergence, though simulations with dissipation converge better than those without it.

5.6 High-frequency simulation results in 2D

5.6.1 Variable frequency

For this battery of simulations, we considered $\nu_1 = \nu_{\max}/2$, and $J_1 = 1/10$ V/m, with otherwise the same parameters and conditions as in Sec. 5.5 and Eq. (5.56).

$$\begin{aligned}
J_1 &= \frac{1}{10}\ \mathrm{Cm^{-2}s^{-1}}\ (\mathrm{V/m}) \\
a &= \frac{1}{2}\ \mathrm{m} \\
x_1 &= 0\ \mathrm{m} \\
y_1 &= 0\ \mathrm{m} \\
\nu_1 &= \frac{\nu_{\max}}{2} = \frac{299792458[\mathrm{ms^{-1}}]}{2(\mathrm{CFL})(h^0[\mathrm{m}])}\ \mathrm{s^{-1}}\ (\mathrm{Hz}) \\
h^0 &= \frac{1}{4},\ \frac{1}{8},\ \frac{1}{16},\ \frac{1}{32}\ \mathrm{m} \\
h^1 &= \frac{1}{16},\ \frac{1}{32},\ \frac{1}{64},\ \frac{1}{128}\ \mathrm{m} \\
\mathrm{CFL} &= \frac{1}{2} \\
\Omega^0 &= [-10, 10]\ \mathrm{m} \\
\Omega^1 &= [-2, 2]\ \mathrm{m} \\
\text{nesting depth} &= 0,\ 2 \text{ coarse cells} \\
\text{ramping depth} &= 0,\ 2 \text{ coarse cells}
\end{aligned} \tag{5.58}$$

As such, we expect the oscillations in $\mathbf{J}$ to be marginally resolved at best.

The figures below show some expected and some unexpected properties. Outside of the refined patch Ω^1, the dominant oscillations in B_z have a period of $6h^0$ for each h^0. Moreover, as we increase the resolution (i.e. decrease h), and with it ν_1, the trapped waves in Ω^1 are eliminated, despite the fact that ν_1 is specifically chosen not to be properly resolved on Ω^0. As in Sec. 5.5, artificial dissipation appears to eliminate trapped waves inside Ω^1 better than damping. However, it also eliminates many of the $6h^0$ waves as they propagate out of Ω^1 to Ω^0. Then again, those waves are not physical, which suggests that artificial dissipation is a superior mechanism to damping in maintaining stability, particularly in lower-resolution simulations.

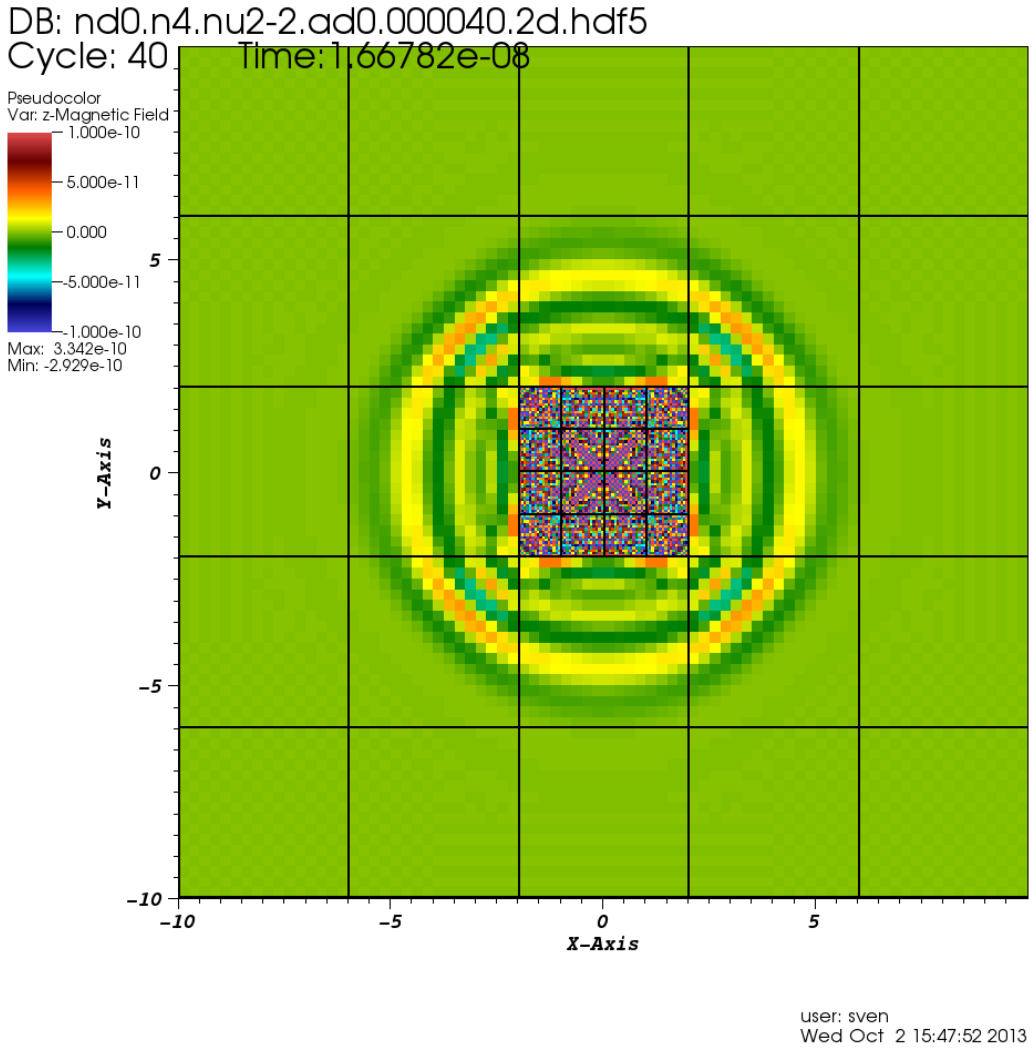

Figure 5.6: Computed B_z with $\Delta x^0 = 1/4$ m at $t = 40\Delta t^0 = 1.66782 \times 10^{-8}$ s for the high-frequency $\mathbf{J}$. This simulation does not employ damping or artificial dissipation. Note abundant trapped waves inside the refined patch and leading oscillations with wavelength ≈ 1.5 m $= 6\Delta x^0$ outside the refined region.

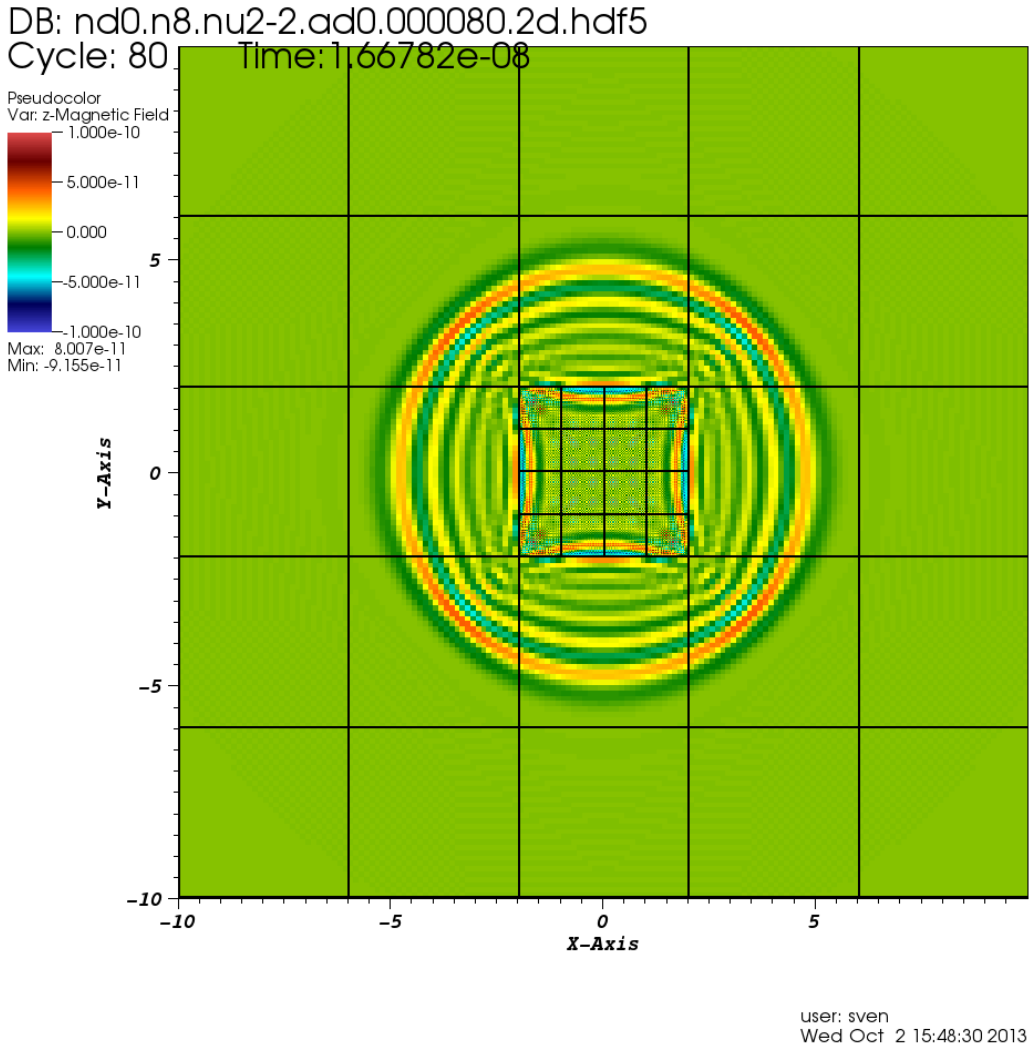

Figure 5.7: Computed B_z with $\Delta x^0 = 1/8$ m at $t = 80\Delta t^0 = 1.66782 \times 10^{-8}$ s for the high-frequency $\mathbf{J}$. This simulation does not employ damping or artificial dissipation. Note trapped waves (albeit not as abundant as in Fig. 5.6) inside the refined region and leading oscillations with wavelength ≈ 0.75 m $= 6\Delta x^0$ outside the refined region.

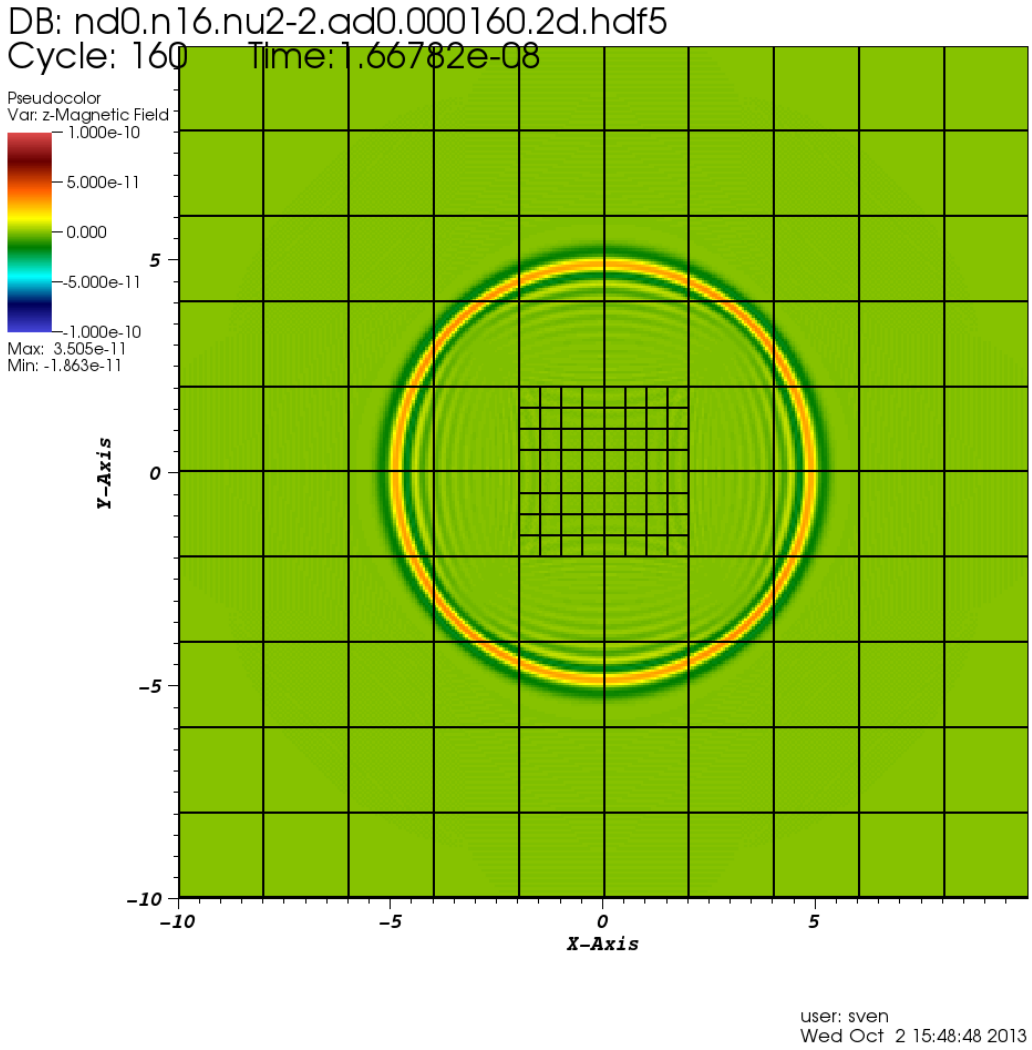

Figure 5.8: Computed B_z with $\Delta x^0 = 1/16$ m at $t = 160\Delta t^0 = 1.66782 \times 10^{-8}$ s for the high-frequency $\mathbf{J}$. This simulation does not employ damping or artificial dissipation. Note the leading oscillations with wavelength ≈ 0.375 m $= 6\Delta x^0$ outside the refined region and the relative lack of trapped waves inside the refined region.

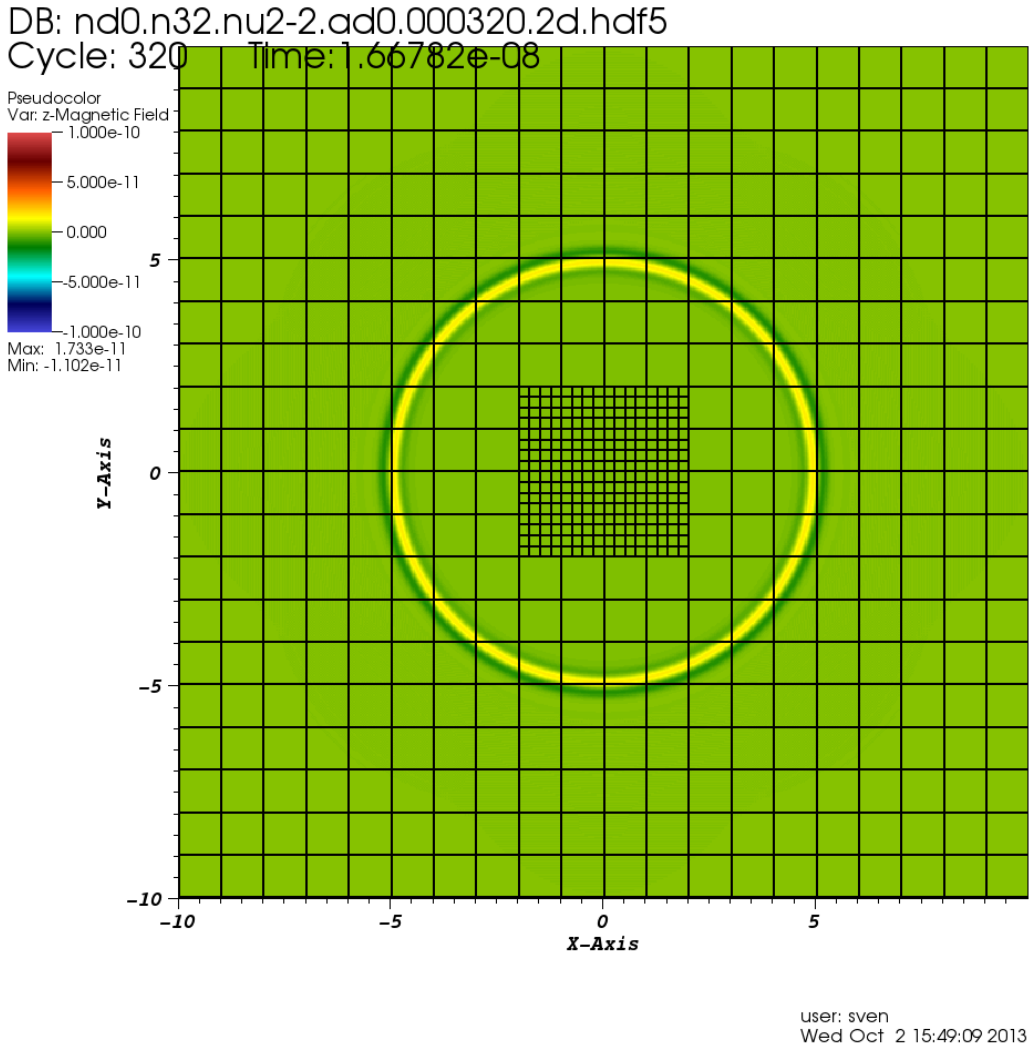

Figure 5.9: Computed B_z with $\Delta x^0 = 1/32$ m at $t = 320\Delta t^0 = 1.66782 \times 10^{-8}$ s for the high-frequency $\mathbf{J}$. This simulation does not employ damping or artificial dissipation. Note the leading oscillations with wavelength ≈ 0.1875 m $= 6\Delta x^0$ outside the refined region. There are virtually no trapped waves inside the refined region.

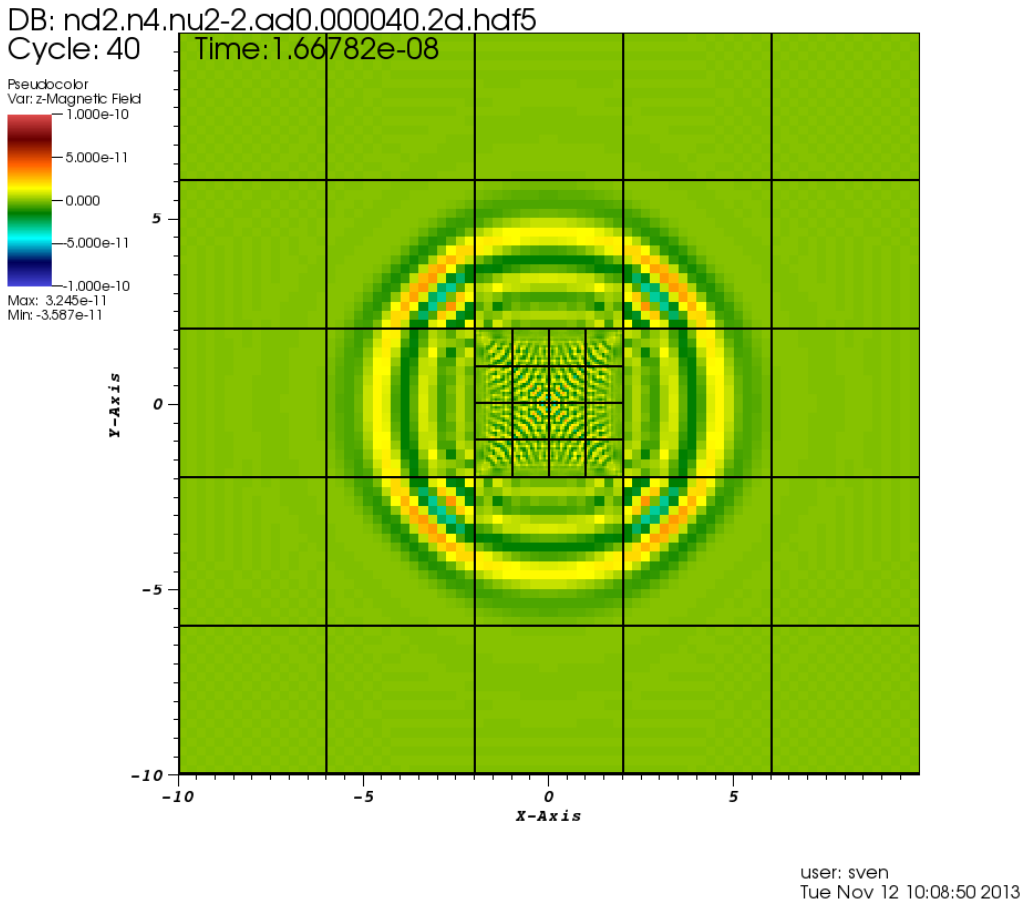

Figure 5.10: Computed B_z with $\Delta x^0 = 1/4$ m at $t = 40\Delta t^0 = 1.66782 \times 10^{-8}$ s for the high-frequency $\mathbf{J}$. This simulation employs damping but not artificial dissipation. Note the leading oscillations with wavelength ≈ 1.5 m $= 6\Delta x^0$ outside the refined region. The trapped waves inside the refined region are much smaller than those in Fig. 5.6. In fact, they are comparable in amplitude to the trapped waves in Fig. 5.7

