An Electromagnetic Hybridizable Discontinuous Galerkin Method Forward Solver with High-Order Geometry for Inverse Problems

Nicholas Geddert

A Thesis submitted to the Faculty of Graduate Studies of the University of Manitoba in partial fulfillment of the requirements for the degree of

Master of Science

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

Abstract

This thesis focuses on the theory, development, and implementation of a time-harmonic hybridizable discontinuous Galerkin method forward solver. This algorithm is capable of representing the physics and geometry of the problem as high-order polynomial expansions. It computes the scattered electric and magnetic fields on unstructured grids with high-order accuracy, and supports boundary conditions and inhomogeneous backgrounds. The high-order capabilities improve convergence which are examined for both synthetic and experimental problems. Furthermore, the algorithm has been accelerated for modern computing architectures allowing it to scale to large problem sizes. The forward solver is integrated into an existing contrast source inversion algorithm used for radiowave and microwave imaging which improves the modeling capabilities and computational demand. Results of the hybridizable discontinuous Galerkin method forward solver are presented, which show improved capabilities and performance over existing solvers.

Contributions

This work is based upon existing code. My contributions include the following:

- The development of a 3D Hybridizable Discontinuous Galerkin Method (HDGM) forward solver C++ code, which includes implementation of the following features:
  - High-order representation of geometry
  - High-order constitutive and field expansions
  - Quadrature-free and quadrature-based integration rules
  - Simulation of electric and/or magnetic fields
  - Inhomogeneous backgrounds
  - Three types of boundary termination conditions: absorbing, perfect electrical conductor, and impedance
  - Formulation of HDGM using an electric field basis, which has performance benefits in metallic chambers
  - Algorithm acceleration using parallel architectures and graphic processing units
- The implementation of a HDGM forward solver into an existing contrast source inversion (CSI) algorithm, including:
  - Derivation of matrix operators required to use the HDGM forward solver within CSI
  - Multifrequency features for CSI using frequency hopping, simultaneous frequency, or dynamic frequency weighting

Provided below is a list of the published works resulting from this research.

Refereed Journal Papers

1. Keeley Edwards, **Nicholas Geddert**, Kennedy Krakalovich, Ryan Kruk, Joe LoVetri, Colin Gilmore, and Ian Jeffrey. "Stored Grain Inventory Management Using Neural-Network-Based Parametric Electromagnetic Inversion". IEEE Access, (Accepted 2020).

2. Colin Gilmore, Ian Jeffrey, Mohammad Asefi, **Nicholas Geddert**, Kevin Brown, and Joe LoVetri. "Phaseless Parametric Inversion for System Calibration and Obtaining Prior Information". IEEE Access, vol. 7, 2019.

3. Kevin Brown, **Nicholas Geddert**, Mohammad Asefi, Joe LoVetri, and Ian Jeffrey. "Hybridizable Discontinuous Galerkin Method Contrast Source Inversion of 2-D and 3-D Dielectric and Magnetic Targets". IEEE Transactions on Microwave Theory and Techniques, vol. 67, no. 5, pp. 1766-1777, 2019.

4. Ian Jeffrey, **Nicholas Geddert**, Kevin Brown, and Joe LoVetri. "The Time-Harmonic Discontinuous Galerkin Method as a Robust Forward Solver for Microwave Imaging Applications". Progress in Electromagnetics Research, vol. 154, pp. 1-21, 2015.

Provisional Patents

1. Colin Gilmore, Ian Jeffrey, Joe LoVetri, Mohammad Asefi, **Nicholas Geddert**, and Kevin Brown. "Electromagnetic Imaging and Inversion of Simple Parameters in Storage Bins". August 2019.

2. Keeley Edwards, Ian Jeffrey, Joe LoVetri, Mohammad Asefi, **Nicholas Geddert**, Kennedy Krakalovich, and Ryan Kruk. "Stored Grain Inventory Neural Network". August 2020.

Refereed Conference Papers

1. **Nicholas Geddert**, Liam Sparling, Joe LoVetri, Ian Jeffrey. "Frequency-Weighting for Multi-Frequency Electromagnetic Source Contrast Inversion". 2018 International Workshop on Computing, Electromagnetics, and Machine Intelligence (CEMi), November 21 - 24, 2018.

2. **Nicholas Geddert**, Ian Jeffrey. "A Dynamically Balanced OpenMP-CUDA Implementation of PDE-Based Contrast Source Inversion for Microwave Imaging". 2018

18th International Symposium on Antenna Technology and Applied Electromagnetics (ANTEM), August 19 - 22, 2018.

3. Kevin Brown, Mohammad Asefi, **Nicholas Geddert**, Ian Jeffrey, Joe LoVetri. "An Electric and Magnetic Field Transceiver Configuration Study for a Biomedical Microwave Imaging System". URSI Atlantic Radio Science Meeting, May 28 - June 1, 2018.

4. Cam Kaye, Kevin Brown, Mohammad Asefi, **Nicholas Geddert**, Ian Jeffrey, and Joe LoVetri. "Experimental 3D Microwave Imaging of Magnetic Targets using Discontinuous Galerkin Contrast Source Inversion and a Two-Stage Reconstruction Technique". URSI Atlantic Radio Science Meeting, May 28 - June 1, 2018.

5. Joe LoVetri, Mohammad Asefi, Nasim Abdollahi, Kevin Brown, **Nicholas Geddert**, and Ian Jeffrey. "Phantom Study of a Flat-Faceted Air-Based Microwave Breast Imaging System". European Conference on Antennas and Propagation, April 9 - 13, 2018.

6. Ian Jeffrey, Joe LoVetri, Kevin Brown, **Nicholas Geddert**, Mohammad Asefi, Kyle Nemez, Anastasia Baran, and Gabriel Faucher. "3D PDE-based Contrast Source Inversion for Biomedical and Agricultural Applications". Radio Science Meeting (General Assembly), August 19 - 26, 2017. Paper ID: 2072.

7. Kevin Brown, **Nicholas Geddert**, Gabriel Faucher, Anastasia Baran, Mohammad Asefi, Kyle Nemez, Joe LoVetri, and Ian Jeffrey. "3D Microwave Imaging using the Time-Harmonic Discontinuous Galerkin Method - Contrast Source Inversion". Radio Science Meeting (Joint with AP-S Symposium), July 9 - 14, 2017. Paper ID: 2561.

8. **Nicholas Geddert**, Kevin Brown, Ian Jeffrey. "Improving Discontinuous Galerkin Method Contrast Source Inversion Microwave Imaging using a Hybridizable Forward Solver Formulation". Radio Science Meeting (Joint with AP-S Symposium), July 9 - 14, 2017. Paper ID: 2524.

9. Kevin Brown, **Nicholas Geddert**, Ian Jeffrey. "A Mixed Discontinuous Galerkin Formulation for Time-Harmonic Scattering Problems". International Symposium on Antenna Technology and Applied Electromagnetics, July 10 - 13, 2016. Paper ID: 116.

Abbreviations and Symbols

Abbreviation	Description
2D	Two Dimensional
3D	Three Dimensional
ABC	Absorbing Boundary Condition
CSI	Contrast Source Inversion
DGM	Discontinuous Galerkin Method
DOF	Degrees of Freedom
EIL	Electromagnetic Imaging Lab
FEM	Finite Element Method
HDGM	Hybridizable Discontinuous Galerkin Method
IBC	Impedance Boundary Condition
OI	Object of Interest
MWI	Microwave Imaging
PDE	Partial Differential Equation
PEC	Perfect Electric Conductor
PMC	Perfect Magnetic Conductor

Symbols

In this work we use the following notation:

- Vector functions are denoted with an arrow over uppercase or lowercase non-bold letters e.g. $\vec{E}(\vec{x}), \vec{f}'(\vec{x})$
- Vectors of scalar coefficients are denoted with an underlined non-bold letter e.g. $\underline{E}_x$
- Matrices are denoted with bold uppercase letters, such as $\boldsymbol{\mathcal{K}}$
- Vectors of spatial quantites at different points are denoted with bold lowercase letters, such as $\boldsymbol{b}$

Symbol	Description
Geometry	
$\hat{x}, \hat{y}, \hat{z}$	Unit vectors for the x, y, and z coordinates
$\vec{x}$	Cartesian position vector consisting of $\hat{x}, \hat{y}, \hat{z}$ components
$\hat{r}, \hat{s}, \hat{t}$	Unit vectors for the barycentric r, s, and t coordinates
$\vec{r}$	Barycentric position vector consisting of $\hat{r}, \hat{s}$, and $\hat{t}$ components
Ω	The entire computational domain
Ω_N	Numerical trace of the entire computational domain
V_n	A single arbitrary element of the decomposed domain
S_n	The surface of V_n
$\hat{n}(\vec{x})$	Outward unit normal to surface S_n at $\vec{x}$
B_n	The corresponding barycentric element of V_n
∂B_n	The boundary of B_n
g	Polynomial expansion order for geometry
Physics	
f	Frequency of excitation
ω	Angular frequency $= 2\pi f$
j	Imaginary unit $= \sqrt{-1}$
p	Polynomial expansion order
$\varepsilon(\vec{x})$	Permittivity at $\vec{x}$
$\mu(\vec{x})$	Permeability at $\vec{x}$
$\vec{E}(\vec{x})$	Electric field vector at $\vec{x}$
$\vec{H}(\vec{x})$	Magnetic field vector at $\vec{x}$
Operators	
$\nabla f(\vec{r})$	Gradient of scalar function f at $\vec{r}$
$\nabla \cdot \vec{F}(\vec{r})$	Divergence of vector function $\vec{F}$ at $\vec{r}$
$\nabla \times \vec{F}(\vec{r})$	Curl of vector function $\vec{F}$ at $\vec{r}$
$\vec{E}^*$	Complex conjugate of $\vec{E}$
$(\boldsymbol{F})^H$	Hermitian operator of discrete matrix $\boldsymbol{F}$
$\lvert\boldsymbol{F}\rvert$	Determinant of discrete matrix $\boldsymbol{F}$

Acknowledgements

First and foremost I thank my advisor, Dr. Ian Jeffrey, for giving me this opportunity to broaden my knowledge and understanding. Your guidance, wisdom, and support have been invaluable to me throughout my graduate studies.

I would like to thank my colleagues from the EIL, whose hard work and insight have paved the way for my research. Thank you Dr. Mohammad Asefi, Dr. Colin Gilmore, Gabriel Faucher, Kyle Nemez, and Cameron Kaye. A special thanks to Dr. Joe LoVetri, the head of the EIL, for his continual knowledge and dedication to the field.

I would also like to thank Kevin Brown, who has been at my side since we began working together as undergraduate researchers. Without his collaboration, the past 6 years of research and development simply would not be possible. Thank you for your knowledge and friendship.

I extend my thanks to my M.Sc. examining committee, Dr. Shaun Lui and Dr. Vladimir Okhmatovski, for taking time to read and evaluate this thesis.

This work would not have been possible without the financial support from Natural Sciences and Engineering Research Council of Canada, the University of Manitoba, and 151 Research Inc.

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

Table of Contents

List of Figures

List of Tables

1

Introduction

This thesis presents the theory and development of a time-harmonic 3D Hybridizable Discontinuous Galerkin Method (HDGM) for use in Microwave Imaging (MWI) algorithms. HDGM is used to numerically solve the discrete Partial Differential Equation (PDE) form of Maxwell's equations in the presence of complex permittivity and permeability. HDGM allows for high-order representation of the constitutive parameters, high-order geometry on an unstructured grid, and a high-order solution to the electric and/or magnetic fields. While the standard Discontinuous Galerkin Method (DGM) is capable of supporting these features, HDGM reduces the number of Degrees of Freedom (DOF) in the forward operator while maintaining the same degree of accuracy in the solution. A high-order Finite Element Method (FEM) could also support these features, but requires a complex formulation. The purpose of this work is to improve the performance of Contrast Source Inversion (CSI) imaging algorithms by utilizing the benefits of the HDGM forward solver.