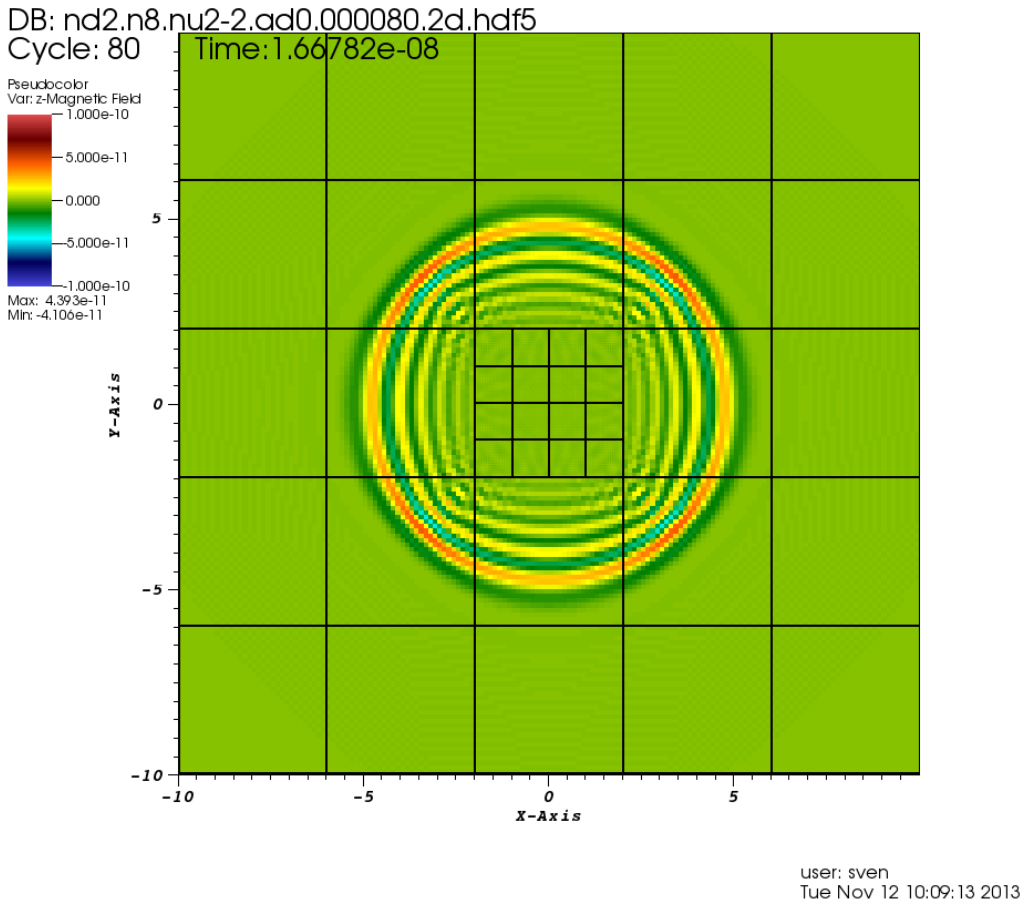

Figure 5.11: Computed B_z with $\Delta x^0 = 1/8$ m at $t = 80\Delta t^0 = 1.66782 \times 10^{-8}$ s for the high-frequency $\mathbf{J}$. This simulation employs damping but not artificial dissipation. Note the leading oscillations with wavelength ≈ 0.75 m $= 6\Delta x^0$ outside the refined region. The trapped waves inside the refined region are barely visible at this scale.

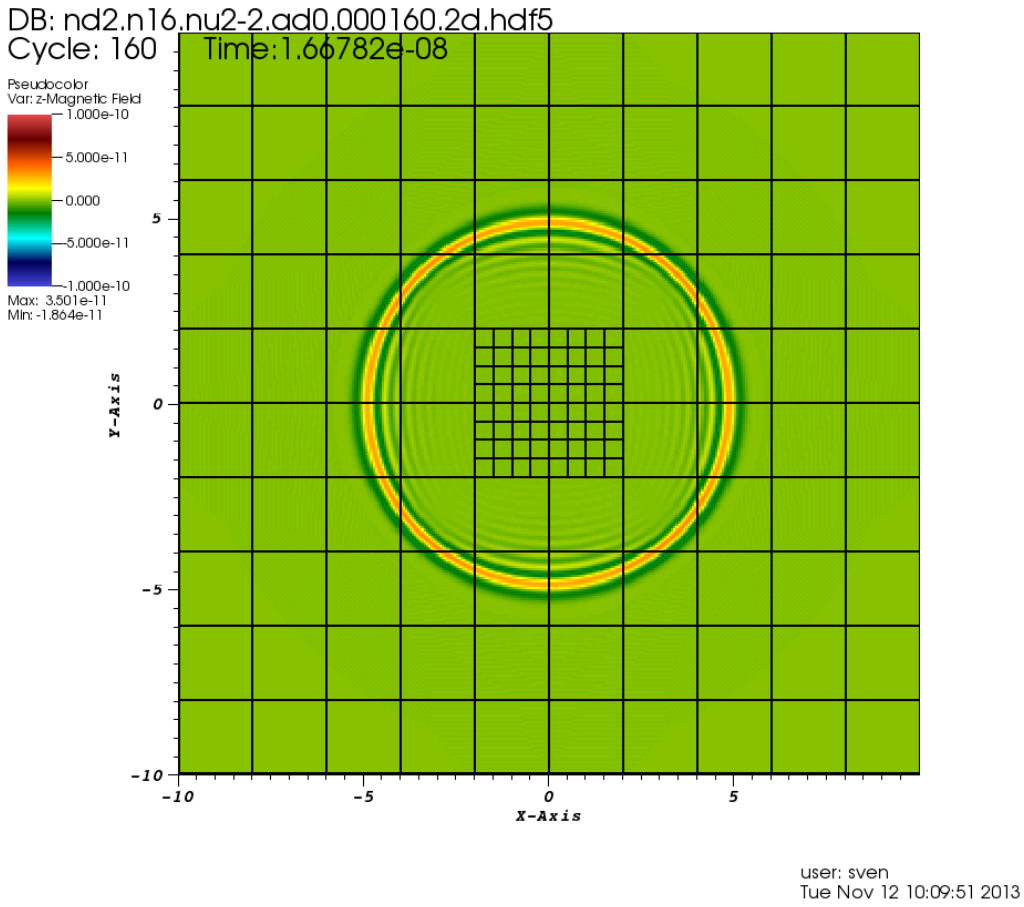

Figure 5.12: Computed B_z with $\Delta x^0 = 1/16$ m at $t = 160\Delta t^0 = 1.66782 \times 10^{-8}$ s for the high-frequency $\mathbf{J}$. This simulation employs damping but not artificial dissipation. Note the leading oscillations with wavelength ≈ 0.375 m $= 6\Delta x^0$ outside the refined region. There are virtually no trapped waves inside the refined region.

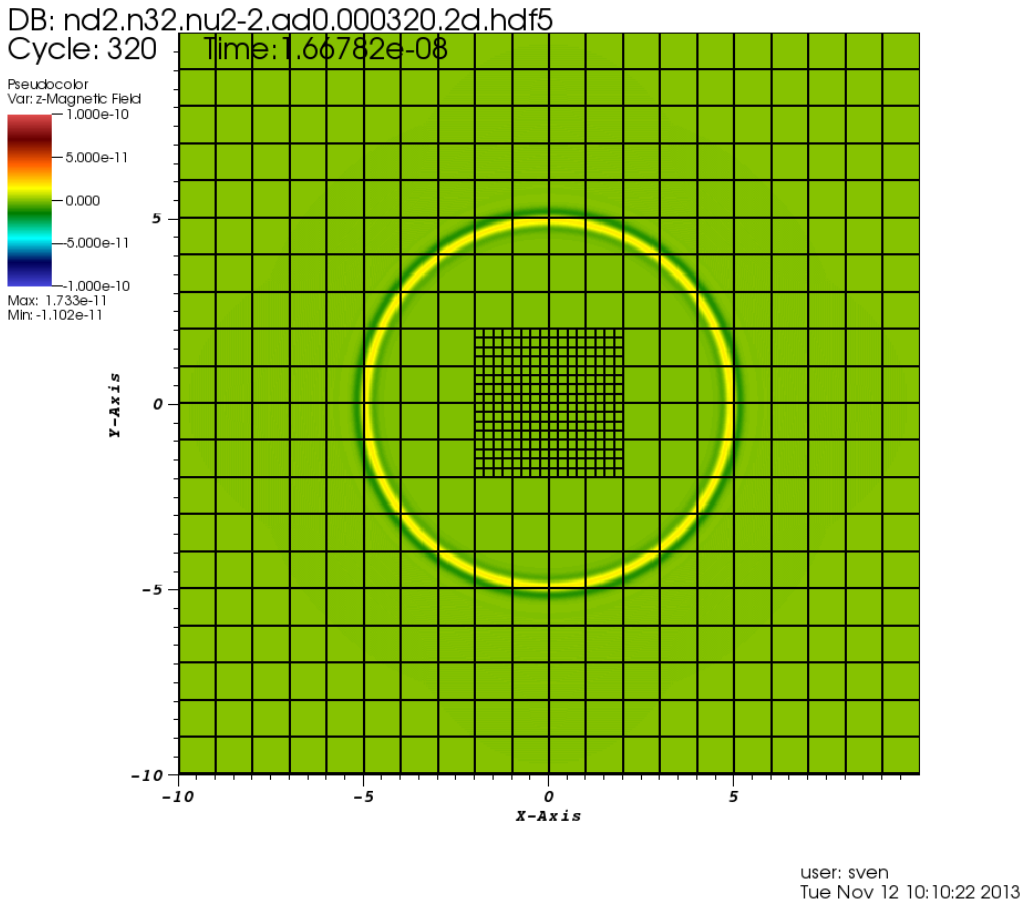

Figure 5.13: Computed B_z with $\Delta x^0 = 1/32$ m at $t = 320\Delta t^0 = 1.66782 \times 10^{-8}$ s for the high-frequency $\mathbf{J}$. This simulation employs damping but not artificial dissipation. Note the leading oscillations with wavelength ≈ 0.1875 m $= 6\Delta x^0$ outside the refined region. There are virtually no trapped waves inside the refined region.

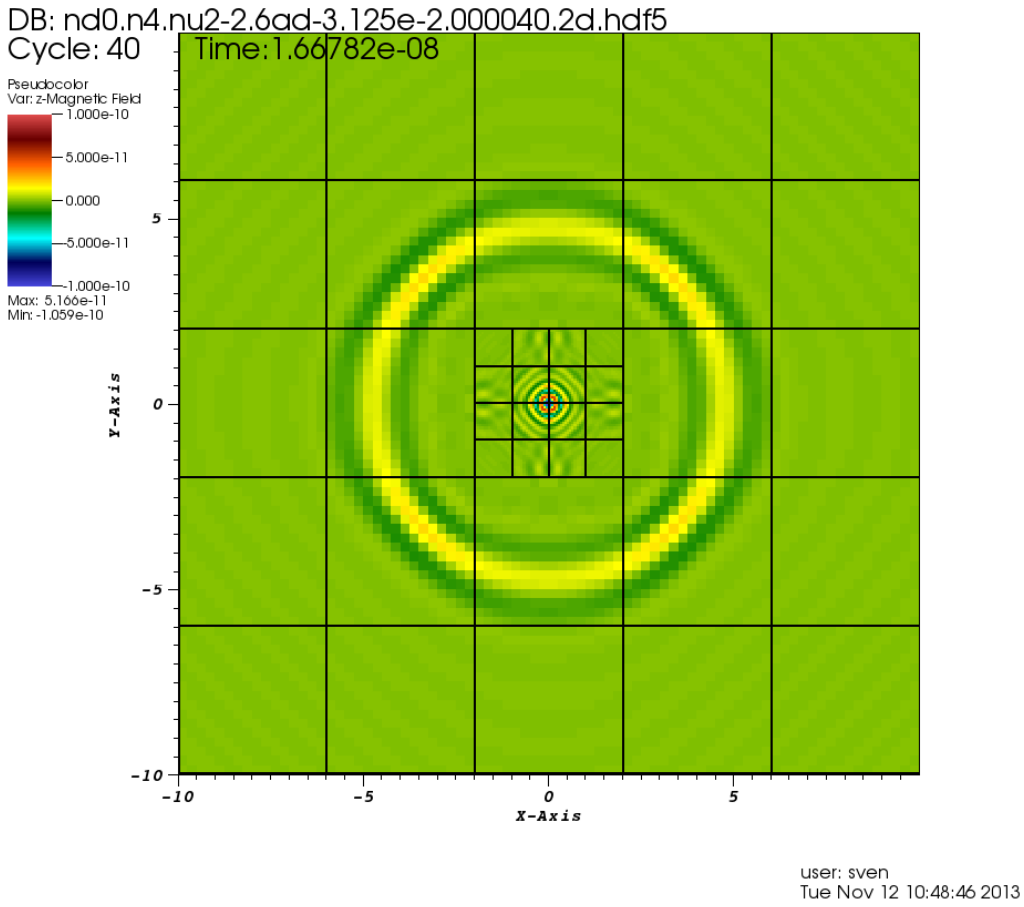

Figure 5.14: Computed B_z with $\Delta x^0 = 1/4$ m at $t = 40\Delta t^0 = 1.66782 \times 10^{-8}$ s for the high-frequency $\mathbf{J}$. This simulation does not employ damping but does employ artificial dissipation. Note the leading oscillations with wavelength ≈ 1.5 m $= 6\Delta x^0$ outside the refined region. The trapped waves inside the refined region less numerous and smaller in amplitude than those in Fig. 5.10, expect very close to the center.

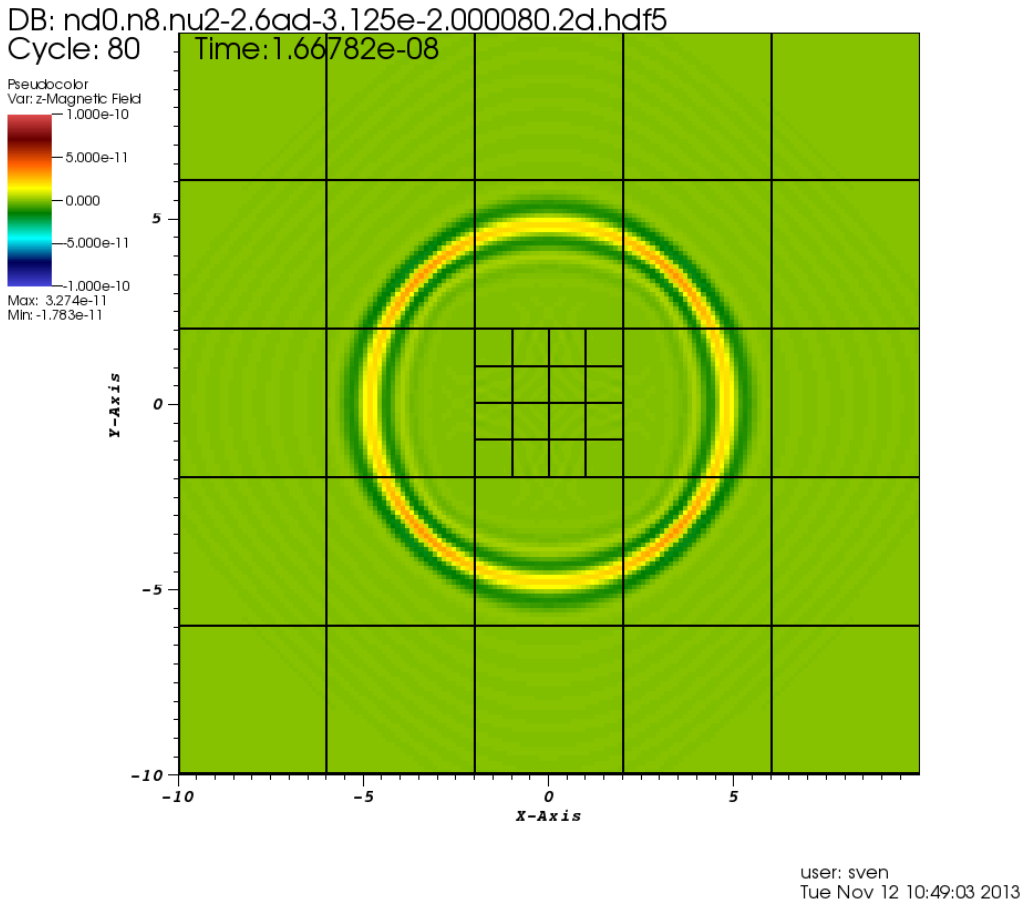

Figure 5.15: Computed B_z with $\Delta x^0 = 1/8$ m at $t = 80\Delta t^0 = 1.66782 \times 10^{-8}$ s for the high-frequency $\mathbf{J}$. This simulation does not employ damping but does employ artificial dissipation. Note the leading oscillations with wavelength ≈ 0.75 m $= 6\Delta x^0$ outside the refined region. The trapped waves inside the refined region are even less visible than in Fig. 5.11

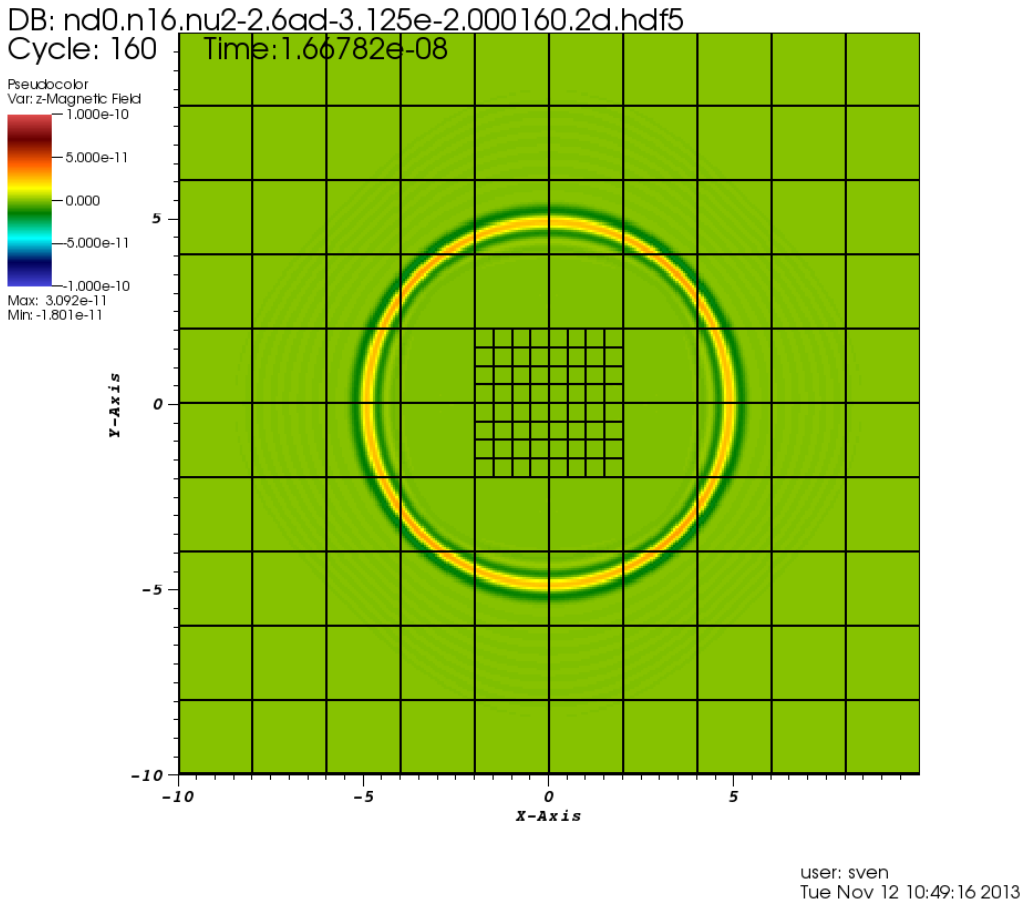

Figure 5.16: Computed B_z with $\Delta x^0 = 1/16$ m at $t = 160\Delta t^0 = 1.66782 \times 10^{-8}$ s for the high-frequency $\mathbf{J}$. This simulation does not employ damping but does employ artificial dissipation. Note the leading oscillations with wavelength ≈ 0.375 m $= 6\Delta x^0$ outside the refined region. There are virtually no trapped waves inside the refined region.

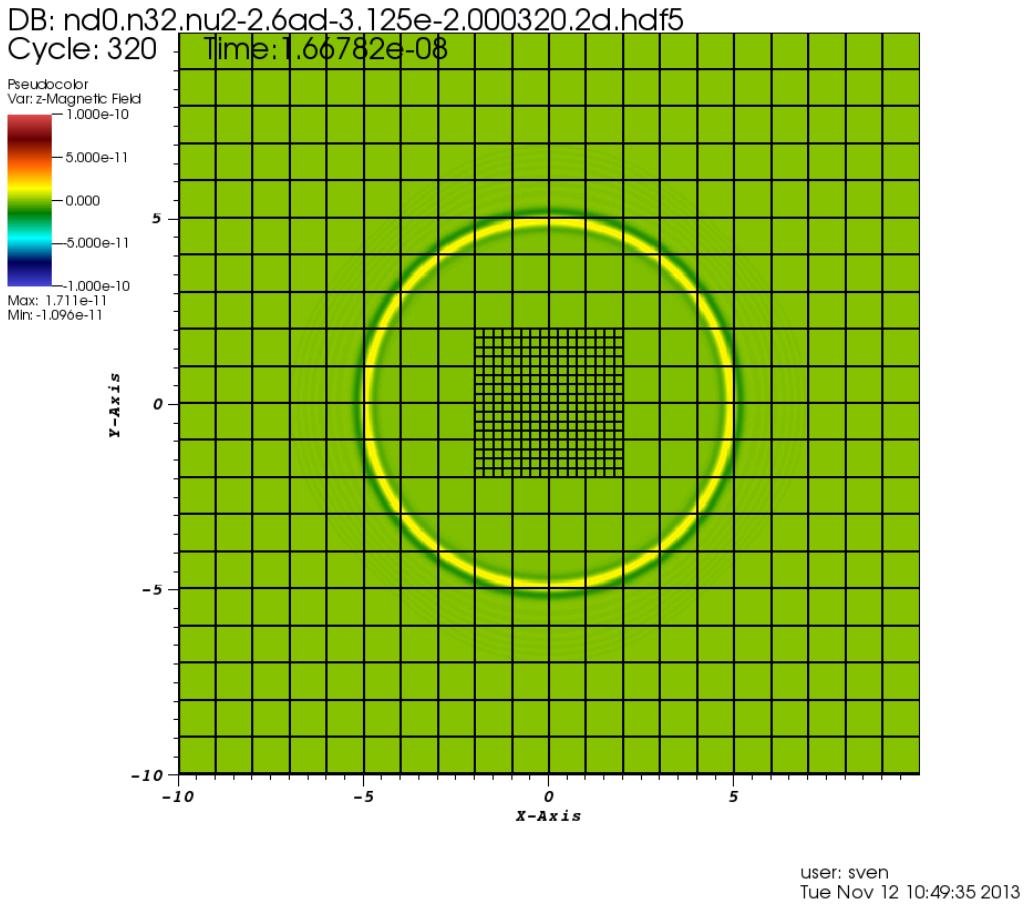

Figure 5.17: Computed B_z with $\Delta x^0 = 1/32$ m at $t = 320\Delta t^0 = 1.66782 \times 10^{-8}$ s for the high-frequency $\mathbf{J}$. This simulation does not employ damping but does employ artificial dissipation. Note the leading oscillations with wavelength ≈ 0.1875 m $= 6\Delta x^0$ outside the refined region. There are virtually no trapped waves inside the refined region.

5.6.2 Fixed frequency

For this battery of simulations, we considered the following parameters and conditions:

$$\begin{aligned} J_1 &= 1 \text{ Cm}^{-2}\text{s}^{-1} \text{ (V/m)} \\ a &= \frac{1}{2} \text{ m} \\ x_1 &= 0 \text{ m} \\ y_1 &= 0 \text{ m} \\ \nu_1 &= 1199169832 \text{ s}^{-1} \text{ (Hz)} \\ h^0 &= \frac{1}{4}, \frac{1}{8}, \frac{1}{16}, \frac{1}{32} \text{ m} \\ h^1 &= \frac{1}{16}, \frac{1}{32}, \frac{1}{64}, \frac{1}{128} \text{ m} \\ \text{CFL} &= \frac{1}{2} \\ \Omega^0 &= [-10, 10] \text{ m} \\ \Omega^1 &= [-2, 2] \text{ m} \\ \text{nesting depth} &= 0, \text{ 2 coarse cells} \\ \text{ramping depth} &= 0, \text{ 2 coarse cells} \end{aligned} \tag{5.59}$$

ν_1 was specifically chosen as $\nu_{\max}/2$ for $h^0 = 1/4$ m. As such, we do not expect the oscillations in $\mathbf{J}$ to be well resolved until $h^0 = 1/32$ m and lower.

The figures below show some expected and some unexpected properties. Outside of the refined patch Ω^1, the dominant oscillations in B_z have a period of $6h^0$ for each h^0 ¿ 1/32 m. Once we reach that level of resolution, however, the oscillations in B_z have a spatial period of 1/4 m, as they should. As in Sec. 5.5, artificial dissipation appears to eliminate trapped waves inside Ω^1 better than damping. However, it also eliminates many of the $6h^0$ waves as they propagate out of Ω^1 to Ω^0. Then again, those waves are not physical, which suggests that artificial dissipation is a superior mechanism to damping in maintaining stability, particularly in lower-resolution simulations.

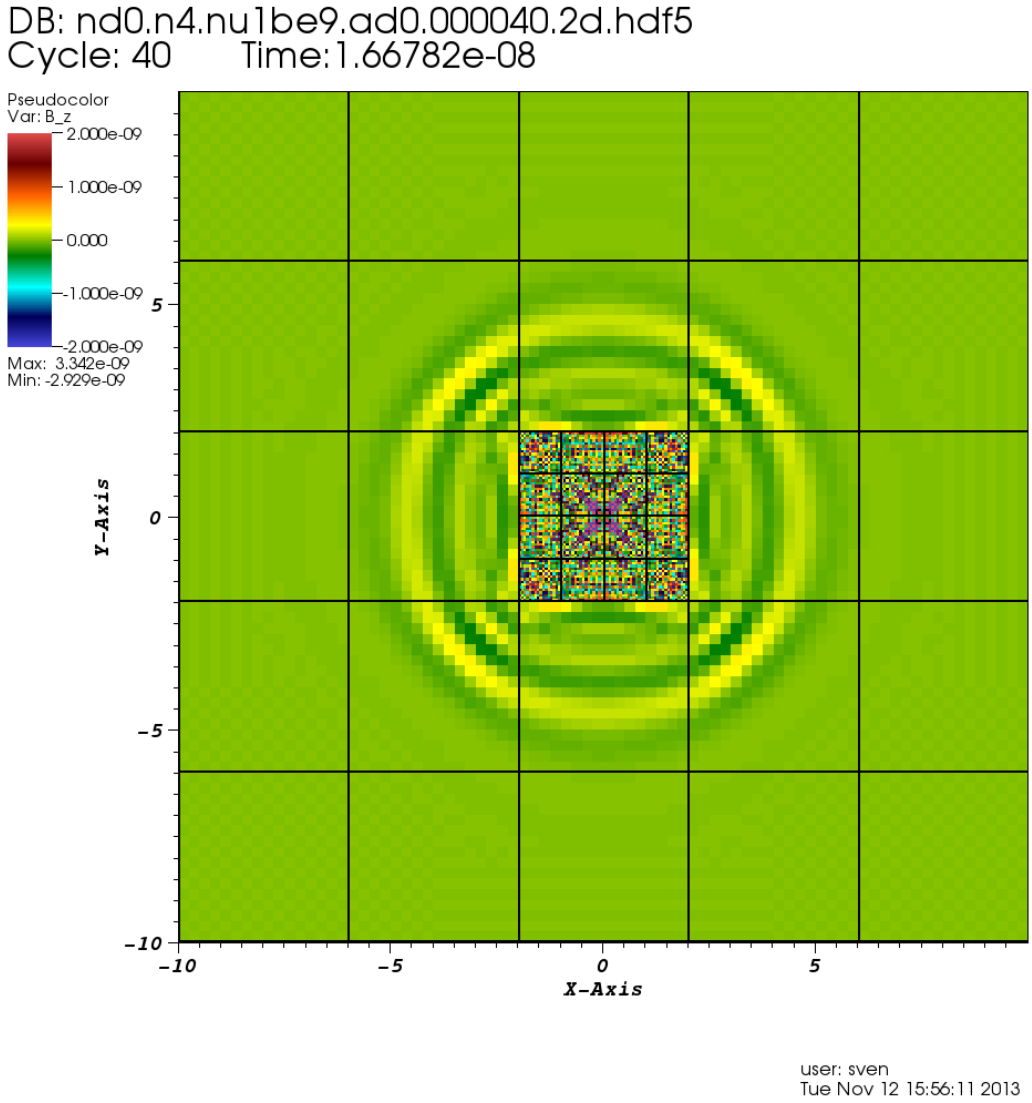

Figure 5.18: Computed B_z with $\Delta x^0 = 1/4$ m at $t = 40\Delta t^0 = 1.66782 \times 10^{-8}$ s for the (fixed) high-frequency $\mathbf{J}$. This simulation does not employ damping or artificial dissipation. Note abundant trapped waves inside the refined patch and leading oscillations with wavelength $\approx$ 1.5 m $= 6\Delta x^0$ outside the refined region.

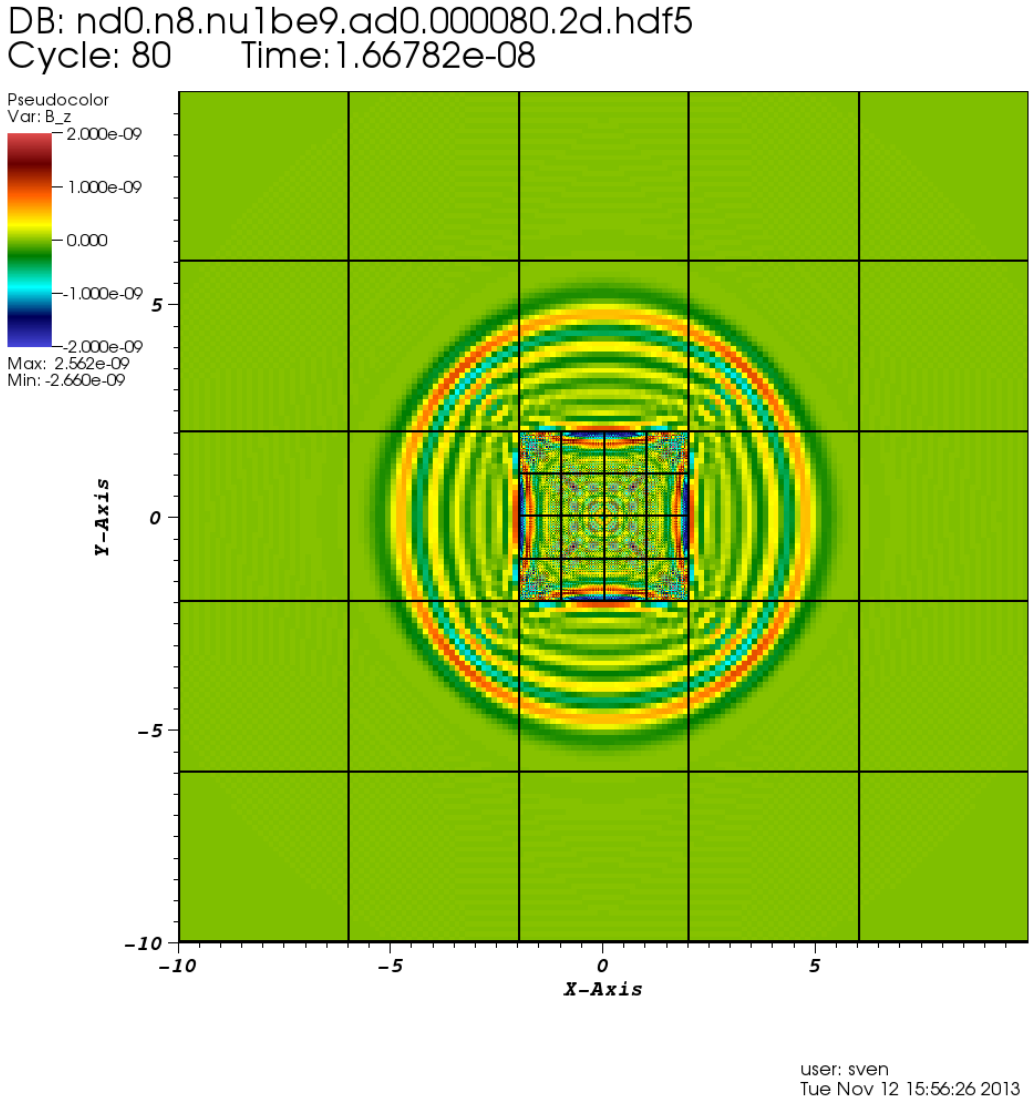

Figure 5.19: Computed B_z with $\Delta x^0 = 1/8$ m at $t = 80\Delta t^0 = 1.66782 \times 10^{-8}$ s for the (fixed) high-frequency $\mathbf{J}$. This simulation does not employ damping or artificial dissipation. Note trapped waves (albeit not as abundant as in Fig. 5.18) inside the refined region and leading oscillations with wavelength ≈ 0.75 m $= 6\Delta x^0$ outside the refined region.

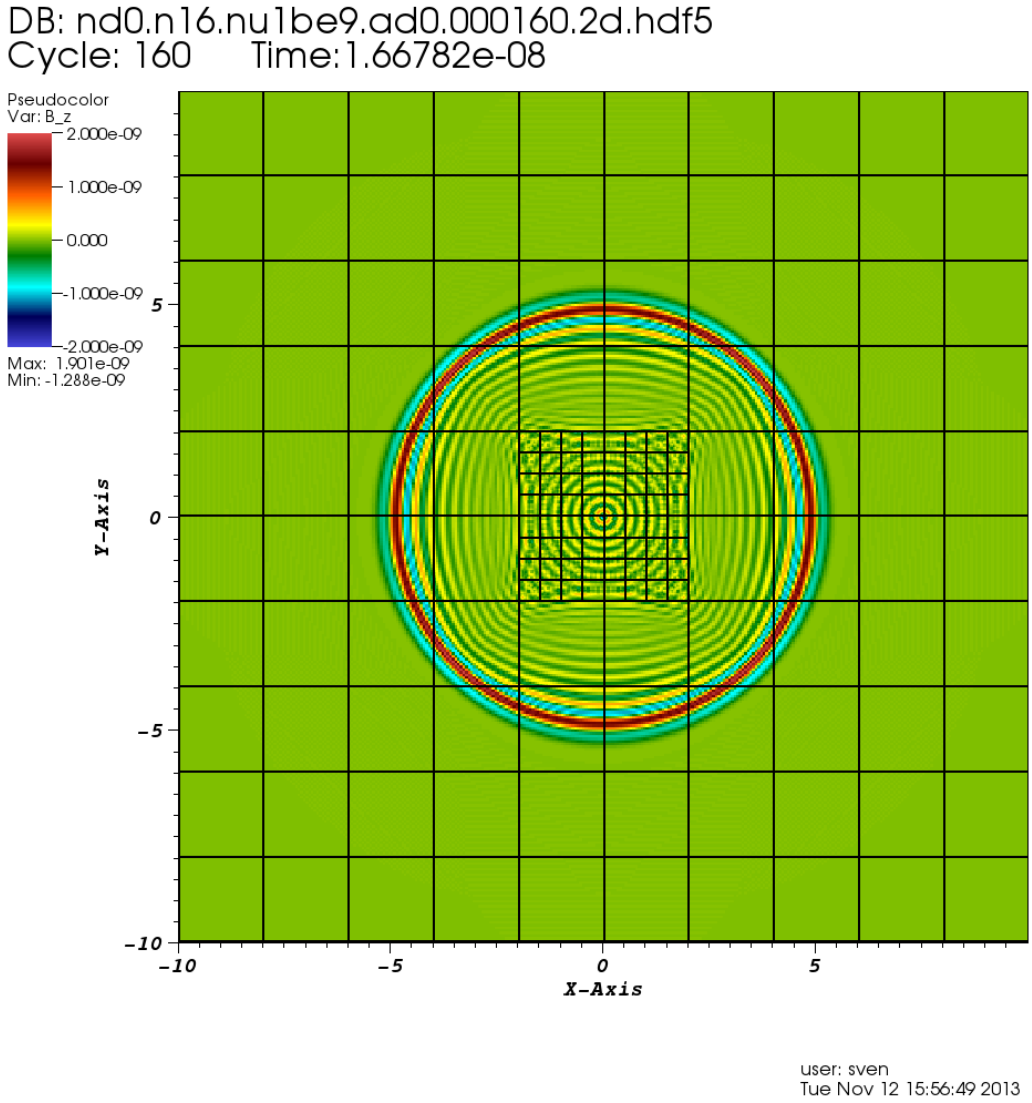

Figure 5.20: Computed B_z with $\Delta x^0 = 1/16$ m at $t = 160\Delta t^0 = 1.66782 \times 10^{-8}$ s for the (fixed) high-frequency $\mathbf{J}$. This simulation does not employ damping or artificial dissipation. Note the leading oscillations with wavelength ≈ 0.375 m $= 6\Delta x^0$ outside the refined region and the relative lack of trapped waves inside the refined region.

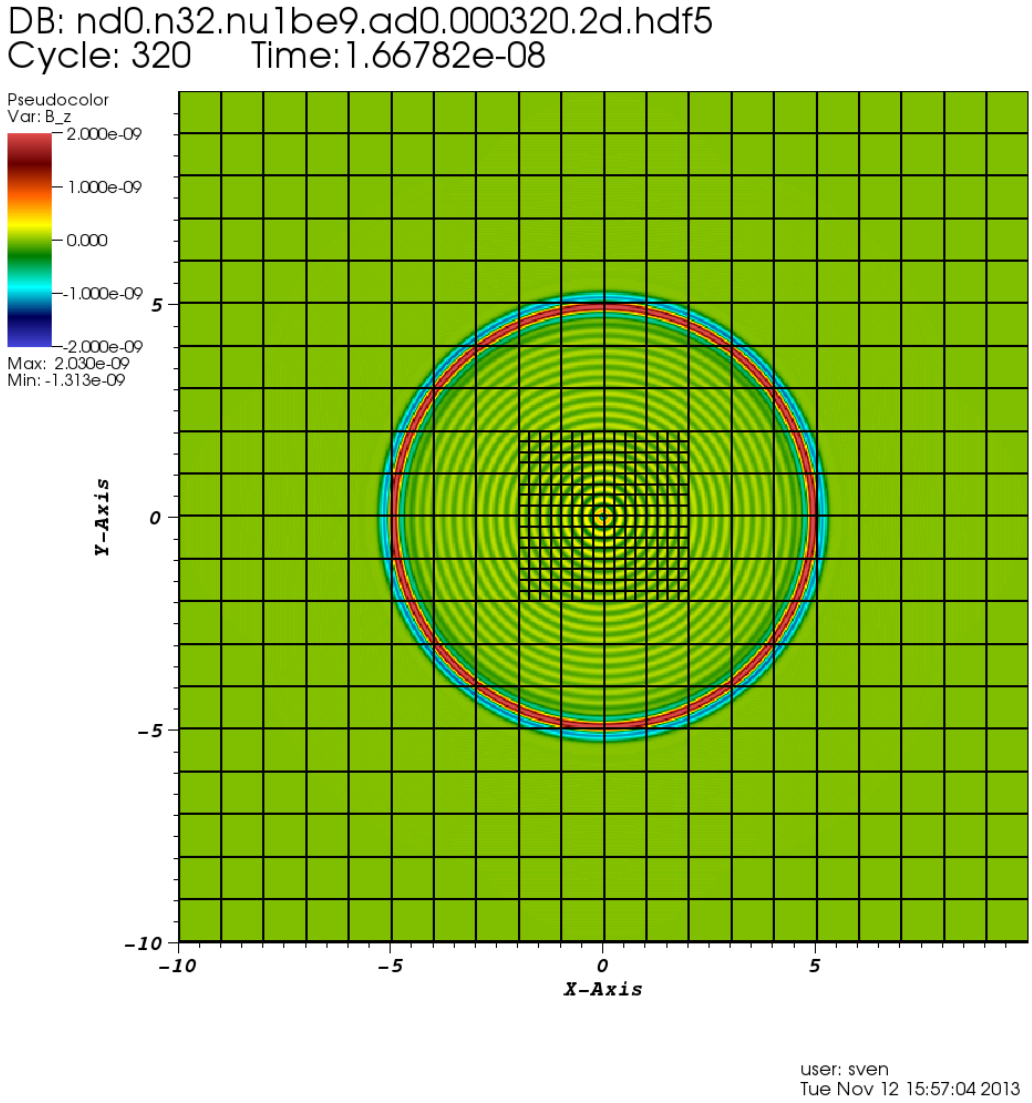

Figure 5.21: Computed B_z with $\Delta x^0 = 1/32$ m at $t = 320\Delta t^0 = 1.66782 \times 10^{-8}$ s for the (fixed) high-frequency $\mathbf{J}$. This simulation does not employ damping or artificial dissipation. We are now properly resolved; the oscillations have a wavelength of 0.25 m $= 8\Delta x^0$, as they should. There are virtually no trapped waves inside the refined region.