1.1 Imaging

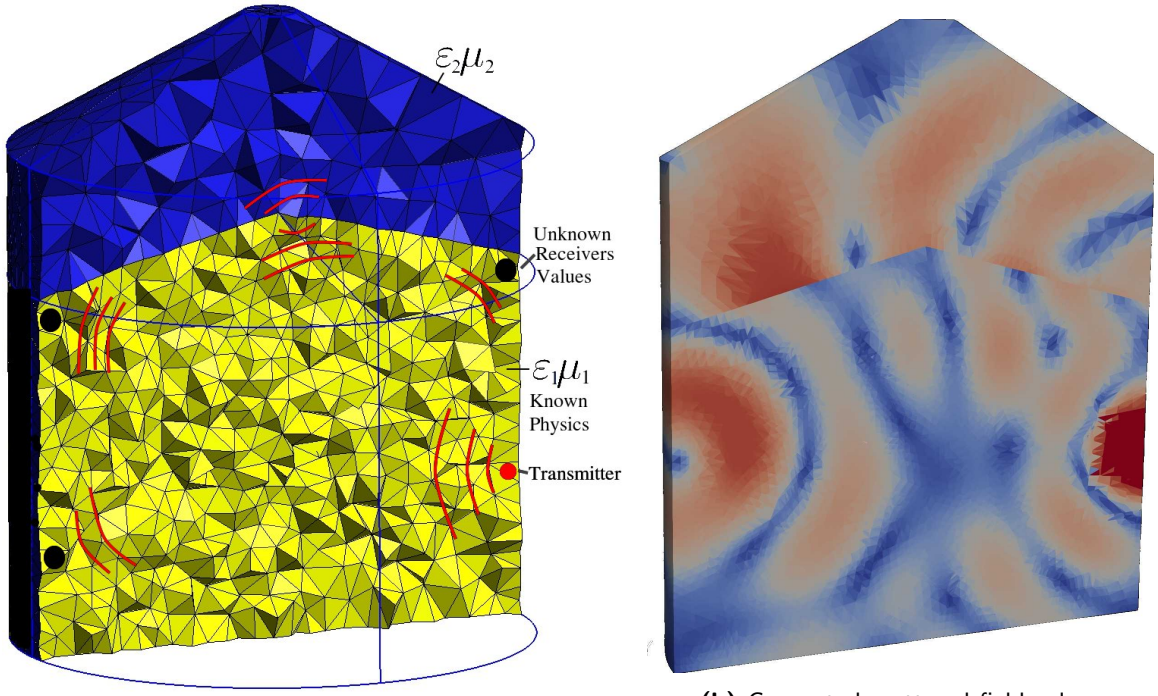
The Electromagnetic Imaging Lab (EIL) at the University of Manitoba has spent several years studying inverse problem in electromagnetics using microwave technology [1, 2]. Benefits of MWI over other imaging techniques include its relatively low cost, and use of non-ionizing radiation [3]. However, the large computational demand of MWI is a major drawback, which is why the benefits of HDGM are of interest.

MWI attempts to reconstruct the constitutive parameters, herein limited to the complex permittivity and permeability, of an Object of Interest (OI). The OI is surrounded with antennas which sequentially transmit electromagnetic waves. These waves scatter off the OI and are measured by the non-transmitting antennas. Data from these measured values are used by inversion algorithms which computes estimates of the constitutive parameters by simulating the forward problem and minimizing the field difference at the antenna locations. Inverse problems are both ill-posed and nonlinear, making these types of problems difficult to solve.

Critical to this thesis is the forward solver used by inversion algorithms which numerically simulates the electromagnetic field scattering in the presence of the constitutive estimate. Reconstruction quality is improved if the forward solver supports arbitrary boundary shapes, high-order constitutive profiles, high-order field solutions, inhomogeneous backgrounds, electric and/or magnetic field antennas, as well as absorbing, perfectly conducting, and impedance boundary conditions [4, 5]. Ideally the forward solver does this while reducing computational cost.

The EIL and their industrial parner, 151 Research Inc., are interested in the application of MWI for the detection of spoiled grain inside grain bins [6]. This thesis will use this

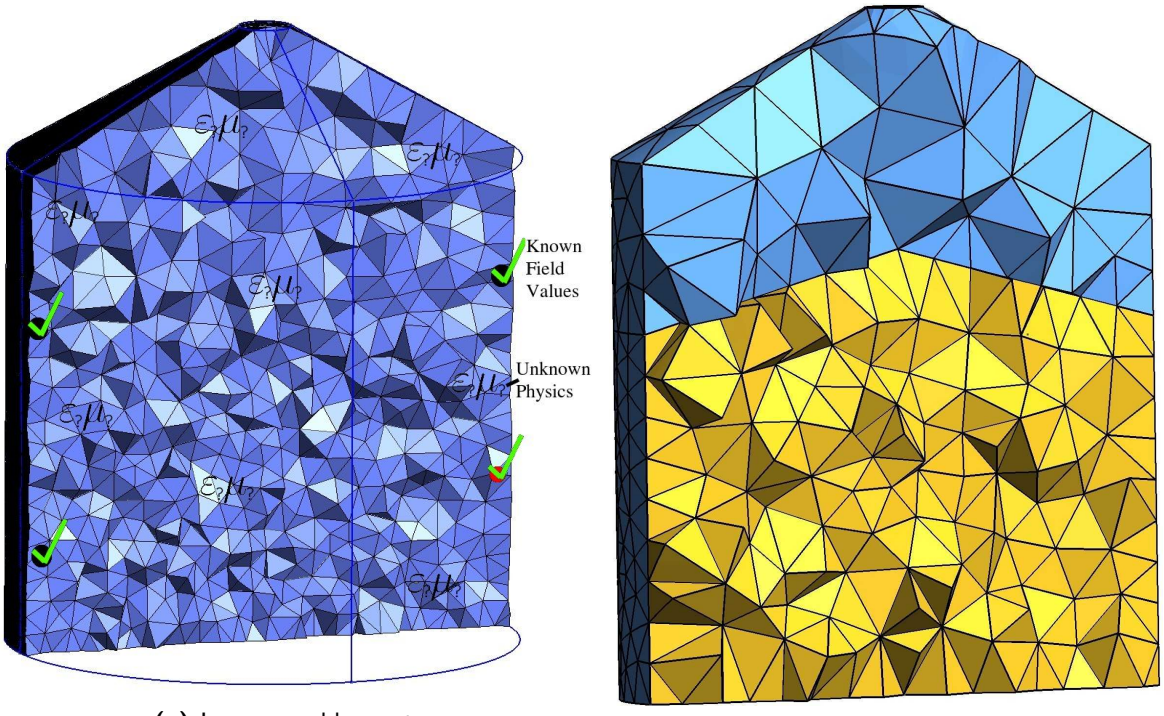
application in a number of examples. Figure 1.1 a) shows the forward problem in a half-filled grain bin from which we gather the scattered fields shown in Figure 1.1 b). The inverse problem which makes use of these values is shown in Figure 1.2 and is discussed in Chapter 7.