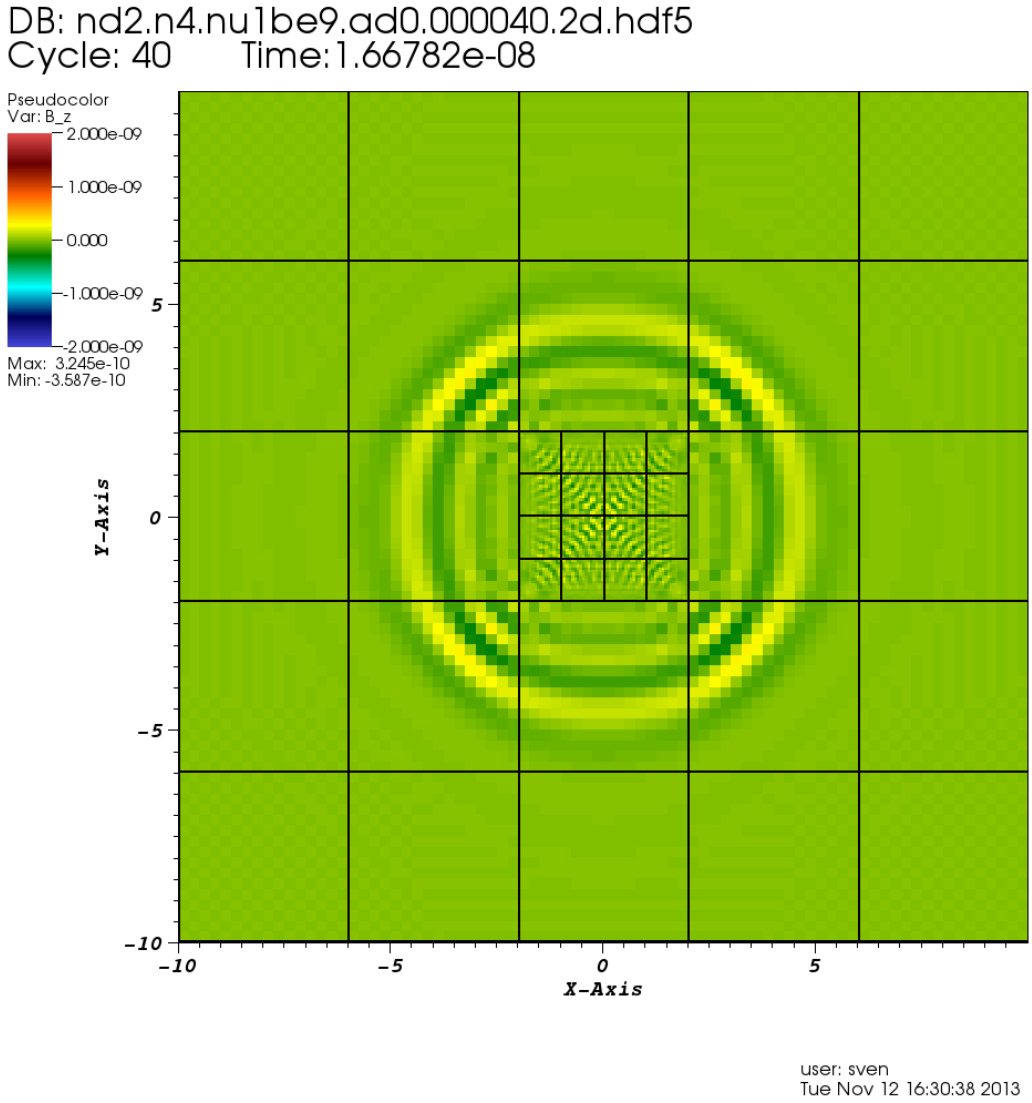

Figure 5.22: Computed B_z with $\Delta x^0 = 1/4$ m at $t = 40\Delta t^0 = 1.66782 \times 10^{-8}$ s for the (fixed) high-frequency $\mathbf{J}$. This simulation employs damping, but not artificial dissipation. Note the leading oscillations with wavelength ≈ 1.5 m $= 6\Delta x^0$ outside the refined region. Trapped waves inside the refined region are damped significantly compared to Fig. 5.18.

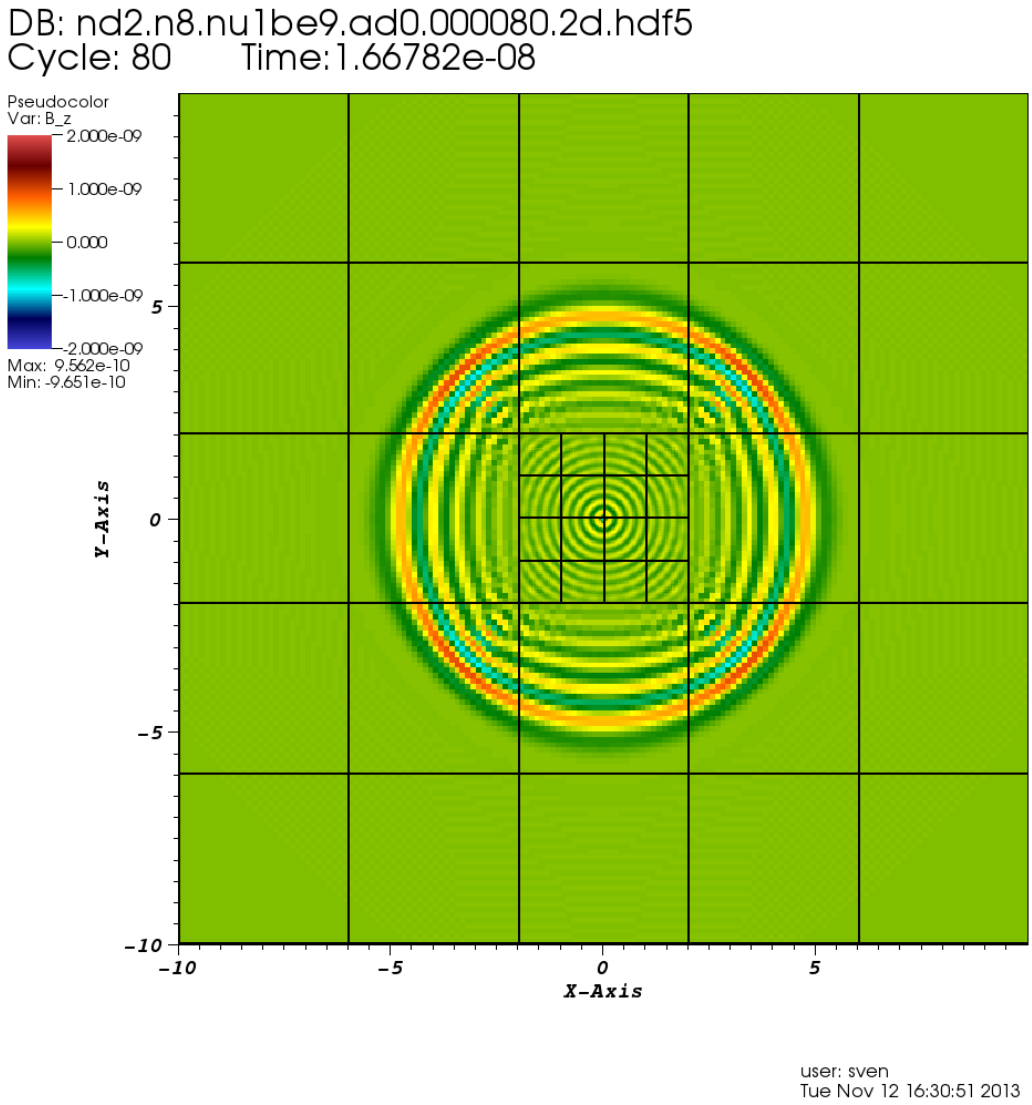

Figure 5.23: Computed B_z with $\Delta x^0 = 1/8$ m at $t = 80\Delta t^0 = 1.66782 \times 10^{-8}$ s for the (fixed) high-frequency $\mathbf{J}$. This simulation employs damping, but not artificial dissipation. Note the leading oscillations with wavelength ≈ 0.75 m $= 6\Delta x^0$ outside the refined region. Trapped waves inside the refined region are barely visible at this scale.

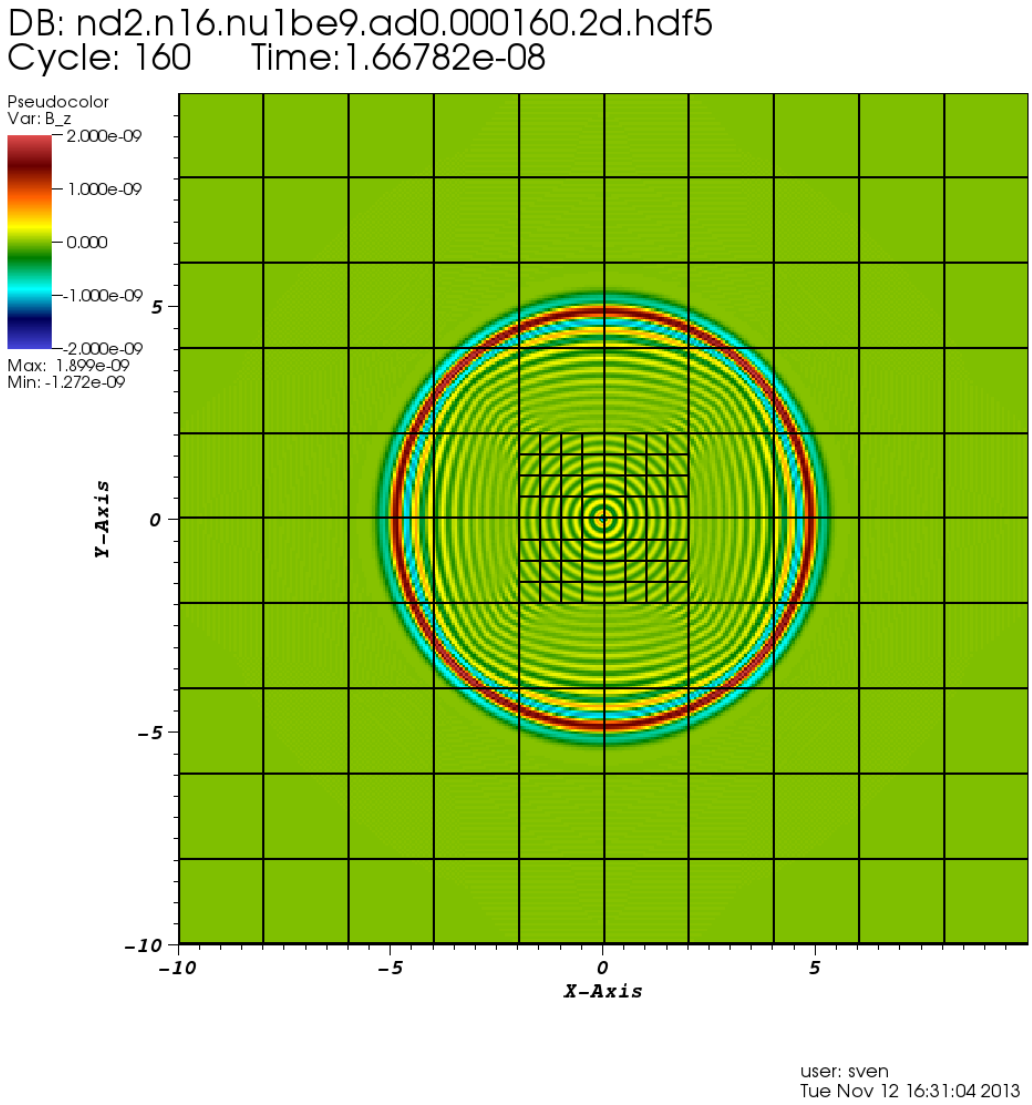

Figure 5.24: Computed B_z with $\Delta x^0 = 1/16$ m at $t = 160\Delta t^0 = 1.66782 \times 10^{-8}$ s for the (fixed) high-frequency $\mathbf{J}$. This simulation employs damping, but not artificial dissipation. Note the leading oscillations with wavelength ≈ 0.375 m $= 6\Delta x^0$ outside the refined region. There are virtually no trapped waves inside the refined region.

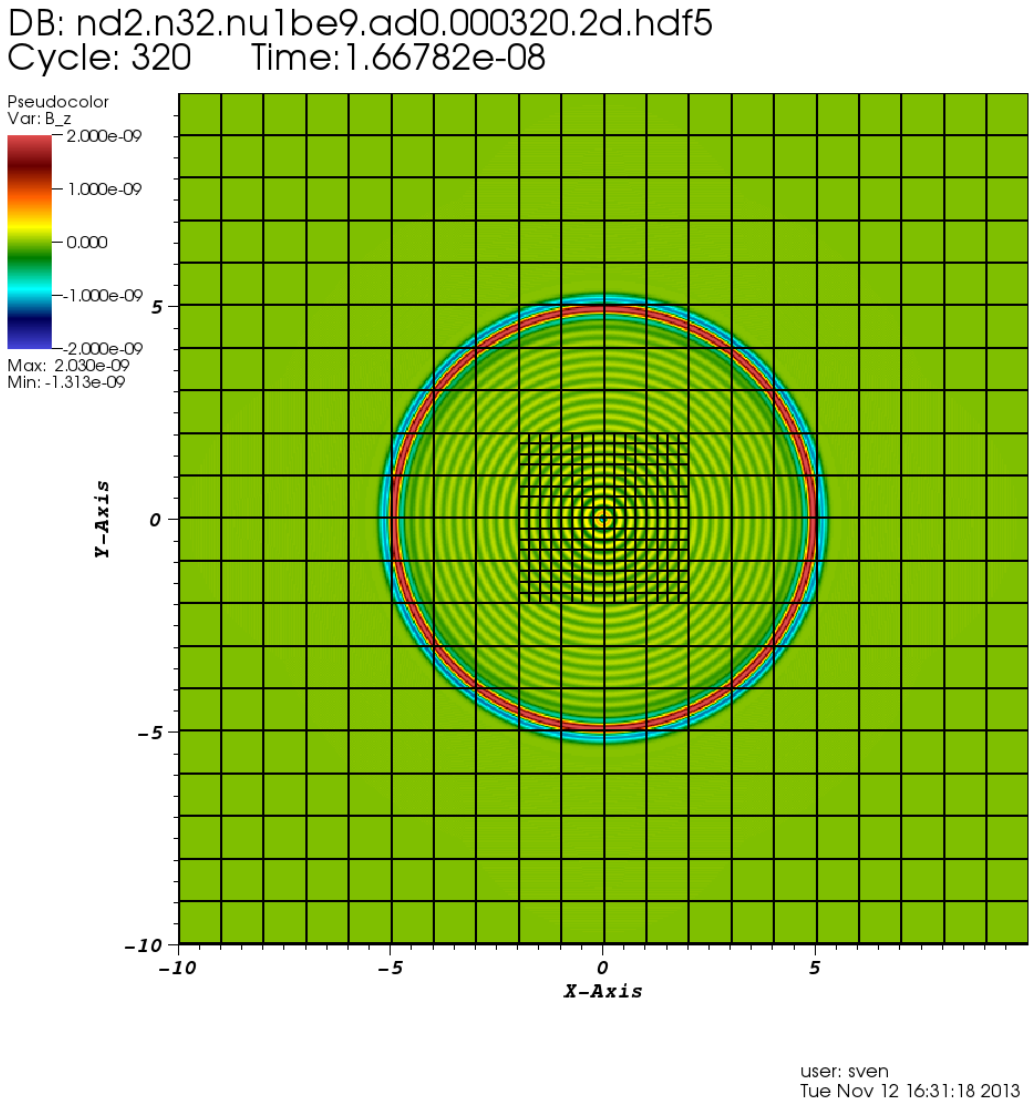

Figure 5.25: Computed B_z with $\Delta x^0 = 1/32$ m at $t = 320\Delta t^0 = 1.66782 \times 10^{-8}$ s for the (fixed) high-frequency $\mathbf{J}$. This simulation employs damping, but not artificial dissipation. We are now properly resolved; the oscillations have a wavelength of 0.25 m $= 8\Delta x^0$, as they should. There are virtually no trapped waves inside the refined region.

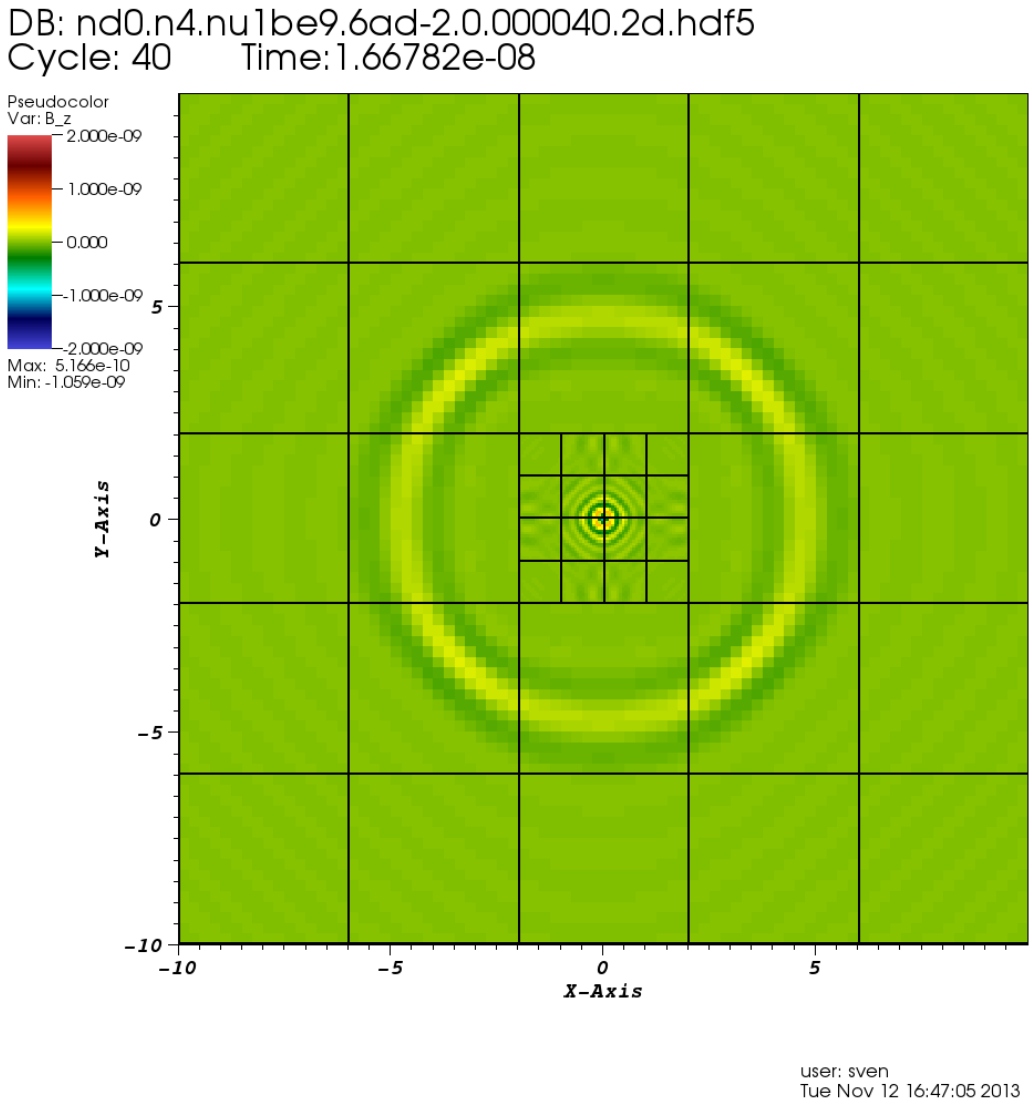

Figure 5.26: Computed B_z with $\Delta x^0 = 1/4$ m at $t = 40\Delta t^0 = 1.66782 \times 10^{-8}$ s for the (fixed) high-frequency $\mathbf{J}$. This simulation employs artificial dissipation, but not damping. Note the leading oscillations with wavelength ≈ 1.5 m $= 6\Delta x^0$ outside the refined region. Trapped waves inside the refined region are damped significantly compared to Fig. 5.18 and even Fig. 5.22, but several oscillations outside the refined region are also eliminated.