(a) Forward problem setup

(b) Computed scattered field values

Fig. 1.1: Forward problem for a half-filled grain bin in which the physics are known and we aim to compute the scattered field values.



(a) Inverse problem setup

(b) Estimate of the constitutive parameters

Fig. 1.2: Inverse problem for a grain bin where the fields at the antenna locations are known, and we aim to determine the constitutive parameters.

1.2 Thesis Overview

The scope of this thesis is the theory, implementation, and results obtained from a 3D HDGM forward solver with distributed parallel support for use in imaging algorithms. The previously mentioned advantages of the HDGM forward solver, including high-order expansion orders for fields, geometry, and constitutive profiles will be examined for both synthetic and experimental 3D problems. Chapter 2 begins with Maxwell's equations and formulates them in a way which can be solved numerically. Chapter 3 covers the basis and elements, which are used to numerically simulate these equations. We examine how the DGM formulates a global system of equations which can be solved to obtain the unknown electric and magnetic field values in Chapter 4. Chapter 5 introduces the HDGM formulation, and discusses advantages and disadvantages. The implementation details of HDGM are discussed in Chapter 6, which includes a section on steps taken to ensure fast computations on modern computing architectures. Chapter 7 then covers a few key details on a CSI algorithm used for imaging, including steps required to make use of the HDGM forward solver. Chapter 8 validates the HDGM forward solver, along with its features and use in CSI, and compares it to existing forward solver algorithms. Lastly, Chapter 9 concludes and summarizes the thesis, and discusses future work which can further improve performance.

2

Electromagnetic Concepts

We can scarcely avoid the conclusion that light consists in the transverse undulations of the same medium which is the cause of electric and magnetic phenomena.

-James Maxwell, 1862

James Maxwell forever changed the world when he formulated the relation between electricity, magnetism, and light [7]. His equations, which date back to 1865, are the foundation of this thesis [8]. Numerical methods are used to simulate these equations in order to capture electromagnetic phenomena. In Section 2.1 we examine Maxwell's equations and make connections with physical parameters of materials. Incident and scattered field components are defined in Section 2.2, which are used in Section 2.3 to formulate a strong form flux-split equation capable of simulating elecotrmagnetic waves. The theory developed in this chapter serves as the groundwork for the formulation of HDGM in Chapter 5.

2.1 Maxwell's Equations

We begin with Maxwell's equations in (2.1) which define electromagnetic phenomena [7]. In these equations, t represents time, ∂_t is the time derivative, $\vec{x}$ is the Cartesian position vector, $\vec{E}$ and $\vec{H}$ are the electric and magnetic field vectors, $\vec{D}$ and $\vec{B}$ are the electric and magnetic flux densities, ρ_E and ρ_M are the electric and magnetic charge densities, $\vec{J}^{\,i}$ and $\vec{M}^i$ are the impressed electric and magnetic sources, and $\vec{J}^{\,c}$ and $\vec{M}^c$ are the electric and magnetic conduction sources. We ignore condution charges as we assume our problems do not contain these.

$$\begin{aligned}
\nabla \times \vec{E}(t,\vec{x}) &= -\partial_t \vec{B}(t,\vec{x}) - \vec{M}^i(t,\vec{x}) - \vec{M}^c(t,\vec{x}) \\
\nabla \times \vec{H}(t,\vec{x}) &= \partial_t \vec{D}(t,\vec{x}) + \vec{J}^{\,i}(t,\vec{x}) + \vec{J}^{\,c}(t,\vec{x}) \\
\nabla \cdot \vec{D}(t,\vec{x}) &= \rho_E(t,\vec{x}) \\
\nabla \cdot \vec{B}(t,\vec{x}) &= \rho_M(t,\vec{x})
\end{aligned} \tag{2.1}$$

We also assume time and frequency invariant isotropic media for which the following constitutive relations hold:

$$\begin{aligned}
\vec{D}(t,\vec{x}) &= \varepsilon_0 \varepsilon_r^{'}(\vec{x})\, \vec{E}(t,\vec{x}) \\
\vec{B}(t,\vec{x}) &= \mu_0 \mu_r^{'}(\vec{x})\, \vec{H}(t,\vec{x}) \\
\vec{J}^{\,c} &= \sigma_e(\vec{x}) \vec{E}(t,\vec{x}) \\
\vec{M}^c &= \sigma_m(\vec{x}) \vec{H}(t,\vec{x}).
\end{aligned} \tag{2.2}$$

Here, ε_0 and μ_0 are the permittivity and permeability of free space, $\varepsilon_r^{'}(\vec{x})$ and $\mu_r^{'}(\vec{x})$ are the relative permittivity and relative permeability at position $\vec{x}$, and $\sigma_e(\vec{x})$ and $\sigma_m(\vec{x})$

are electric and magnetic conductivities. These constitutive relations are used to re-write Maxwell's equations as:

$$\begin{aligned}
\nabla \times \vec{E}(t,\vec{x}) &= -\partial_t(\mu_0 \mu_r^{'}(\vec{x})\, \vec{H}(t,\vec{x})) - \vec{M}^i(t,\vec{x}) - \vec{M}^c(t,\vec{x}) \\
\nabla \times \vec{H}(t,\vec{x}) &= \partial_t(\varepsilon_0 \varepsilon_r^{'}(\vec{x})\, \vec{E}(t,\vec{x})) + \vec{J}^{\,i}(t,\vec{x}) + \vec{J}^{\,c}(t,\vec{x}) \\
\nabla \cdot \vec{D}(t,\vec{x}) &= \rho_E(t,\vec{x}) \\
\nabla \cdot \vec{B}(t,\vec{x}) &= \rho_M(t,\vec{x}).
\end{aligned} \tag{2.3}$$

A time-harmonic dependency of e^{jwt} is assumed, where $j = \sqrt{-1}$ is the imaginary unit and $\omega = 2\pi f$ is the angular frequency of frequency f. The partial derivatives with respect to time in the time domain become jw in the frequency domain. Using this relation, we convert (2.3) to the frequency domain. Stationary media is assumed.

$$\begin{aligned}
\nabla \times \vec{E}(\vec{x}) &= -j\omega\mu_0 \mu_r^{'}(\vec{x})\vec{H}(\vec{x}) - \vec{M}^i(t,\vec{x}) - \vec{M}^c(t,\vec{x}) \\
\nabla \times \vec{H}(\vec{x}) &= j\omega\varepsilon_0 \varepsilon_r^{'}(\vec{x})\vec{E}(\vec{x}) + \vec{J}^{\,i}(t,\vec{x}) + \vec{J}^{\,c}(t,\vec{x}) \\
\nabla \cdot \vec{D}(\vec{x}) &= \rho_E(\vec{x}) \\
\nabla \cdot \vec{B}(\vec{x}) &= \rho_M(\vec{x})
\end{aligned} \tag{2.4}$$

As the first two equations of (2.4) consist of six scalar equations and six unknowns they are sufficient to formulate a solution for the unknown electric and magnetic fields in each of the three Cartesian coordinates. We ignore the divergence equations henceforth.