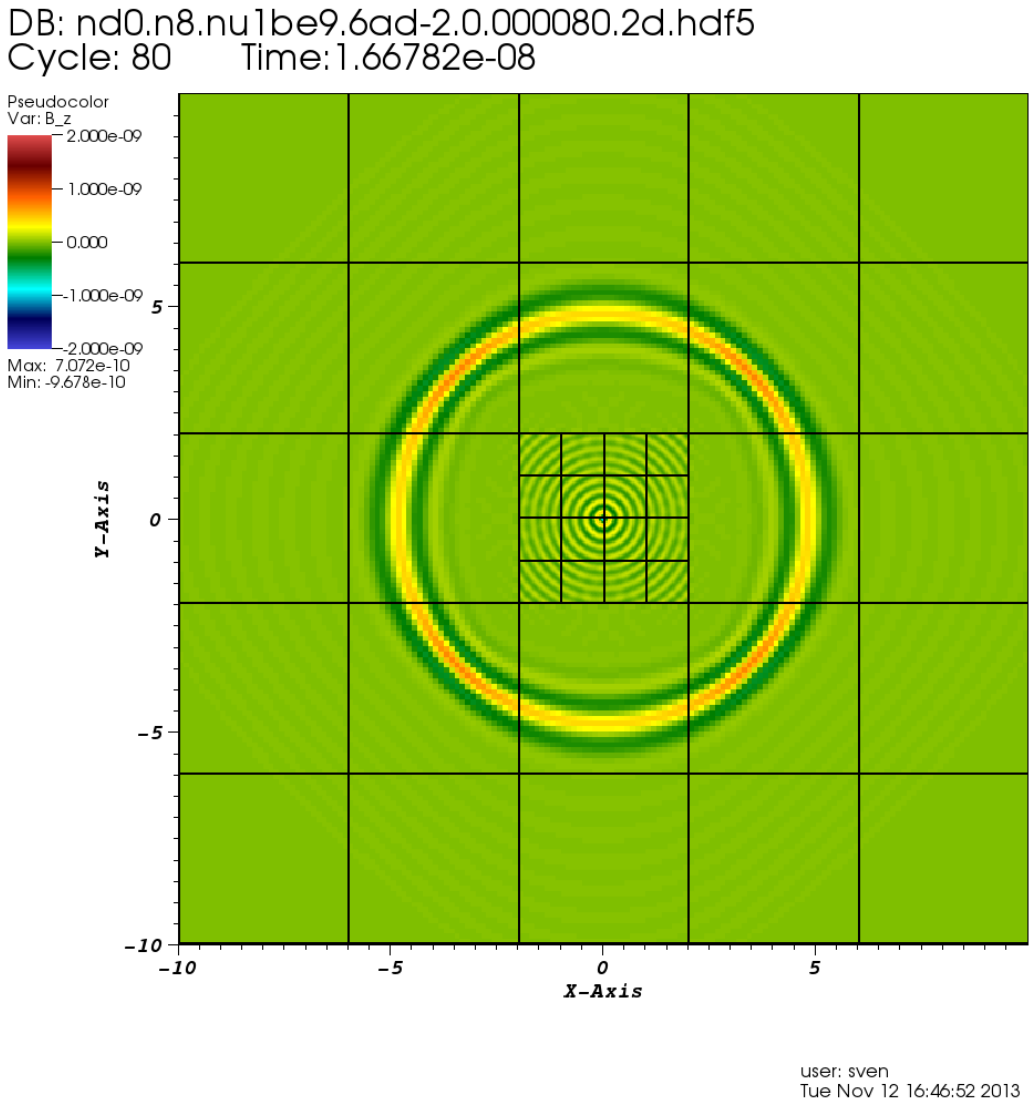

Figure 5.27: Computed B_z with $\Delta x^0 = 1/8$ m at $t = 80\Delta t^0 = 1.66782 \times 10^{-8}$ s for the (fixed) high-frequency $\mathbf{J}$. This simulation employs artificial dissipation, but not damping. Note the leading oscillations with wavelength ≈ 0.75 m $= 6\Delta x^0$ outside the refined region. The refined region appears to be quite well resolved; there are virtually no trapped waves, and the oscillations inside the refined region match those in Figs. 5.28 and 5.29. Once again, however, several oscillations outside the refined region are eliminated.

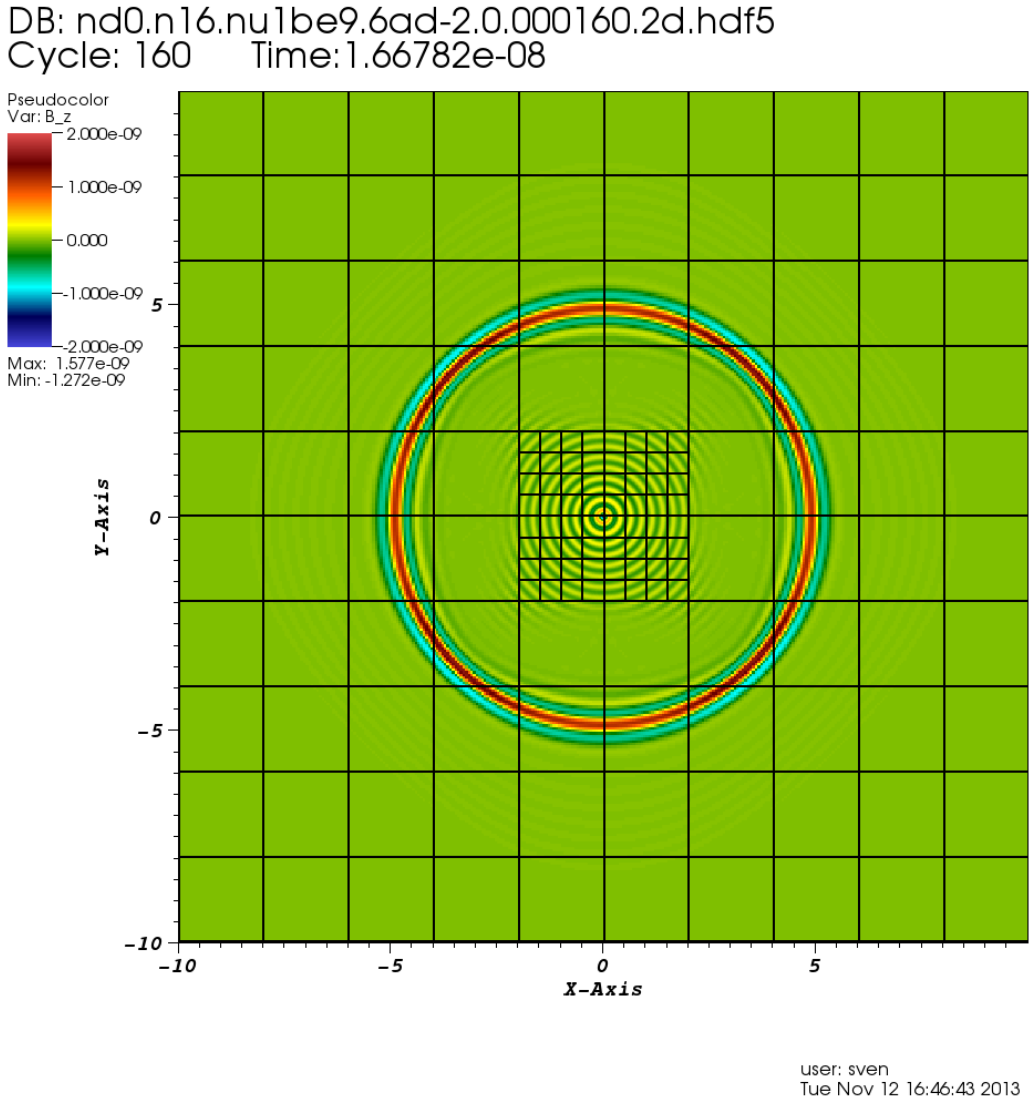

Figure 5.28: Computed B_z with $\Delta x^0 = 1/16$ m at $t = 160\Delta t^0 = 1.66782 \times 10^{-8}$ s for the (fixed) high-frequency $\mathbf{J}$. This simulation employs artificial dissipation, but not damping. Note the leading oscillations with wavelength ≈ 0.375 m $= 6\Delta x^0$ outside the refined region. The refined region appears to be quite well resolved; there are virtually no trapped waves, and the oscillations inside the refined region match those in Fig. 5.29. Once again, however, several oscillations outside the refined region are eliminated.

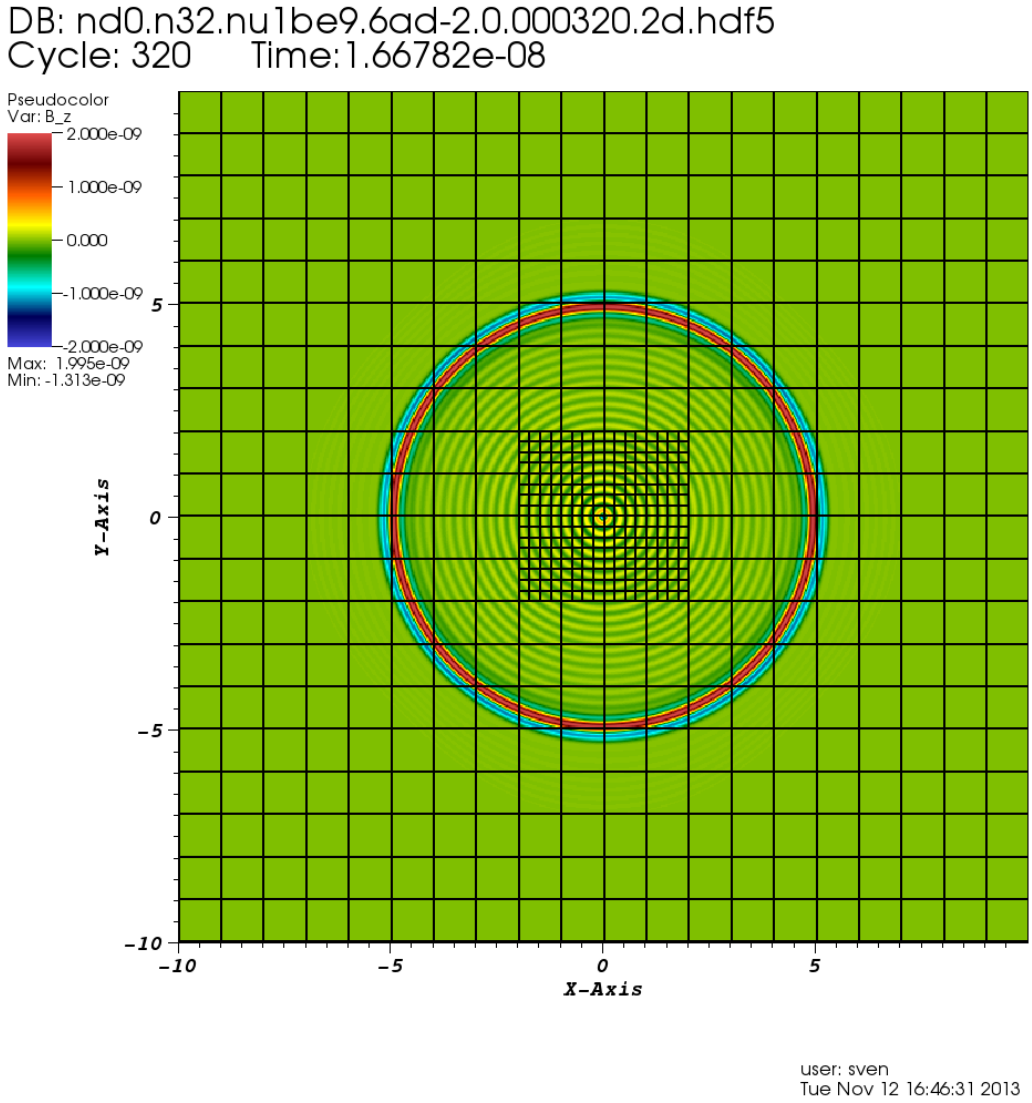

Figure 5.29: Computed B_z with $\Delta x^0 = 1/32$ m at $t = 320\Delta t^0 = 1.66782 \times 10^{-8}$ s for the (fixed) high-frequency $\mathbf{J}$. This simulation employs artificial dissipation, but not damping. We are now properly resolved; the oscillations have a wavelength of 0.25 m $= 8\Delta x^0$, as they should. There are virtually no trapped waves inside the refined region.

5.7 Modulated J simulation results in 2D

For this battery of simulations, we considered

$$\mathbf{J}(x, y, t) = \mathbf{J}_1(x, y) \sin(2\pi\nu_1 t) \sin(2\pi\nu_2 t), \tag{5.60}$$

with the following parameters:

$$\begin{aligned}
J_1 &= 1 \text{ Cm}^{-2}\text{s}^{-1} \text{ (V/m)} \\
a &= \frac{1}{2} \text{ m} \\
x_1 &= 0 \text{ m} \\
y_1 &= 0 \text{ m} \\
\nu_1 &= 149896229 \text{ s}^{-1} \text{ (Hz)} \\
\nu_2 &= 1199169832 \text{ s}^{-1} \text{ (Hz)} \\
h^0 &= \frac{1}{4}, \frac{1}{8}, \frac{1}{16}, \frac{1}{32} \text{ m} \\
h^1 &= \frac{1}{16}, \frac{1}{32}, \frac{1}{64}, \frac{1}{128} \text{ m} \\
\text{CFL} &= \frac{1}{2} \\
\Omega^0 &= [-10, 10] \text{ m} \\
\Omega^1 &= [-2, 2] \text{ m} \\
\text{nesting depth} &= 0, \ 2 \text{ coarse cells} \\
\text{ramping depth} &= 0, \ 2 \text{ coarse cells}
\end{aligned} \tag{5.61}$$

Since

$$\sin a \sin b = \frac{\cos(a-b) - \cos(a+b)}{2}, \tag{5.62}$$

we expect that $\nu_1 + \nu_2$ is the frequency we must resolve before we observe convergence. At $h^0 = 1/4$ m, $\nu_1 + \nu_2 = 9\nu_{\max}/16$, meaning the temporal period of this wave is $16\Delta t^0/9$ and the spatial period is $8h^0/9 = 2/9$ m. To have any hope of resolving the wave properly, we need a spatial period greater than $6h^0$. Thus, we expect $h^0 = 1/27$ m to be the greatest possible coarse spatial step that resolves the problem.

As the figures below show, we see dominating $6h^0$ waves in Ω^0 for $h^0 = 1/4$, $1/8$, and $1/16$ m. Once again, artificial dissipation proves the superior mechanism for eliminating standing waves (as it is supposed to) and trapped waves inside Ω^1, and non-physical $6h^0$ waves that have propagated from Ω^1 to Ω^0.

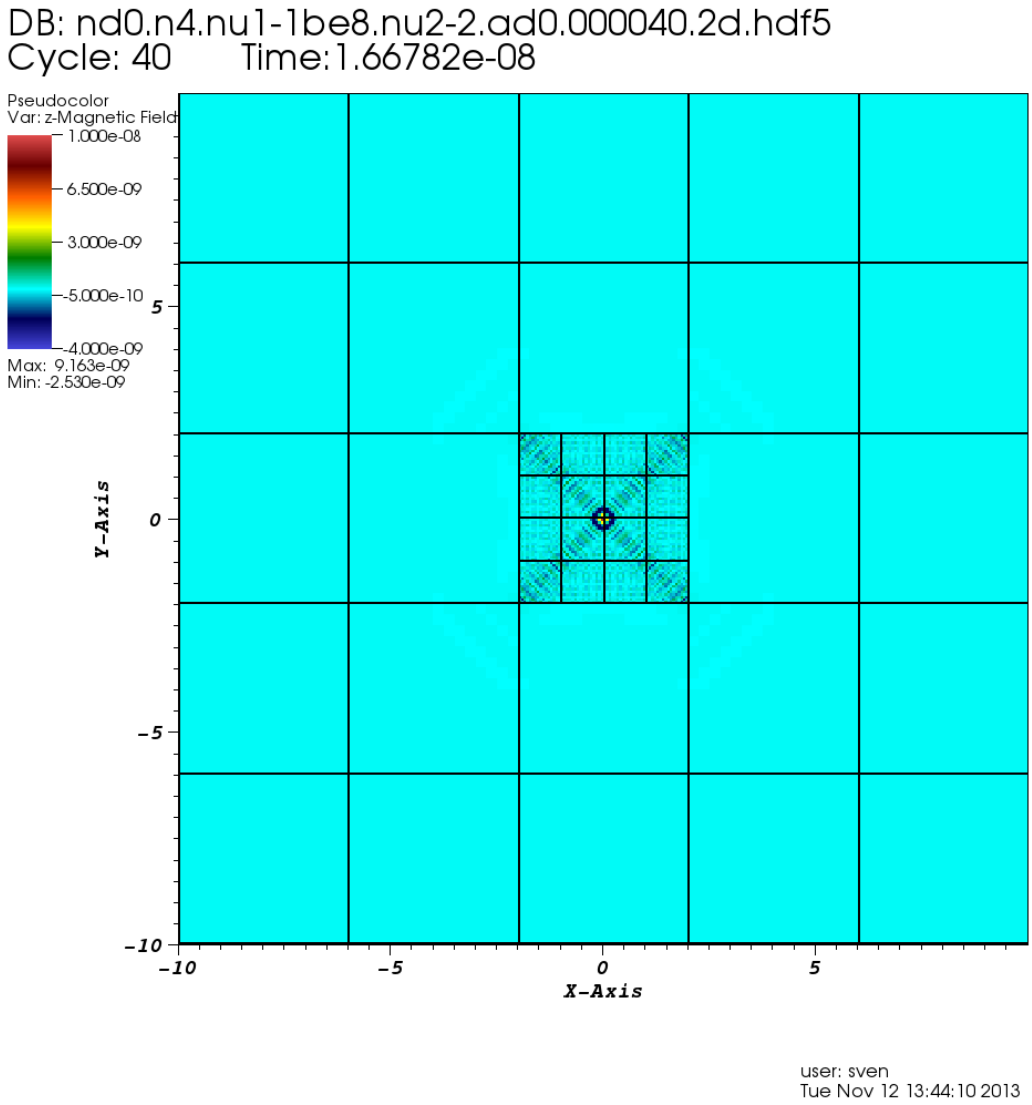

Figure 5.30: Computed B_z with $\Delta x^0 = 1/4$ m at $t = 40\Delta t^0 = 1.66782 \times 10^{-8}$ s for the $\mathbf{J}$ given by Eq. (5.60). This simulation does not employ damping or artificial dissipation. The leading oscillations outside the refined region (which are practically invisible at this scale) have wavelength ≈ 1.5 m $= 6\Delta x^0$. Note abundant trapped and standing waves inside the refined region.

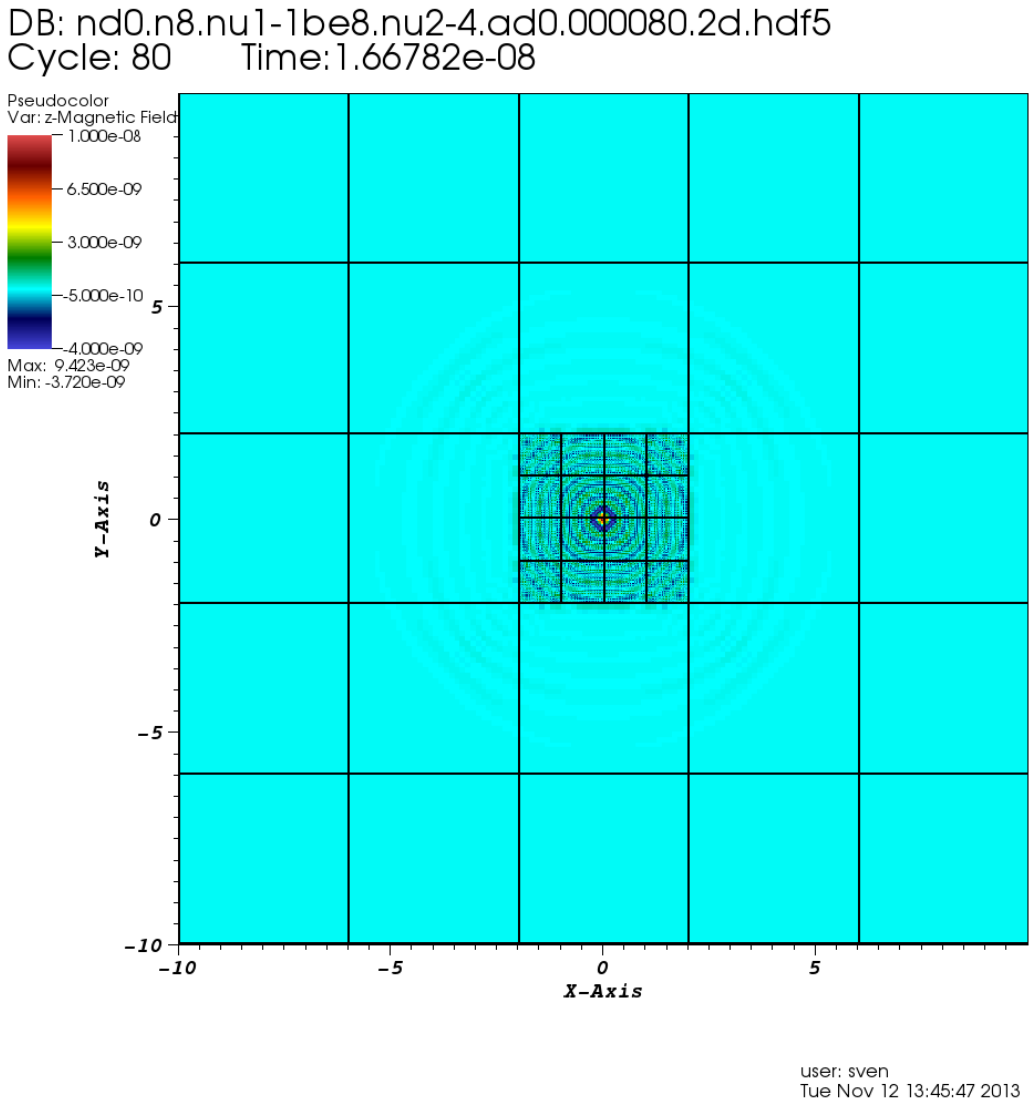

Figure 5.31: Computed B_z with $\Delta x^0 = 1/8$ m at $t = 80\Delta t^0 = 1.66782 \times 10^{-8}$ s for the $\mathbf{J}$ given by Eq. (5.60). This simulation does not employ damping or artificial dissipation. The leading oscillations outside the refined region (which are barely visible at this scale) have wavelength ≈ 0.75 m $= 6\Delta x^0$. Note abundant trapped waves inside the refined region.

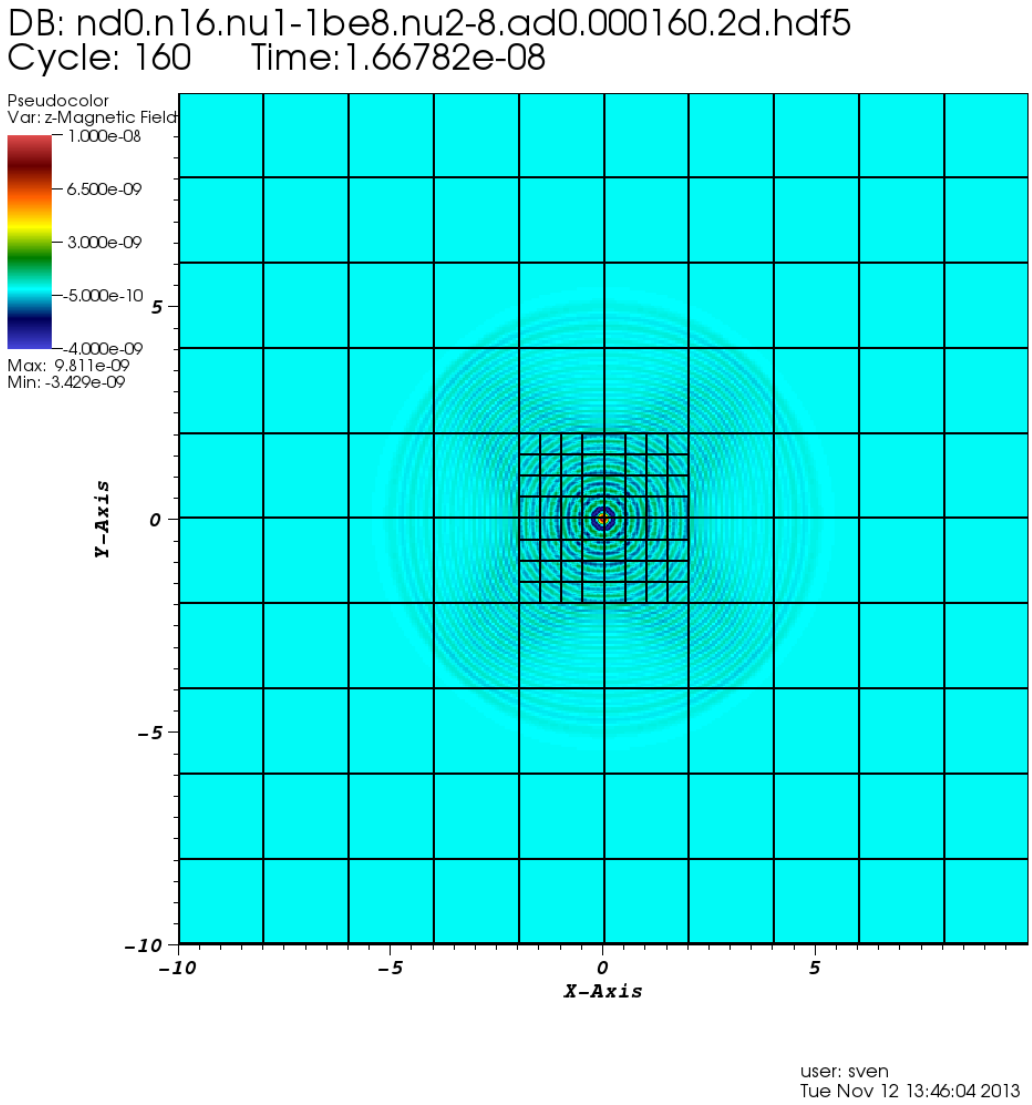

Figure 5.32: Computed B_z with $\Delta x^0 = 1/16$ m at $t = 160\Delta t^0 = 1.66782 \times 10^{-8}$ s for the **J** given by Eq. (5.60). This simulation does not employ damping or artificial dissipation. Note the leading oscillations with wavelength ≈ 0.375 - 0.5 m $= 6 - 8\Delta x^0$ outside the refined region. Trapped waves and standing waves are barely visible.

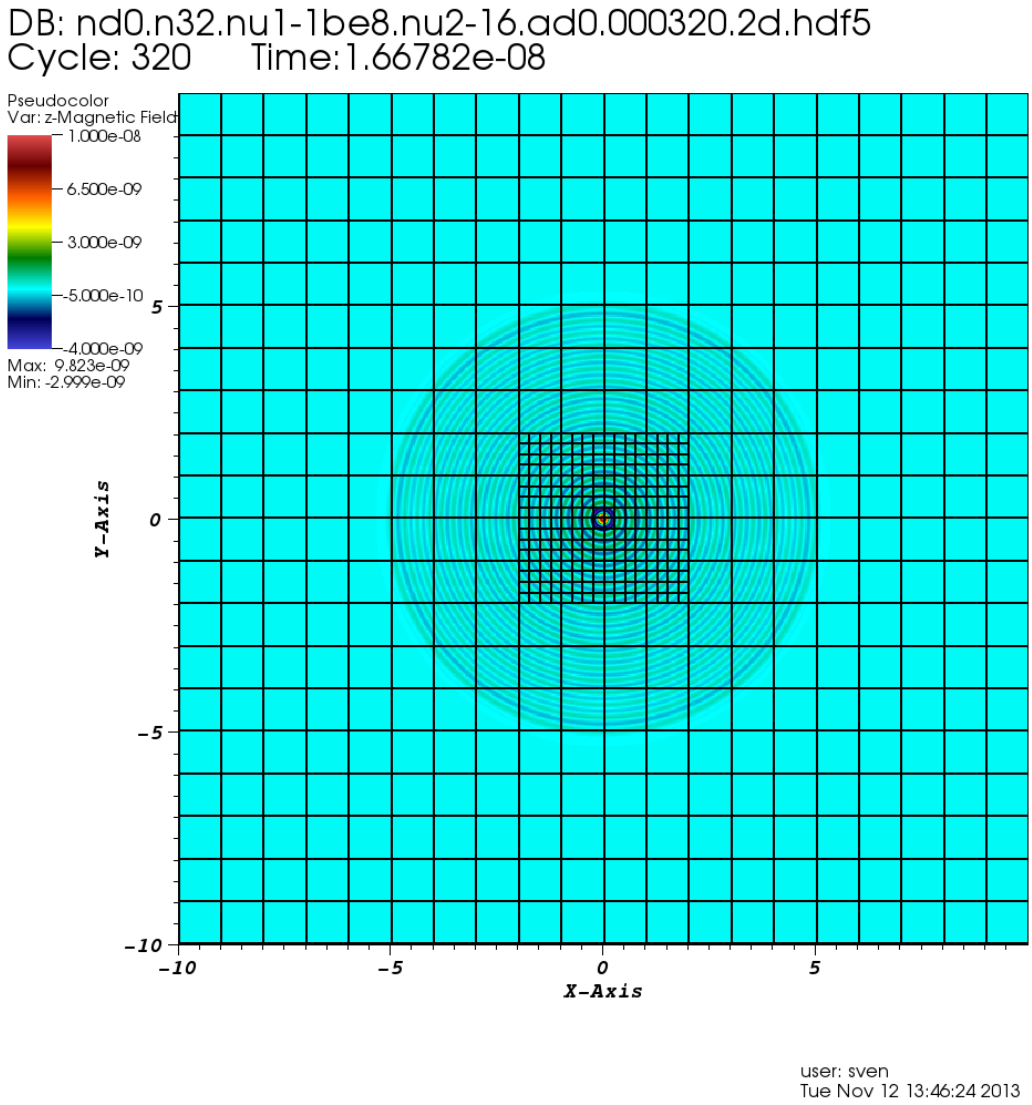

Figure 5.33: Computed B_z with $\Delta x^0 = 1/32$ m at $t = 320\Delta t^0 = 1.66782 \times 10^{-8}$ s for the $\mathbf{J}$ given by Eq. (5.60). This simulation does not employ damping or artificial dissipation. Note the leading oscillations with wavelength $\approx 2/9$ m outside the refined region. Trapped waves and standing waves are virtually nonexistent.

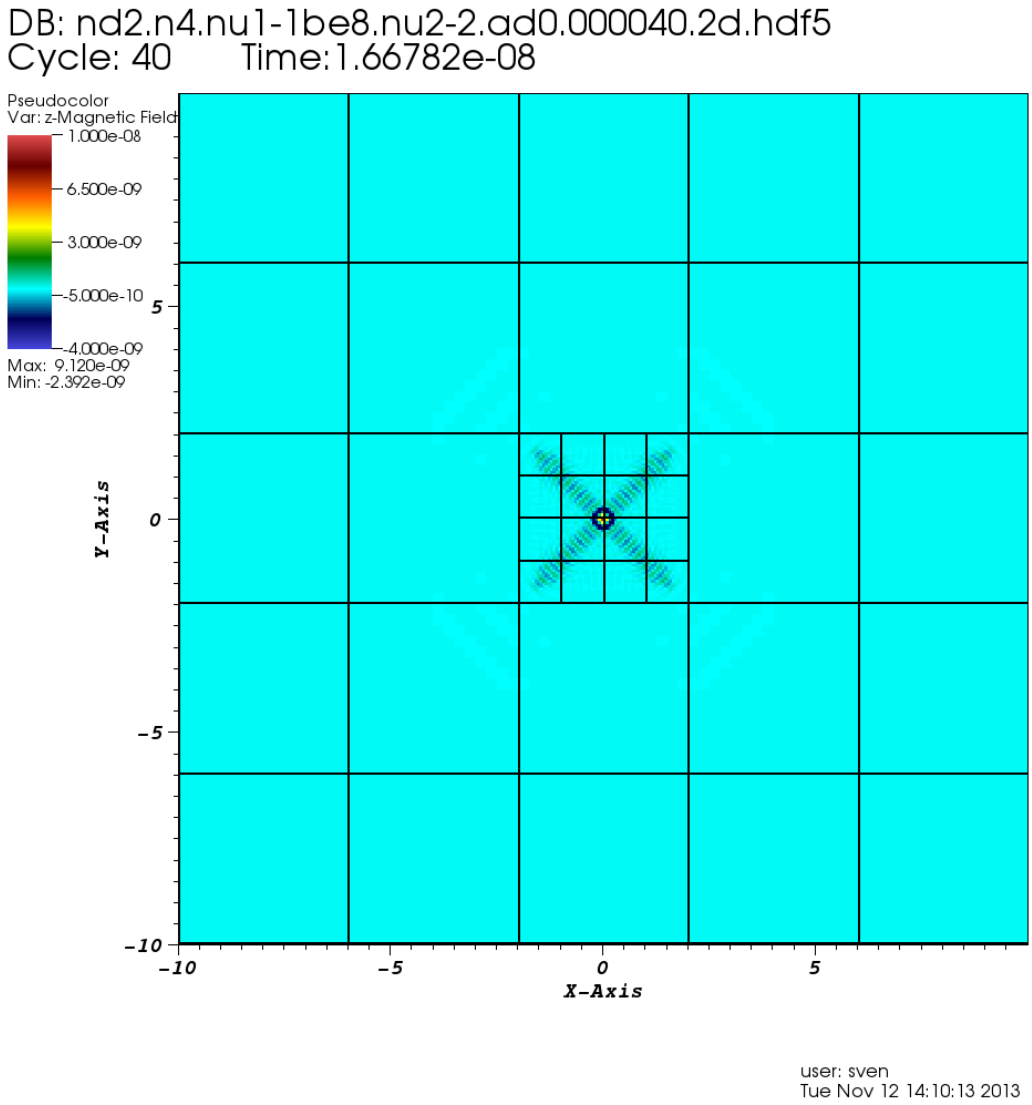

Figure 5.34: Computed B_z with $\Delta x^0 = 1/4$ m at $t = 40\Delta t^0 = 1.66782 \times 10^{-8}$ s for the $\mathbf{J}$ given by Eq. (5.60). This simulation employs damping but not artificial dissipation. The leading oscillations outside the refined region (which are practically invisible at this scale) have wavelength ≈ 1.5 m $= 6\Delta x^0$. Trapped waves are all but eliminated. However, there remain standing waves along the $y = \pm x$ diagonals inside the refined region.

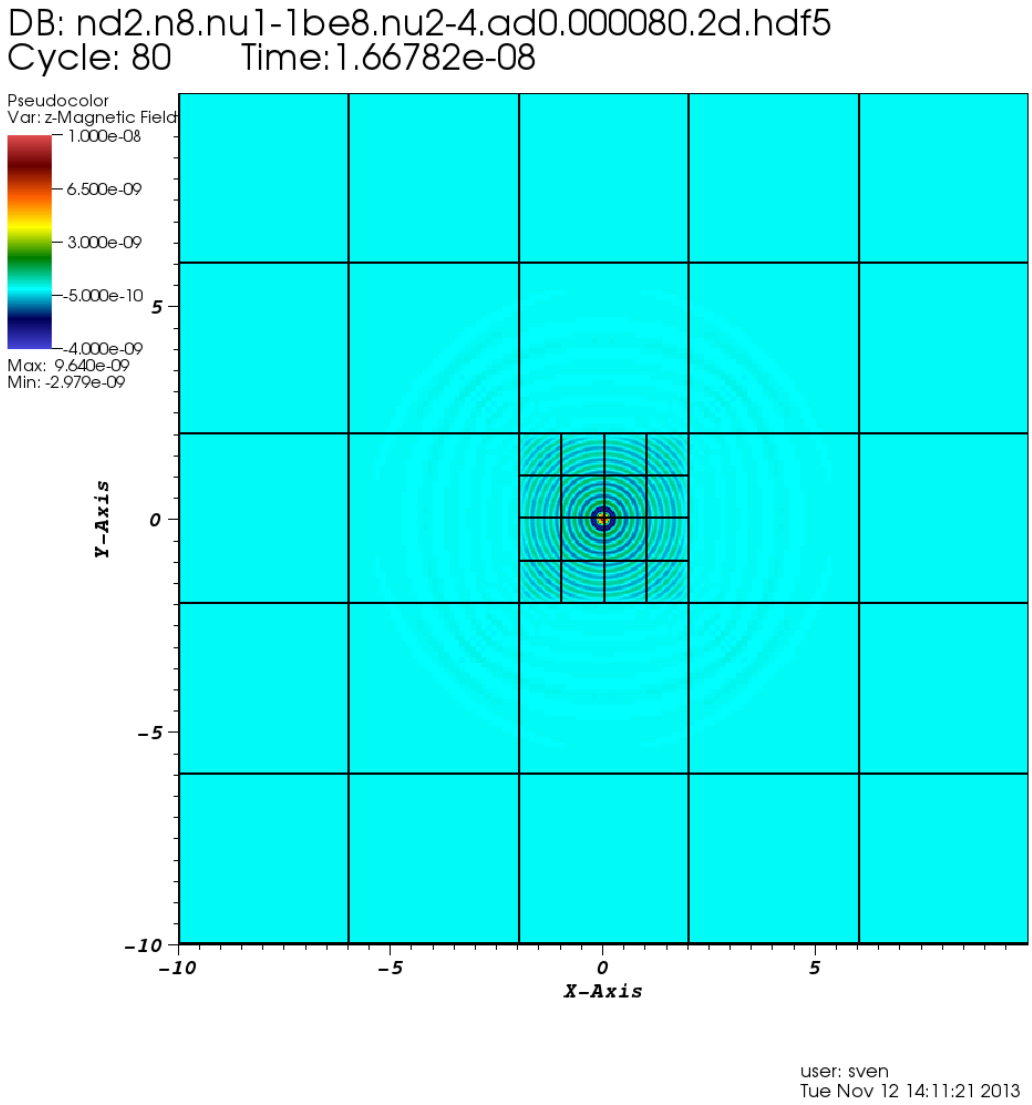

Figure 5.35: Computed B_z with $\Delta x^0 = 1/8$ m at $t = 80\Delta t^0 = 1.66782 \times 10^{-8}$ s for the $\mathbf{J}$ given by Eq. (5.60). This simulation employs damping but not artificial dissipation. The leading oscillations outside the refined region (which are barely visible at this scale) have wavelength ≈ 0.75 m $= 6\Delta x^0$. Trapped waves are all but eliminated; the refined region appears to be reasonably well resolved.

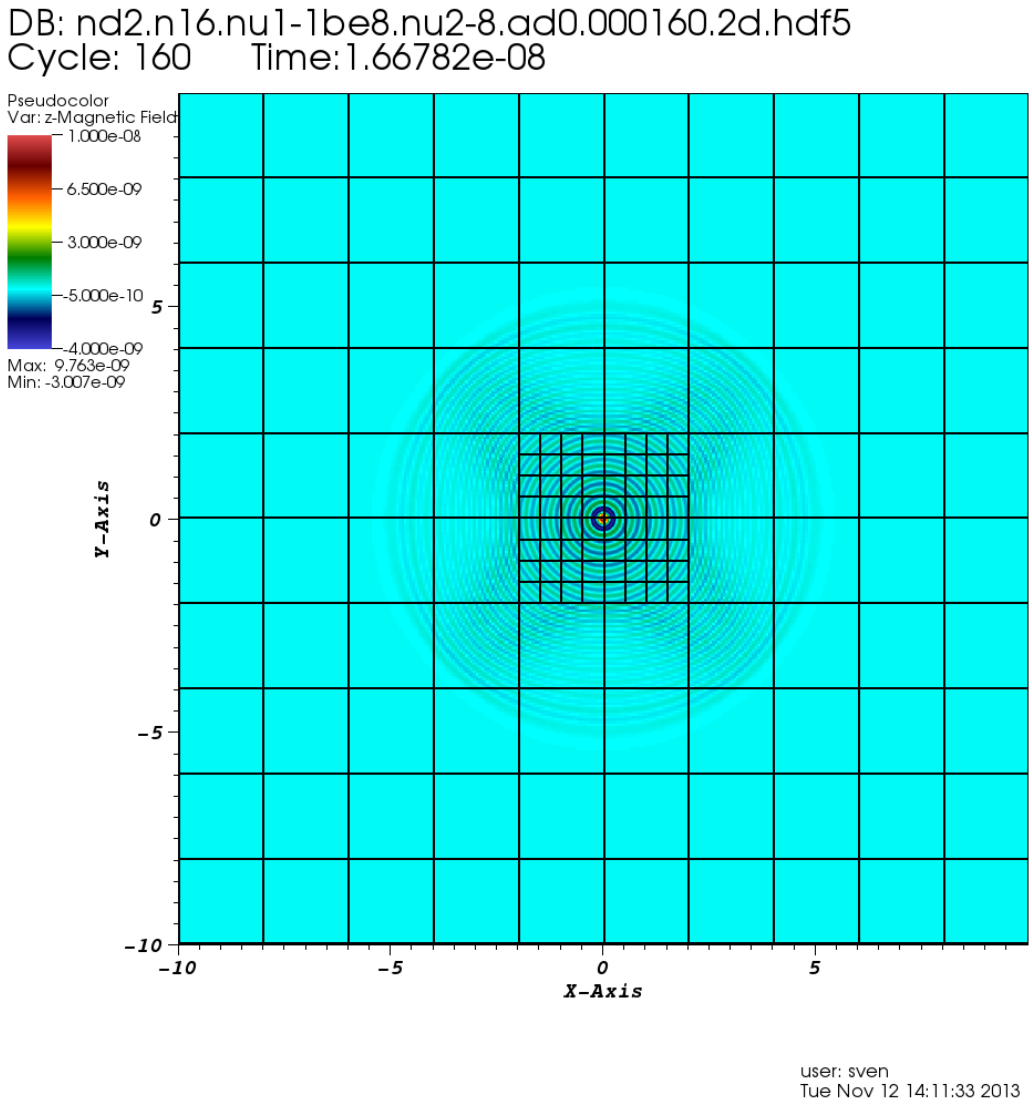

Figure 5.36: Computed B_z with $\Delta x^0 = 1/16$ m at $t = 160\Delta t^0 = 1.66782 \times 10^{-8}$ s for the $\mathbf{J}$ given by Eq. (5.60). This simulation employs damping but not artificial dissipation. Note the leading oscillations with wavelength ≈ 0.375 - 0.5 m $= 6 - 8\Delta x^0$ outside the refined region. Trapped waves are virtually non-existent.