In addition to these equations, we require boundary conditions for interfaces with different constitutive parameters. Of interest are the conditions between dielectric media in which the tangential electric and magnetic fields are continuous, as well as Perfect Electrical Conductor (PEC) interfaces for which the tangential electric field must be zero [9].

2.2 Scattered Field Formulations

We have assumed $\vec{E}$ and $\vec{H}$ to be the total electric and magnetic fields. Now, we introduce a scattered field formulation which decomposes the fields into two additive components. This decomposition is useful as we can represent the solution in terms of known incident fields and unknown scattered fields. The following definition holds:

$$\begin{aligned} \vec{E}(\vec{x}) &= \vec{E}^{tot}(\vec{x}) \triangleq \vec{E}^{inc}(\vec{x}) + \vec{E}^{sct}(\vec{x}) \\ \vec{H}(\vec{x}) &= \vec{H}^{tot}(\vec{x}) \triangleq \vec{H}^{inc}(\vec{x}) + \vec{H}^{sct}(\vec{x}) \end{aligned} \tag{2.5}$$

where $\vec{E}^{tot}$ and $\vec{H}^{tot}$ are the total electric and magnetic fields, $\vec{E}^{inc}$ and $\vec{H}^{inc}$ are the incident electric and magnetic fields, and $\vec{E}^{sct}$ and $\vec{H}^{sct}$ are the scattered electric and magnetic fields. Complex constitutive parameters, defined in (2.6), are used to combine real permittivity with conductivity, and real permeability with magnetic loss.

$$\begin{aligned} \varepsilon(\vec{x}) &\triangleq \varepsilon_0 \varepsilon_r^{'}(\vec{x}) - j\frac{\sigma_e(\vec{x})}{\omega} \\ \mu(\vec{x}) &\triangleq \mu_0 \mu_r^{'}(\vec{x}) - j\frac{\sigma_m(\vec{x})}{\omega} \end{aligned} \tag{2.6}$$

Two sets of constitutive parameters are considered. The first, $\varepsilon(\vec{x})$ and $\mu(\vec{x})$, define the permittivity and permeability of the medium of interest, or foreground. The second set, $\varepsilon_{bkg}(\vec{x})$ and $\mu_{bkg}(\vec{x})$, characterize a reference medium as the background. We define the incident fields as the fields which satisfy:

$$\begin{aligned} j\omega\varepsilon_{bkg}(\vec{x})\vec{E}^{inc}(\vec{x}) - \nabla \times \vec{H}^{inc}(\vec{x}) + \sigma_e(\vec{x})\vec{E}^{inc}(t,\vec{x}) &= -\vec{J}^{\,i}(t,\vec{x}) \\ j\omega\mu_{bkg}(\vec{x})\vec{H}^{inc}(\vec{x}) + \nabla \times \vec{E}^{inc}(\vec{x}) + \sigma_m(\vec{x})\vec{H}^{inc}(t,\vec{x}) &= -\vec{M}^{i}(t,\vec{x}) \end{aligned} \tag{2.7}$$

where the impressed electric and magnetic sources are due to ideal dipole. This has closed form fields that can be computed analytically using the free space Green's function [5, 10].

We use (2.6) to simplify (2.7):

$$\begin{aligned} j\omega\varepsilon_{bkg}(\vec{x})\vec{E}^{inc}(\vec{x}) - \nabla \times \vec{H}^{inc}(\vec{x}) &= -\vec{J}^{\,i}(t,\vec{x}) \\ j\omega\mu_{bkg}(\vec{x})\vec{H}^{inc}(\vec{x}) + \nabla \times \vec{E}^{inc}(\vec{x}) &= -\vec{M}^{i}(t,\vec{x}) \end{aligned} \tag{2.8}$$

The scattered fields are similarily defined, and require background constitutives, foreground constitutives, and the incident fields:

$$\begin{aligned} j\omega\varepsilon(\vec{x})\vec{E}^{sct}(\vec{x}) - \nabla \times \vec{H}^{sct}(\vec{x}) &= -j\omega(\varepsilon(\vec{x}) - \varepsilon_{bkg}(\vec{x}))\vec{E}^{inc}(\vec{x}) \\ j\omega\mu(\vec{x})\vec{H}^{sct}(\vec{x}) + \nabla \times \vec{E}^{sct}(\vec{x}) &= -j\omega(\mu(\vec{x}) - \mu_{bkg}(\vec{x}))\vec{H}^{inc}(\vec{x}) \end{aligned} \tag{2.9}$$

2.3 Discretization of Maxwell's Equations

The chapter until now has focused on the continuous form of Maxwell's equations. Now, we choose to partition the domain of interest, Ω, into N_e tetrahedral elements such that $\Omega \approx \cup_{N_e} V_n$ where V_n is the n^{th} volume partition. All meshes used in this thesis are created using Gmsh meshing software [11].

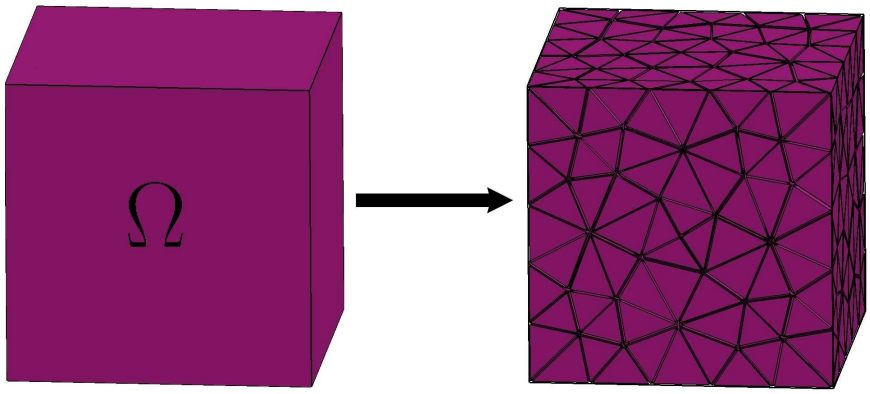


Fig. 2.1: A cuboid domain $\Omega \in \mathbb{R}^3$ which is partitioned into N_e elements.

In each tetrahedral element we use a Galerkin testing procedure with scalar testing functions $\boldsymbol{\psi}$ to formulate a system of equations that can be solved to compute the desired field values. While details of the testing functions are omited for now, it is required to formulate Maxwell's equations in a strong form flux-split equation.

To begin the formulation of the strong form flux-split equation we use (2.9) and test the variables by integrating them within each element using positionally dependent testing functions.

$$\begin{aligned}\int_{V_n} \boldsymbol{\psi} j\omega\varepsilon\vec{E}^{sct} dV_n - \int_{V_n} \boldsymbol{\psi}\nabla\times\vec{H}^{sct} dV_n &= -\int_{V_n} \boldsymbol{\psi} j\omega(\varepsilon-\varepsilon_{bkg})\vec{E}^{inc} dV_n \\ \int_{V_n} \boldsymbol{\psi} j\omega\mu\vec{H}^{sct} dV_n + \int_{V_n} \boldsymbol{\psi}\nabla\times\vec{E}^{sct} dV_n &= -\int_{V_n} \boldsymbol{\psi} j\omega(\mu-\mu_{bkg})\vec{H}^{inc} dV_n\end{aligned} \tag{2.10}$$

We use the mathematical identity in (2.11) to modify the curl terms in (2.10). Note that we have dropped the positional dependence of all variables for notational brevity.

$$\boldsymbol{\psi}\nabla\times\vec{X} = -(\nabla\boldsymbol{\psi})\times\vec{X} + \nabla\times(\boldsymbol{\psi}\vec{X}) \tag{2.11}$$

$$\begin{aligned}\int_{V_n} \boldsymbol{\psi}\nabla\times\vec{H}^{sct} dV_n &= -\int_{V_n} (\nabla\boldsymbol{\psi})\times\vec{H}^{sct} dV_n + \int_{V_n} \nabla\times(\boldsymbol{\psi}\vec{H}^{sct}) dV_n \\ \int_{V_n} \boldsymbol{\psi}\nabla\times\vec{E}^{sct} dV_n &= -\int_{V_n} (\nabla\boldsymbol{\psi})\times\vec{E}^{sct} dV_n + \int_{V_n} \nabla\times(\boldsymbol{\psi}\vec{E}^{sct}) dV_n\end{aligned} \tag{2.12}$$

We use the Kelvin-Stokes theorem [12] on the last terms in (2.12) to re-write the volume integral as the surface integral in (2.13). This will allow us to couple neighbouring elements together using the surface values.

$$\begin{aligned}\int_{V_n} \boldsymbol{\psi}\nabla\times\vec{H}^{sct} dV_n &= -\int_{V_n} (\nabla\boldsymbol{\psi})\times\vec{H}^{sct} dV_n + \oiint_{S_n} \boldsymbol{\psi}\hat{n}\times\vec{H}^{\wedge,sct}\, dS_n \\ \int_{V_n} \boldsymbol{\psi}\nabla\times\vec{E}^{sct} dV_n &= -\int_{V_n} (\nabla\boldsymbol{\psi})\times\vec{E}^{sct} dV_n + \oiint_{S_n} \boldsymbol{\psi}\hat{n}\times\vec{E}^{\wedge,sct}\, dS_n\end{aligned} \tag{2.13}$$

In this equation S_n is the surface of element V_n consisting of all its faces. $\hat{n}\times\vec{E}^{\wedge,sct}$

and $\hat{n} \times \vec{H}^{\wedge,sct}$ are the electric and magnetic fields tangential to the boundary that satisfy Maxwell's equations. Flux between elements is defined using a standard formulation of boundary flux [13]:

$$\begin{aligned}
&\hat{n} \times \vec{E}^{\wedge,sct} \triangleq \\
&\quad \frac{Y^-}{Y^- + Y^+} \hat{n} \times \vec{E}^{-,sct} + \frac{Y^+}{Y^- + Y^+} \hat{n} \times \vec{E}^{+,sct} - \frac{1}{Y^- + Y^+} \alpha \hat{n} \times \hat{n} \times (\vec{H}^{-,sct} - \vec{H}^{+,sct}) \\
&\hat{n} \times \vec{H}^{\wedge,sct} \triangleq \\
&\quad \frac{Z^-}{Z^- + Z^+} \hat{n} \times \vec{H}^{-,sct} + \frac{Z^+}{Z^- + Z^+} \hat{n} \times \vec{H}^{+,sct} + \frac{1}{Z^- + Z^+} \alpha \hat{n} \times \hat{n} \times (\vec{E}^{-,sct} - \vec{E}^{+,sct})
\end{aligned} \tag{2.14}$$

where Z is the wave impedance and Y is the wave admittiance:

$$\begin{aligned}
Z &= \sqrt{\frac{\mu}{\varepsilon}} \\
Y &= \frac{1}{Z}
\end{aligned} \tag{2.15}$$

A value of $\alpha = 1$ is used for a fully upwind scheme, the details of which are out of the scope of this thesis but can be found in [13].