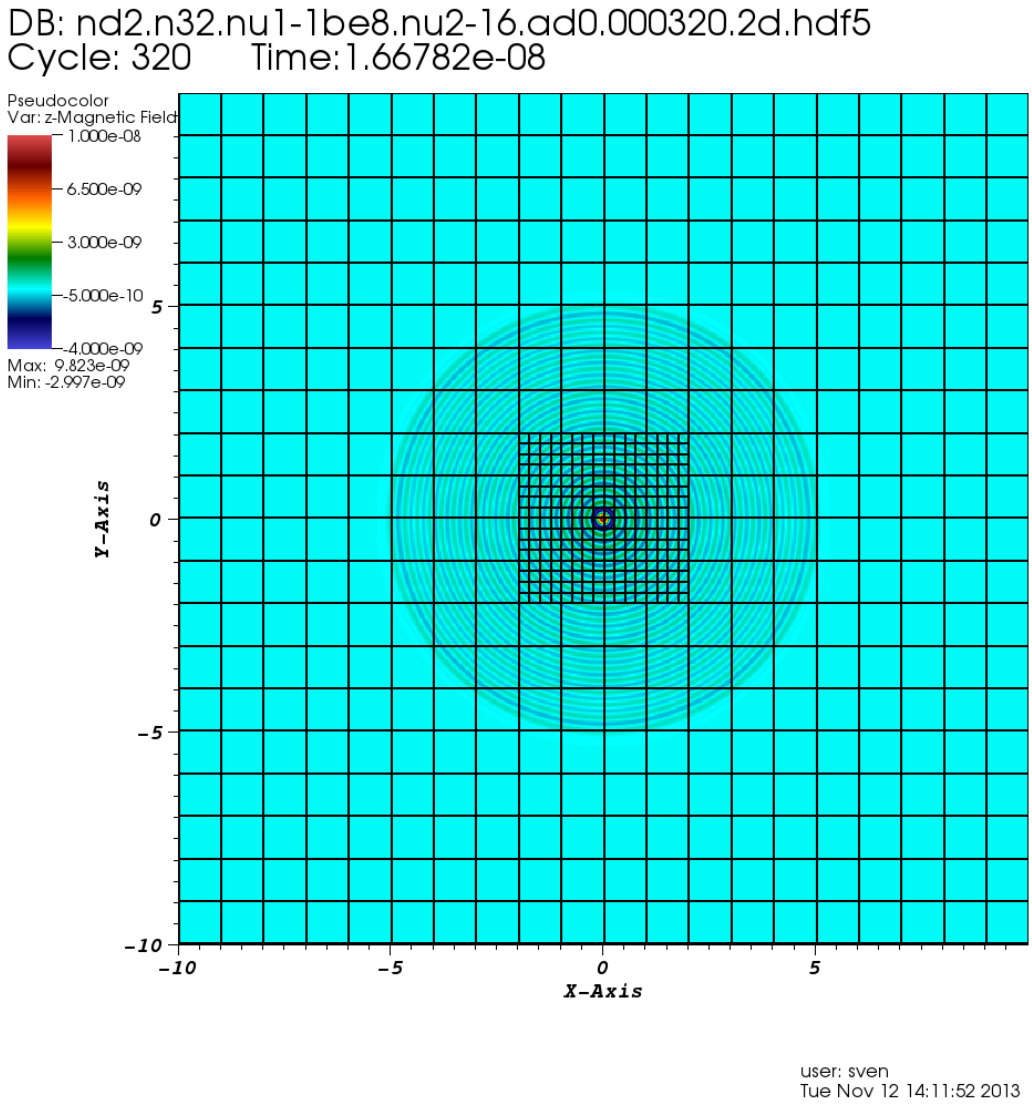

Figure 5.37: Computed B_z with $\Delta x^0 = 1/32$ m at $t = 320\Delta t^0 = 1.66782 \times 10^{-8}$ s for the $\mathbf{J}$ given by Eq. (5.60). This simulation employs damping but not artificial dissipation. Note the leading oscillations with wavelength $\approx 2/9$ m outside the refined region. The simulation appears to be resolved properly.

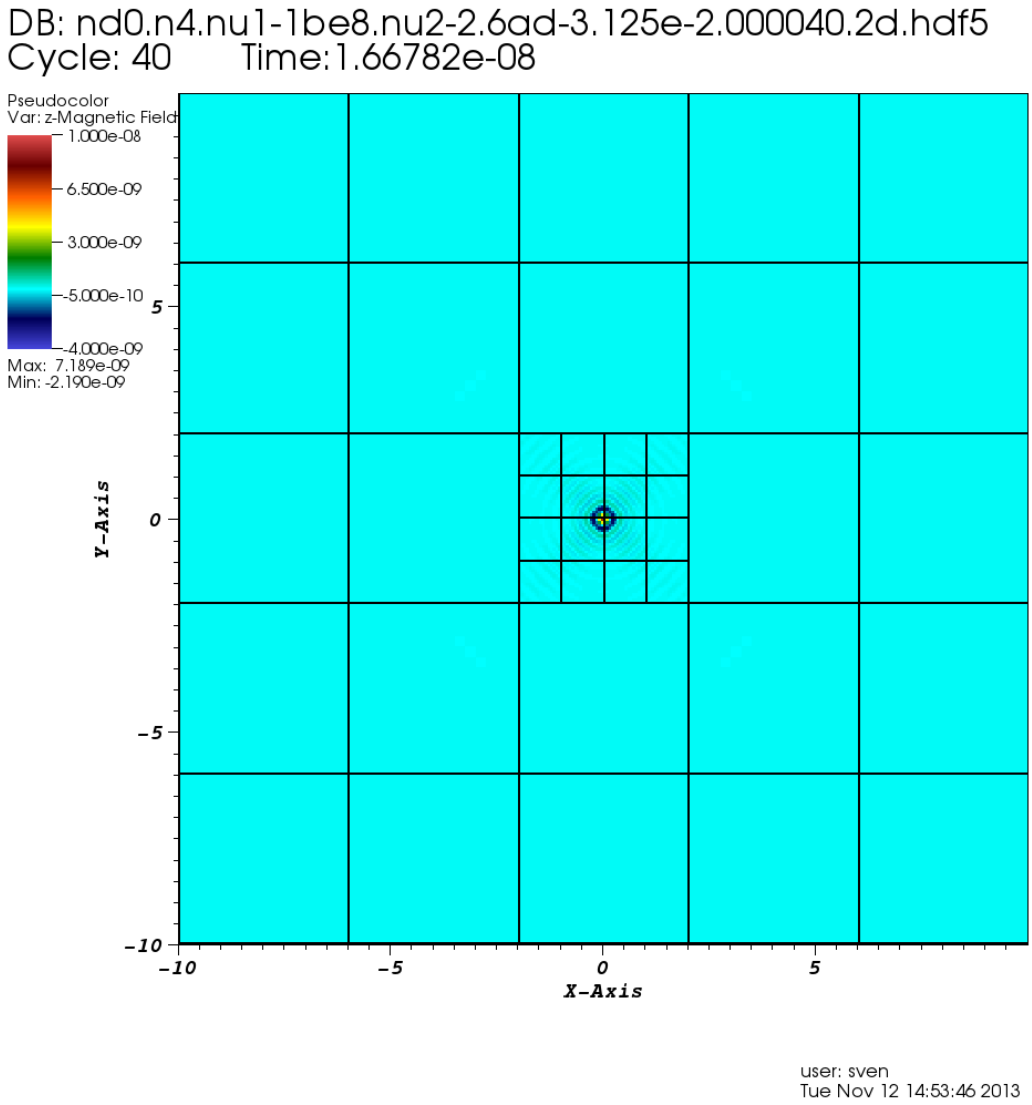

Figure 5.38: Computed B_z with $\Delta x^0 = 1/4$ m at $t = 40\Delta t^0 = 1.66782 \times 10^{-8}$ s for the $\mathbf{J}$ given by Eq. (5.60). This simulation does not employ damping but does employ artificial dissipation. The leading oscillations outside the refined region (which are practically invisible at this scale) have wavelength ≈ 1.5 m $= 6\Delta x^0$. Trapped waves and standing waves are both all but eliminated.

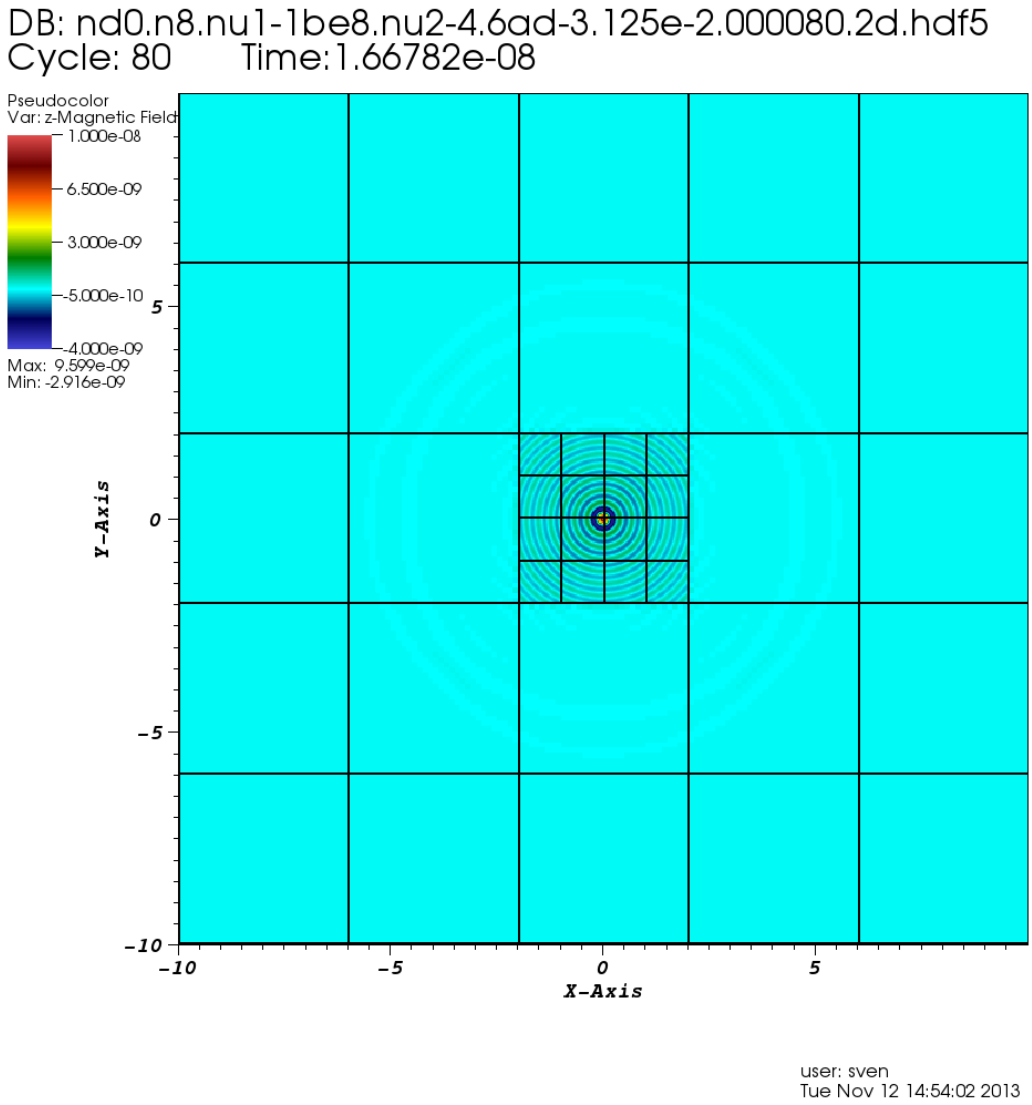

Figure 5.39: Computed B_z with $\Delta x^0 = 1/8$ m at $t = 80\Delta t^0 = 1.66782 \times 10^{-8}$ s for the $\mathbf{J}$ given by Eq. (5.60). This simulation does not employ damping but does employ artificial dissipation. The leading oscillations outside the refined region (which are barely visible at this scale) have wavelength ≈ 0.75 m $= 6\Delta x^0$. Trapped waves are all but eliminated; the refined region appears to be reasonably well resolved. Compared to Fig. 5.35, we do not see as many $6\Delta x^0$ oscillations.

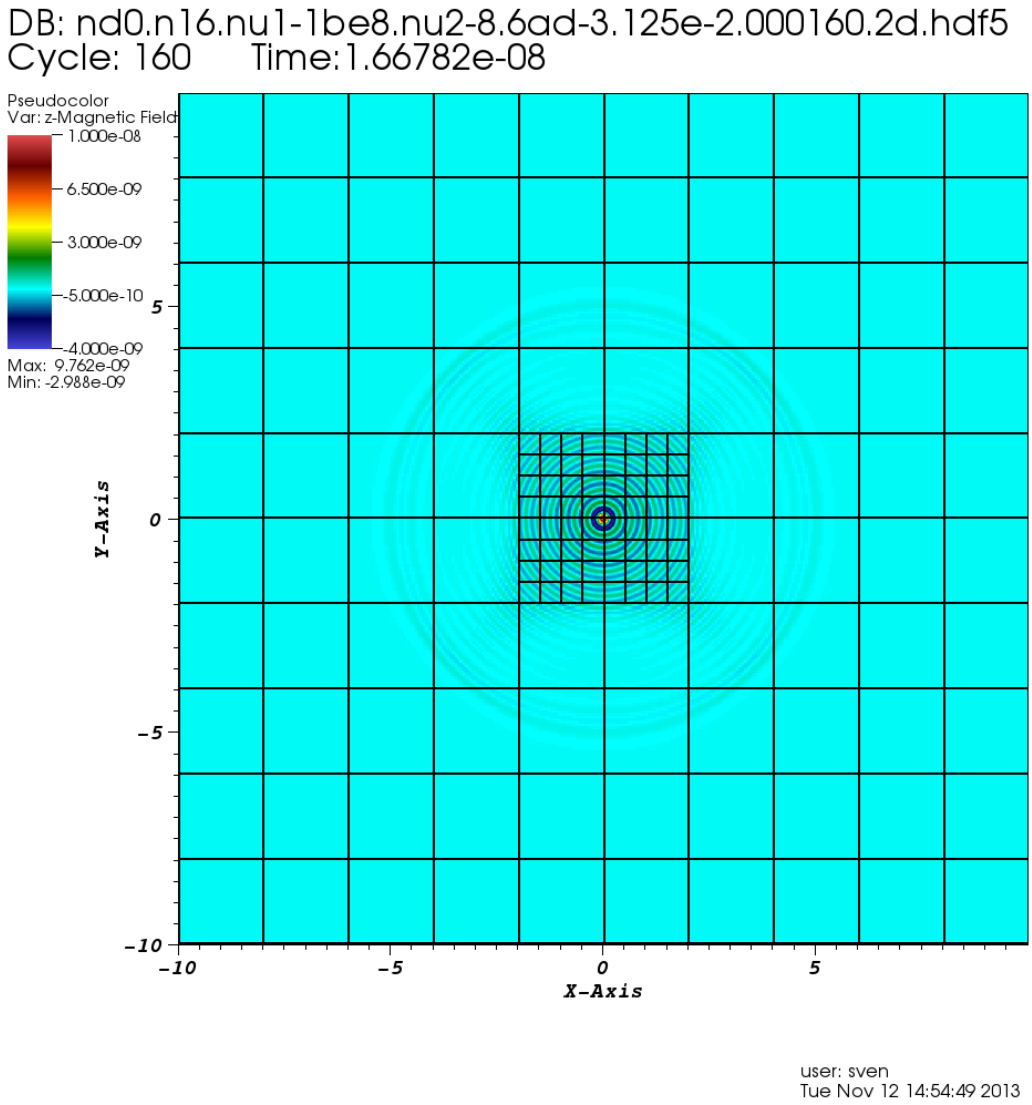

Figure 5.40: Computed B_z with $\Delta x^0 = 1/16$ m at $t = 160\Delta t^0 = 1.66782 \times 10^{-8}$ s for the **J** given by Eq. (5.60). This simulation does not employ damping but does employ artificial dissipation. Note the leading oscillations with wavelength ≈ 0.375 - 0.5 m $= 6-8\Delta x^0$ outside the refined region. Trapped waves are virtually non-existent; the refined region appears to be well resolved. Compared to Fig. 5.36, we do not see as many $6\Delta x^0$ oscillations.

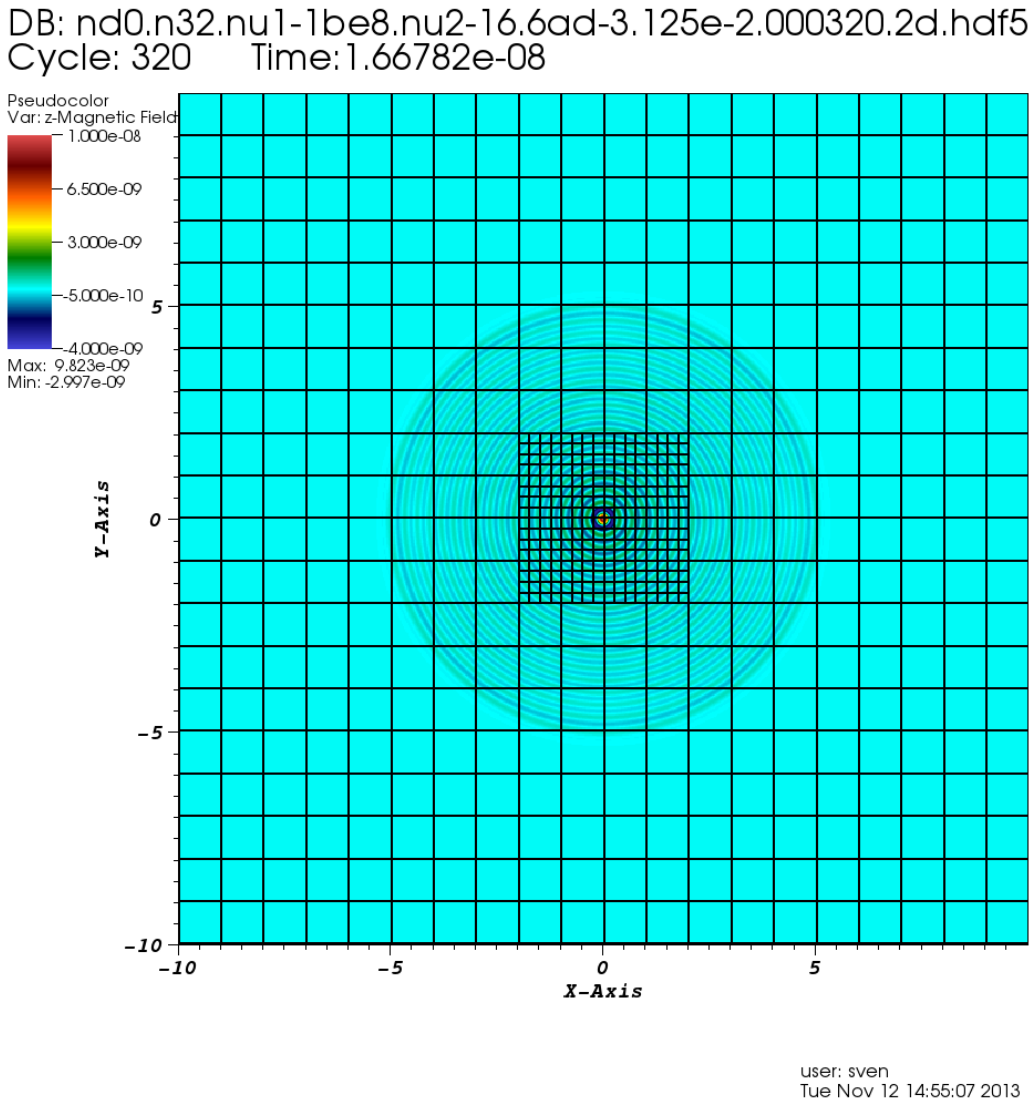

Figure 5.41: Computed B_z with $\Delta x^0 = 1/32$ m at $t = 320\Delta t^0 = 1.66782 \times 10^{-8}$ s for the $\mathbf{J}$ given by Eq. (5.60). This simulation does not employ damping but does employ artificial dissipation. Note the leading oscillations with wavelength $\approx 2/9$ m outside the refined region. The simulation appears to be resolved properly.

5.8 Low-frequency simulation results in 3D

For this battery of simulations, we considered

$$\mathbf{J}(x, y, z, t) = \mathbf{J}_1(x, y, z) \sin(2\pi\nu_1 t) \tag{5.63}$$

for $t \geq 0$ and all field components equal to 0 at $t = 0$.

The relevant parameters are summarized below:

$$\begin{aligned} J_1 &= 1 \text{ Cm}^{-2}\text{s}^{-1} \text{ (V/m)} \\ a &= \frac{1}{2} \text{ m} \\ x_1 &= 0 \text{ m} \\ y_1 &= 0 \text{ m} \\ \nu_1 &= 149896229 \text{ s}^{-1} \text{ (Hz)} \\ h^0 &= \frac{1}{4}, \ \frac{1}{8}, \ \frac{1}{16}, \ \frac{1}{32}, \text{ m} \\ h^1 &= \frac{1}{16}, \ \frac{1}{32}, \ \frac{1}{64}, \ \frac{1}{128}, \text{ m} \\ \text{CFL} &= \frac{1}{2} \\ \Omega^0 &= [-5, 5] \text{ m} \\ \Omega^1 &= [-1, 1] \text{ m} \\ \text{nesting depth} &= 0 \text{ coarse cells} \\ \text{ramping depth} &= 0 \text{ coarse cells} \end{aligned} \tag{5.64}$$

These simulations were run both with and without 6th-derivative artificial dissipation.

ν_1 was specifically chosen as $\nu_{\max}/16$ for $h^0 = 1/4$ m. As such, we expect the oscillations in $\mathbf{J}$ to be quite well resolved at all levels of refinement considered in Eq. (5.64), and, since CFL $= 1/2$, that the spatial period of the oscillations in $\mathbf{J}$ will equal $8h^0$ for $h^0 = 1/4$ m, i.e. the period will equal 2 m.

The convergence properties of two-level, low-resolution simulations with the parameters above are summarized in the tables below. We attempted to run higher-resolution simulations as well, but the runs with $h^0 = 1/64$ m crashed on ravioli.lbl.gov (I believe due to excessive memory requirements), and we had already exceeded our processor-hour allocation on hopper.nersc.gov. As the tables show, these low resolutions do not quite reach the asymptotic regime where convergence rates stabilize at around 4.