While it may seem counter-intuitive to introduce additional parameters $\hat{n} \times \vec{E}^{\wedge,sct}$ and $\hat{n} \times \vec{H}^{\wedge,sct}$, they play an important role in DGM and HDGM as they are used to couple fields between elements. They do not introduce any additional unknowns, as they are formulated in terms of the existing unknown field values in element V_n and the neighbouring elements. Additionally, they are used to formulate boundary conditions which we will examine further in Chapter 4. We select

To help distinguish between the elements sharing a face, the notation *+'ve* and *-'ve* is

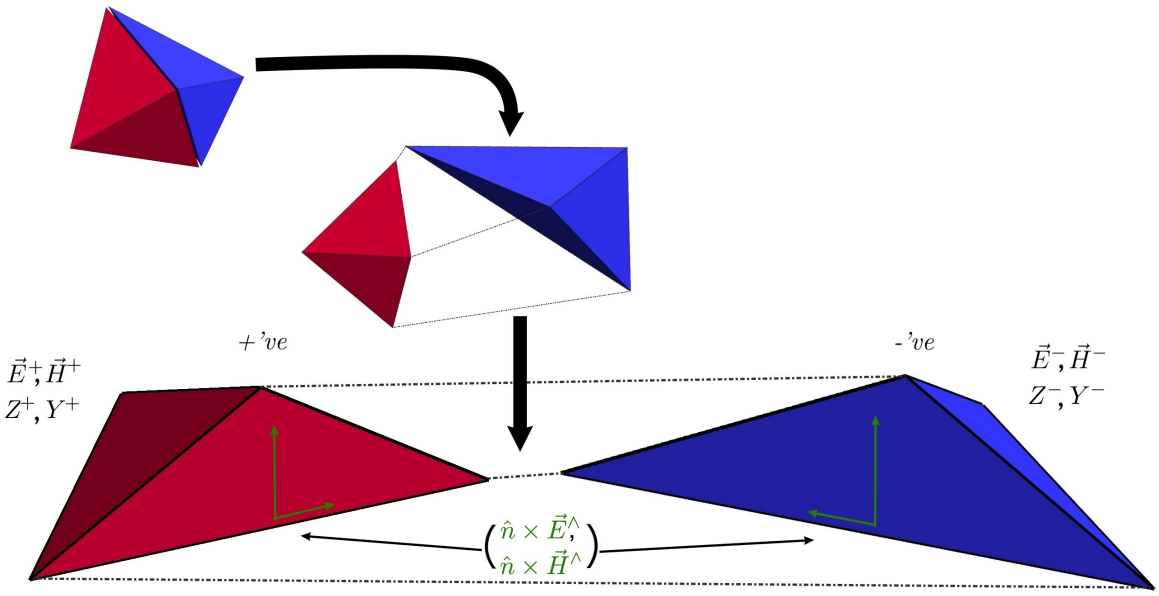


Fig. 2.2: Neighbouring elements in the mesh labeled *+'ve* and *-'ve*. Each element contains its own constitutive parameters and fields, but $\hat{n} \times \vec{E}^\wedge$ and $\hat{n} \times \vec{H}^\wedge$ are defined on the surface as a function of both elements' parameters.

introduced as shown in Figure 2.2 [13]. This notation is relative to each element in the computational domain.

By again applying (2.11) and Stokes' theorem, this time on the middle term of (2.13) and using only values within V_n, (2.13) can be re-written:

$$\begin{aligned} \int_{V_n} \boldsymbol{\psi} \nabla \times \vec{H}^{-,sct} dV_n &= \int_{V_n} \boldsymbol{\psi} \nabla \times \vec{H}^{-,sct} dV_n + \oiint_{S_n} \boldsymbol{\psi} \hat{n} \times (\vec{H}^{\wedge,sct} - \vec{H}^{-,sct}) \, dS_n \\ \int_{V_n} \boldsymbol{\psi} \nabla \times \vec{E}^{-,sct} dV_n &= \int_{V_n} \boldsymbol{\psi} \nabla \times \vec{E}^{-,sct} dV_n + \oiint_{S_n} \boldsymbol{\psi} \hat{n} \times (\vec{E}^{\wedge,sct} - \vec{E}^{-,sct}) \, dS_n \end{aligned} \tag{2.16}$$

While this seemingly undoes what we just did, it allows us to formulate the equations with a surface integral that penalizes field differences. Since the boundary values are defined as a function of both elements' parameters, it forces continuity in field components between elements. As the last step in the derivation, we replace the curl terms from (2.10) using

(2.16). We can verify that (2.17) is equivalent to (2.10) when neighbouring elements have continuous fields. Under this circumstance $\hat{n} \times \vec{E}^{-,sct}$ and $\hat{n} \times \vec{H}^{-,sct}$ approach $\hat{n} \times \vec{E}^{\wedge,sct}$ and $\hat{n} \times \vec{H}^{\wedge,sct}$, respectively, thus the surface integral disappears.

$$\begin{aligned}
&\int_{V_n} \boldsymbol{\psi} j\omega\varepsilon \vec{E}^{-,sct} dV_n - \int_{V_n} \boldsymbol{\psi} \nabla \times \vec{H}^{-,sct} dV_n - \oiint_{S_n} \boldsymbol{\psi} \hat{n} \times (\vec{H}^{\wedge,sct} - \vec{H}^{-,sct})\, dS_n \\
&\qquad = -\int_{V_n} \boldsymbol{\psi} j\omega(\varepsilon - \varepsilon_{bkg}) \vec{E}^{-,inc} dV_n \\
&\int_{V_n} \boldsymbol{\psi} j\omega\mu \vec{H}^{-,sct} dV_n + \int_{V_n} \boldsymbol{\psi} \nabla \times \vec{E}^{-,sct} dV_n + \oiint_{S_n} \boldsymbol{\psi} \hat{n} \times (\vec{E}^{\wedge,sct} - \vec{E}^{-,sct})\, dS_n \\
&\qquad = -\int_{V_n} \boldsymbol{\psi} j\omega(\mu - \mu_{bkg}) \vec{H}^{-,inc} dV_n
\end{aligned} \tag{2.17}$$