Table 5.49: Lower-resolution convergence properties of B_x after one period of $\mathbf{J}$ in two-level simulations without damping and without artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in Tesla (T) = kg s^{-1} C^{-1}.

Norm	1/4 to 1/8	Rate	1/8 to 1/16	Rate	1/16 to 1/32
L_1	1.7753×10^{-9}	1.3904	6.7720×10^{-10}	3.0996	7.9005×10^{-11}
L_2	3.4625×10^{-10}	1.4205	1.2935×10^{-10}	2.9196	1.7095×10^{-11}
L_∞	4.5236×10^{-10}	2.1607	1.0117×10^{-10}	3.0708	1.2041×10^{-11}

Table 5.50: Lower-resolution convergence properties of B_z after one period of **J** in two-level simulations without damping and without artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in Tesla (T) = kg s^{-1} C^{-1}.

Norm	1/4 to 1/8	Rate	1/8 to 1/16	Rate	1/16 to 1/32
L_1	5.9887×10^{-9}	1.2496	2.5187×10^{-9}	2.9992	3.1502×10^{-10}
L_2	1.0762×10^{-9}	1.2170	4.6297×10^{-10}	2.7120	7.0658×10^{-11}
L_∞	9.3019×10^{-10}	1.5284	3.2245×10^{-10}	2.6699	5.0668×10^{-11}

Table 5.51: Lower-resolution convergence properties of E_x after one period of **J** in two-level simulations without damping and without artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in V m^{-1} = kg m s^{-2} C^{-1}.

Norm	1/4 to 1/8	Rate	1/8 to 1/16	Rate	1/16 to 1/32
L_1	1.3409×10^{0}	1.3213	5.3661×10^{-1}	3.0135	6.6450×10^{-2}
L_2	2.5426×10^{-1}	1.3039	1.0298×10^{-1}	2.7015	1.5831×10^{-2}
L_∞	2.2495×10^{-1}	1.8891	6.0733×10^{-2}	1.9959	1.5226×10^{-2}

Table 5.52: Lower-resolution convergence properties of E_z after one period of **J** in two-level simulations without damping and without artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in V m^{-1} = kg m s^{-2} C^{-1}.

Norm	1/4 to 1/8	Rate	1/8 to 1/16	Rate	1/16 to 1/32
L_1	5.3812×10^{-2}	2.4374	9.9349×10^{-3}	3.6501	7.9134×10^{-4}
L_2	1.5685×10^{-2}	2.3048	3.1745×10^{-3}	3.9109	2.1105×10^{-4}
L_∞	3.3095×10^{-2}	1.2087	1.4319×10^{-2}	4.2547	7.5008×10^{-4}

Table 5.53: Lower-resolution convergence properties of B_x after one period of **J** in two-level simulations without damping and with artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in Tesla (T) = kg s^{-1} C^{-1}.

Norm	1/4 to 1/8	Rate	1/8 to 1/16	Rate	1/16 to 1/32
L_1	1.1893×10^{-9}	1.3135	4.7851×10^{-10}	2.7229	7.2478×10^{-11}
L_2	1.9318×10^{-10}	1.0566	9.2874×10^{-11}	2.5065	1.6344×10^{-11}
L_∞	9.9391×10^{-11}	0.95686	5.1204×10^{-11}	2.1622	1.1440×10^{-11}

Table 5.54: Lower-resolution convergence properties of B_z after one period of $\mathbf{J}$ in two-level simulations without damping and with artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in Tesla (T) = kg s^{-1} C^{-1}.

Norm	1/4 to 1/8	Rate	1/8 to 1/16	Rate	1/16 to 1/32
L_1	4.0252×10^{-9}	1.2185	1.7298×10^{-9}	2.5919	2.8691×10^{-10}
L_2	6.0280×10^{-10}	0.87851	3.2788×10^{-10}	2.3043	6.6381×10^{-11}
L_∞	2.0872×10^{-10}	0.39554	1.5867×10^{-10}	1.7528	4.7081×10^{-11}

Table 5.55: Lower-resolution convergence properties of E_x after one period of $\mathbf{J}$ in two-level simulations without damping and with artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in V m^{-1} = kg m s^{-2} C^{-1}.

Norm	1/4 to 1/8	Rate	1/8 to 1/16	Rate	1/16 to 1/32
L_1	8.7443×10^{-1}	1.2366	3.7108×10^{-1}	2.6216	6.0294×10^{-2}
L_2	1.4000×10^{-1}	0.90341	7.4847×10^{-2}	2.3286	1.4900×10^{-2}
L_∞	6.2317×10^{-2}	0.41858	4.6623×10^{-2}	1.7223	1.4130×10^{-2}

Table 5.56: Lower-resolution convergence properties of E_z after one period of $\mathbf{J}$ in two-level simulations without damping and with artificial dissipation. Numbers in the top line are h^0 in units of m. Differences are in V m^{-1} = kg m s^{-2} C^{-1}.

Norm	1/4 to 1/8	Rate	1/8 to 1/16	Rate	1/16 to 1/32
L_1	4.2999×10^{-2}	2.2298	9.1668×10^{-3}	4.0276	5.6208×10^{-4}
L_2	1.3341×10^{-2}	2.2434	2.8174×10^{-3}	4.3751	1.3577×10^{-4}
L_∞	2.6684×10^{-2}	2.4042	5.0411×10^{-3}	5.2250	1.3479×10^{-4}

The convergence properties of $\boldsymbol{\nabla} \cdot \mathbf{B}$ and $\boldsymbol{\nabla} \cdot \mathbf{E}$ are summarized in the tables below:

Table 5.57: Lower-resolution convergence properties of $\boldsymbol{\nabla} \cdot \mathbf{B}$ after one period of $\mathbf{J}$ in two-level simulations without damping and without artificial dissipation. Numbers in the top line are h^0 in units of m. Norm values are in T m^{-1} = kg m^{-1} s^{-1} C^{-1}.

Norm	1/4	Rate	1/8	Rate	1/16
L_1	1.1420×10^{-9}	2.2695	2.3685×10^{-10}	5.3548	5.7879×10^{-12}
L_2	5.4048×10^{-10}	1.5396	1.8592×10^{-10}	4.6640	7.3339×10^{-12}
L_∞	1.9726×10^{-9}	0.90505	1.0534×10^{-9}	3.3793	1.0123×10^{-10}
Norm	1/16	Rate	1/32		
L_1	5.7879×10^{-12}	3.5857	4.8208×10^{-13}		
L_2	7.3339×10^{-12}	2.6857	1.1399×10^{-12}		
L_∞	1.0123×10^{-10}	1.3216	4.0502×10^{-11}		

Table 5.58: Lower-resolution convergence properties of $\boldsymbol{\nabla}\cdot\mathbf{E}$ after one period of $\mathbf{J}$ in two-level simulations without damping and without artificial dissipation. Numbers in the top line are h^0 in units of m. Norm values are in V m^{-2} = kg s^{-2} C^{-1}.

Norm	1/4	Rate	1/8	Rate	1/16
L_1	3.2359×10^{-1}	2.4423	5.9537×10^{-2}	5.3726	1.4371×10^{-3}
L_2	1.5859×10^{-1}	1.5589	5.3828×10^{-2}	4.8857	1.8208×10^{-3}
L_∞	5.7916×10^{-1}	0.85062	3.2117×10^{-1}	3.7074	2.4586×10^{-2}
Norm	1/16	Rate	1/32		
L_1	1.4371×10^{-3}	3.6037	1.1821×10^{-4}		
L_2	1.8208×10^{-3}	2.6610	2.8789×10^{-4}		
L_∞	2.4586×10^{-2}	1.1714	1.0916×10^{-2}		

Table 5.59: Lower-resolution convergence properties of $\boldsymbol{\nabla}\cdot\mathbf{B}$ after one period of $\mathbf{J}$ in two-level simulations without damping and with artificial dissipation. Numbers in the top line are h^0 in units of m. Norm values are in T m^{-1} = kg m^{-1} s^{-1} C^{-1}.

Norm	1/4	Rate	1/8	Rate	1/16
L_1	8.0451×10^{-10}	2.8730	1.0982×10^{-10}	5.9334	1.7970×10^{-12}
L_2	3.1520×10^{-10}	2.0244	7.7476×10^{-11}	5.1357	2.2038×10^{-12}
L_∞	9.2947×10^{-10}	0.093620	8.7107×10^{-10}	3.2629	9.0743×10^{-11}
Norm	1/16	Rate	1/32		
L_1	1.7970×10^{-12}	4.1855	9.8761×10^{-14}		
L_2	2.2038×10^{-12}	3.6307	1.7792×10^{-13}		
L_∞	9.0743×10^{-11}	3.0025	1.1323×10^{-11}		

Table 5.60: Lower-resolution convergence properties of $\boldsymbol{\nabla}\cdot\mathbf{E}$ after one period of $\mathbf{J}$ in two-level simulations without damping and with artificial dissipation. Numbers in the top line are h^0 in units of m. Norm values are in V m^{-2} = kg s^{-2} C^{-1}.

Norm	1/4	Rate	1/8	Rate	1/16
L_1	3.2195×10^{-1}	3.4184	3.0112×10^{-2}	6.0102	4.6720×10^{-4}
L_2	2.2526×10^{-1}	3.2564	2.3572×10^{-2}	5.1393	6.6883×10^{-4}
L_∞	1.1014×10^{0}	2.2212	2.3621×10^{-1}	3.2599	2.4658×10^{-2}
Norm	1/16	Rate	1/32		
L_1	4.6720×10^{-4}	4.2547	2.4475×10^{-5}		
L_2	6.6883×10^{-4}	3.6741	5.2395×10^{-5}		
L_∞	2.4658×10^{-2}	3.0093	3.0625×10^{-3}		

As the tables show, E_x, B_z, and $\boldsymbol{\nabla}\cdot\mathbf{E}$ converge much as in the low-resolution simulations of Sec. 5.5.1 and Sec. 5.5.2. The convergence rates for B_x and $\boldsymbol{\nabla}\cdot\mathbf{B}$ are comparable, while

E_z converges noticeably better. Had our NERSC time allotment not run out, we suspect that the similarities to the 2D simulations would have continued.

Chapter 6

Conclusion

6.1 Discussion

We have achieved at least a significant portion of our objective. The results of the previous two chapters show that given sufficient resolution, our FAM solver is fourth-order accurate. Since Maxwell's Equations form a system of wave equations with sources, we expected that applying sponge layers to the inside of coarse-fine grid boundaries would damp non-physical waves reflected from those boundaries. Our expectations proved correct. When in some of our test problems we observed standing wave instabilities, adding Kreiss- and Oliger-esque artificial dissipation seemed an obvious choice for maintaining stability in our simulated system. Somewhat surprisingly, though, in some test problems, artificial dissipation alone proved a better means to damp non-physical reflected waves than our damping scheme.

6.2 Future work

While we have successfully implemented a fourth-order accurate, adaptive, finite-volume solver for the Faraday-Ampère-Maxwell system, there remains much room for expansion of the work presented in this dissertation.

6.2.1 More complicated geometries and boundary conditions

At present, our Maxwell solver has only been implemented in Cartesian coordinates for problems in rectangular domains with periodic boundary conditions. In some of the test problems in Ch. 4, for which we knew the exact solutions, we imposed Dirichlet boundary conditions on the field components. However, for those problems, we did not observe the desired 4th-order convergence.

Adding the ability to account for non-periodic boundary conditions represents an obvious first step in expanding our Maxwell solver's capabilities. Similarly, one could conceivably

modify the code to allow for problems with more complicated boundaries, which could be solved with embedded boundaries, mapped grids, or curvilinear coordinates.

6.2.2 The Vlasov Equation

The Vlasov Equation describes the time evolution of so-called distribution functions in phase space for a system of charged particles. For a given charge species a (defined by its charge q_a and mass m_a), the (non-relativistic) Vlasov Equation for the distribution function $f_a(\boldsymbol{x}, \boldsymbol{v}, t)$ can be expressed as

$$\left\{ \frac{\partial}{\partial t} + \boldsymbol{v} \cdot \boldsymbol{\nabla} + \frac{q_a}{m_a} \left(\mathbf{E} + \boldsymbol{v} \times \mathbf{B} \right) \cdot \boldsymbol{\nabla}_v \right\} f_a(\boldsymbol{x}, \boldsymbol{v}, t) = 0, \tag{6.1}$$

where $\boldsymbol{\nabla}_v = \left(\frac{\partial}{\partial v_x}, \frac{\partial}{\partial v_y}, \frac{\partial}{\partial v_z} \right)$. The RHS of Eq. (6.1) is 0 because we ignore the effects of particle collisions. Each distribution function can be integrated over velocity space to obtain group-average properties of charge species a. For example, the number density n_a of particles of charge species a is given by

$$n_a(\boldsymbol{x}, t) = \int f_a(\boldsymbol{x}, \boldsymbol{v}, t) d^3 v. \tag{6.2}$$

As is clear from Eq. (6.1), the Vlasov Equation is coupled to Maxwell's Equations through the acceleration term. Furthermore, charge and current densities can be expressed as velocity-space moments of the various distribution functions. Specifically,

$$\rho(\boldsymbol{x}, t) = \sum_a q_a \int f_a(\boldsymbol{x}, \boldsymbol{v}, t) \, d^3 v \tag{6.3}$$

and

$$\mathbf{J}(\boldsymbol{x}, t) = \sum_a q_a \int f_a(\boldsymbol{x}, \boldsymbol{v}, t) \, \boldsymbol{v} \, d^3 v. \tag{6.4}$$

Modeling plasmas with the Maxwell-Vlasov system was the primary motivation for this thesis project. To apply our algorithm to that system, we must consider the representation of distribution functions in phase space. Traditionally, particle-in-cell (PIC) methods have been used for this purpose [Birdsall, 2005]. Another possibility (which would fit into the framework presented here) is to discretize phase space with adaptive grids. This would require a suitable truncation of velocity space. We must also consider the representation of electric fields, charges, and currents, which Sec. 2.2 hints at. In particular, electric fields and currents are separated into divergence-free and curl-free components via Helmholtz decomposition. The curl-free component of the electric field is solved via Poisson's Equation, while the divergence-free component is solved as described in the previous chapters.

Bibliography

[Arfken et al., 2005] Arfken, G. B., Weber, H. J., and Harris, F. E., editors (2005). Mathematical Methods for Physicists. Elsevier Academic Press, London, 6th edition.

[Berenger, 1994] Berenger, J.-P. (1994). A Perfectly Matched Layer for the Absorption of Electromagnetic Waves. Journal of Computational Physics, 114:185–200.

[Berenger, 1996] Berenger, J.-P. (1996). Three-Dimensional Perfectly Matched Layer for the Absorption of Electromagnetic Waves. Journal of Computational Physics, 127:363–379.

[Berger and Colella, 1989] Berger, M. J. and Colella, P. (1989). Local Adaptive Mesh Refinement for Shock Hydrodynamics. Journal of Computational Physics, 82(1):64–84.

[Birdsall, 2005] Birdsall, C. (2005). Plasma Physics via Computer Simulation. CRC Press, 1st edition.

[Butcher, 1964] Butcher, J. C. (1964). On Runge-Kutta processes of high order. Journal of the Australian Mathematical Society, IV:179–194.

[Claridge, 2011] Claridge, J. L. (2011). Introduction to the Chombo User Interface. Technical report, Summer Institute in Taiwan.

[Colella et al., 2012] Colella, P., Graves, D. T., Johnson, J. N., Johansen, H. S., Keen, N. D., Ligocki, T. J., Martin, D. F., McCorquodale, P. W., Modiano, D., Schwartz, P. O., Sternberg, T. D., and Van Straalen, B. (2012). Chombo Software Package for AMR Applications: Design Document. `https://seesar.lbl.gov/anag/chombo/ChomboDesign-3.1.pdf`.

[Davies, 1976] Davies, H. (1976). An unsplit, cell-centered Godunov method for ideal MHD. Quarterly Journal of the Royal Meteorological Society, 102(432):405–418.

[Evans, 2010] Evans, L. C., editor (2010). Partial Differential Equations. American Mathematical Society, Providence, 2nd edition.

[Fok and Rosales, 2008] Fok, P.-W. and Rosales, R. R. (2008). Multirate integration of axisymmetric step-flow equations. `http://arxiv.org/abs/0810.2517v1`.

[Golub and Van Loan, 1996] Golub, G. H. and Van Loan, C. F. (1996). Matrix computations. Johns Hopkins University Press, Baltimore, 3rd edition.

[Hairer et al., 1993] Hairer, E., Nørsett, S. P., and Wanner, G. (1993). Solving Ordinary Differential Equations I: Nonstiff Problems. Springer, Berlin, 2nd edition.

[Hairer and Wanner, 1996] Hairer, E. and Wanner, G. (1996). Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems. Springer, Berlin, 2nd edition.

[Harris and Durran, 2010] Harris, L. M. and Durran, D. R. (2010). An Idealized Comparison of One-Way and Two-Way Grid Nesting. Monthly Weather Review, 138:2174–2187.

[Iserles, 1996] Iserles, A. (1996). A First Course in the Numerical Analysis of Differential Equations. Cambridge University Press, Cambridge, UK, 1st edition.

[Jackson, 1998] Jackson, J. (1998). Classical Electrodynamics. Wiley, 3rd edition.

[Knievel et al., 2007] Knievel, J. C., Bryan, G. H., and Hacker, J. P. (2007). Explicit Numerical Diffusion in the WRF Model. Monthly Weather Review, 135:38083824.

[Kreiss and Oliger, 1972] Kreiss, H.-O. and Oliger, J. (1972). Comparison of accurate methods for the integration of hyperbolic equations. Tellus, 24(3):199–215.

[Kreiss and Oliger, 1973] Kreiss, H.-O. and Oliger, J. (1973). Methods for the Approximate Solution of Time Dependent Problems. Technical Report GARP Publications Series No. 10, Global Atmospheric Research Programme (GARP), World Meteorological Association – International Council of Scientific Unions Joint Organizing Committee.

[LeVeque, 2002] LeVeque, R. J. (2002). Finite Volume Methods for Hyperbolic Problems. Cambridge University Press, Cambridge, UK.

[McCorquodale, 2012] McCorquodale, P. (2012). Fourth-order conservation law example. Available via svn checkout at `anag-repo.lbl.gov/usr/local/svnroot/ChomboDoc/highOrder/fourthOrderHyperbolic/FourthOrderConservation.tex`.

[McCorquodale and Colella, 2011] McCorquodale, P. and Colella, P. (2011). A high-order finite-volume method for conservation laws on locally refined grids. Communications in Applied Mathematics and Computational Science, 6(1):1–25.

[Monk and Süli, 1994] Monk, P. and Süli, E. (1994). A Convergence Analysis of Yee's Scheme on Nonuniform Grids. SIAM Journal of Numerical Analysis, 31(2):393–412.

[Samtaney et al., 2004] Samtaney, R., Jardin, S. C., Colella, P., and Martin, D. (2004). 3D adaptive mesh refinement simulations of pellet injection in tokamaks. Computer Physics Communications, 164:220–228.

[Skamarock et al., 2005] Skamarock, W., Klemp, J., Dudhia, J., Gill, D., Barker, D., Wang, W., and Powers, J. (2005). A Description of the Advanced Research WRF Version 2. Technical Report ADA487419, National Center for Atmospheric Research. NCAR Tech Notes-468+ STR.

[Vay, 2000] Vay, J.-L. (2000). A New Absorbing Layer Boundary Condition for the Wave Equation. Journal of Computational Physics, 165:511–521.

[Vay, 2001] Vay, J.-L. (2001). An Extended FDTD Scheme for the Wave Equation: Application to Multiscale Electromagnetic Simulation. Journal of Computational Physics, 167:72–98.

[Vay, 2002] Vay, J.-L. (2002). Asymmetric Perfectly Matched Layer for the Absorption of Waves. Journal of Computational Physics, 183:367–399.

[Vay et al., 2004a] Vay, J.-L., Colella, P., Friedman, A., Grote, D. P., McCorquodale, P., and Serafini, D. B. (2004a). Implementations of mesh refinement schemes for Particle-In-Cell plasma simulations. Computer Physics Communications, 164:297305.

[Vay et al., 2004b] Vay, J.-L., Colella, P., Kwan, J. W., McCorquodale, P., Serafini, D. B., Friedman, A., Grote, D. P., Westenskow, G., Adam, J.-C., Heron, A., and Haber, I. (2004b). Application of adaptive mesh refinement to particle-in-cell simulations of plasmas and beams. Physics of Plasmas, 11:2928.

[Vay et al., 2002] Vay, J.-L., Colella, P., McCorquodale, P., Van Straalen, B., Friedman, A., and Grote, D. P. (2002). Mesh Refinement for Particle-in-Cell Plasma Simulations: Applications to and Benefits for Heavy Ion Fusion. Laser and Particle Beams, 20(4):569–575.

[Xue, 2000] Xue, M. (2000). High-Order Monotonic Numerical Diffusion and Smoothing. Monthly Weather Review, 128:28532864.

[Yee, 1966] Yee, K. (1966). A Fourth-Order Accurate Finite-Volume Method with Structured Adaptive Mesh Refinement for Solving the Advection-Diffusion Equation. IEEE Transactions Antennas and Propagation, AP-14(3):302–307.