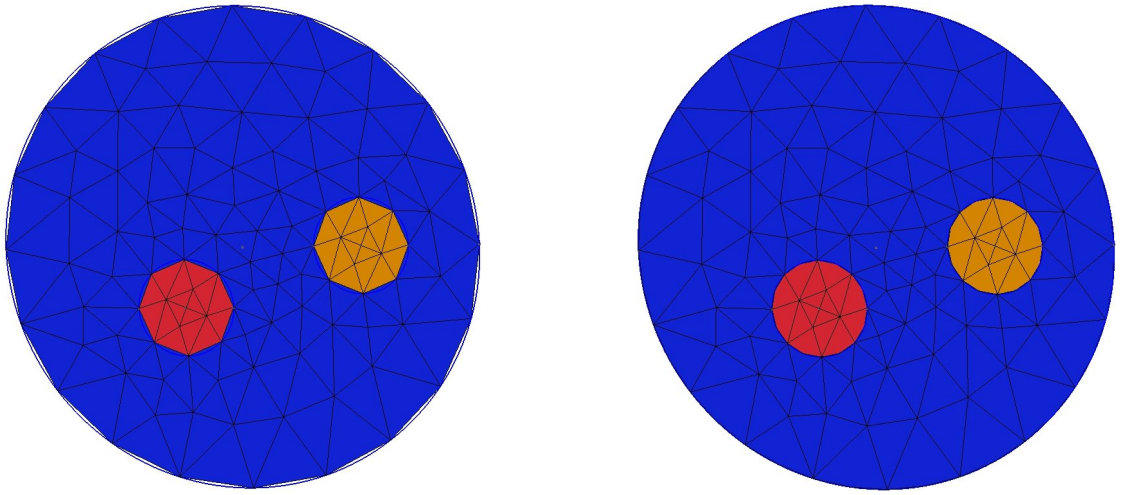
This strong form flux-split elemental system serves as the foundation for the DGM solution to Maxwell's curl equations. In the following chapter we examine how to set up discrete element and basis operators in order to numerically evaluate this equation, and in Chapter 5 we see how HDGM formulates a global system of equations based on a slight modification for improved computational performance.

3

Elements and Basis

So far we have discussed Maxwell's equations and introduced a mesh which separates the simulation domain Ω into N_e tetrahedral elements. Within each element we enforce (2.17) which includes coupling terms to relate the solution in each element to each of its neighbours. In this chapter we discuss the basis used within each element on which (2.17) is solved to enforce Maxwell's equations and obtain the approximate solution at a finite number of points. Of interest is that we can enforce a polynomial basis using DGM or HDGM which allows polynomial accuracy within each element and allows the solution to be interpolated to any point. This chapter also introduces a number of mathematical operations used by DGM and HDGM. It begins with simple, linear elements and discusses their functionality, followed by more complex high-order geometric elements.

When decomposing the mesh we have to make a choice as to the type and size of each element. As a general rule of thumb, we require 10 sampling points per wavelength in the



(a) 2D mesh using first order geometry

(b) 2D mesh using second order geometry

Fig. 3.1: Meshes using first and second order geometry. First order geometry has less accurate geometric approximations of the two circles and boundary.

medium of propagation [14]. A major benefit of DGM and HDGM is the ability to solve Maxwell's equations while maintaining high-order polynomial accuracy within each element. We can use this to our advantage with the use of larger elements utilizing high-order polynomial expansions of the fields and constitutive parameters [15]. Doing so drastically decreases the number of elements within the mesh, decreasing computational demand, while maintaining high accuracy due to the polynomial capture of Maxwell's equations [16].

One problem with large, high-order elements is the geometric error introduced by representing curved surfaces and boundaries with straight-edged elements. This geometric modelling error can seen in Figure 3.1 *a*). To allieviate this problem we introduce curvilinear elements which allow the geometry of each element to be represented using high-order polynomials. High-order elements can be seen in Figure 3.1 *b*) and are later discussed in Section 3.5.

3.1 The Barycentric Element

One problem faced when formulating the global system of equations is the large computational cost of computing an orthonormal basis over each element in Ω which is required to evaluate (2.17). To counteract this we choose to compute this in a new coordinate space, the barycentric coordinate space. In this space we define a canonical element, shown in Figure 3.2, which has equi-length edges on each axis in the range [-1, 1]. Each element, whether high-order or not, can map from the Cartesian coordinate space in which it is defined by unit vectors $\hat{x}, \hat{y}, \hat{z}$ and position $\vec{x}$ to the canonical element in the barycentric coordinate space where it is defined by unit vectors $\hat{r}, \hat{s}, \hat{t}$ and position $\vec{r}$. Each element can map to and from the canonical element allowing us to compute a single orthonormal basis for the canonical element. Since the mapping to this space is dependent on vertices of the element, each element has a unique mapping. We use the notation V_n for the element in Cartesian space and B_n for the element in barycentric coordinate space.

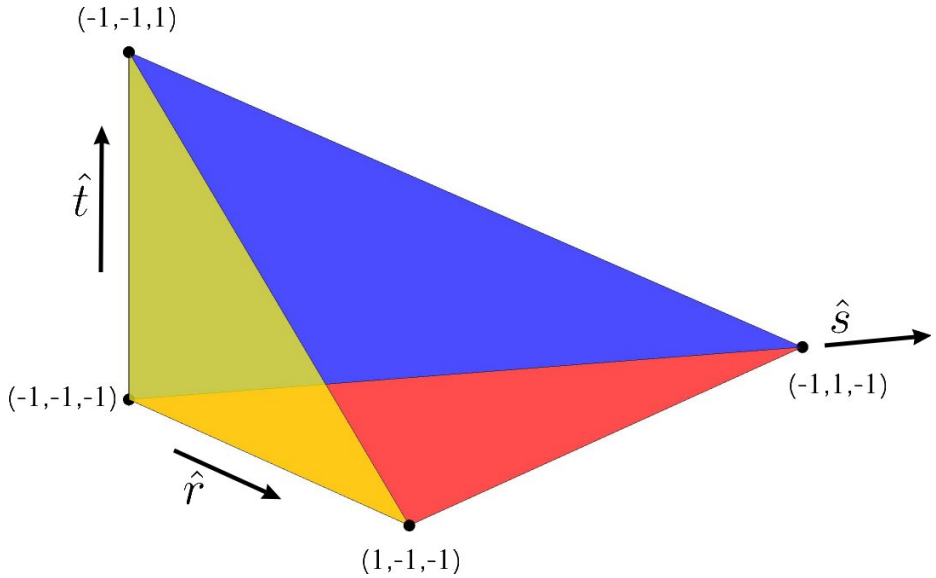


Fig. 3.2: Canonical element in the barycentric coordinate space with planes $r{=}{-}1$ in blue, $s{=}{-}1$ in yellow, and $t{=}{-}1$ in red.

3.2 First Order Tetrahedral Elements

First order tetrahedral elements are constructed with 4 vertices and 6 edges which connect the vertices. The geometry of each edge is linear as they are constructed using straight line segments from vertex to vertex as shown in Figure 3.3. While unstructured meshes using first order tetrahedral elements are better at geometric modeling than regular grids, the flat facets require more elements to accurately model curvature than curved elements. Element density near curvature can be increased to deal with this, however it comes at the expense of increased computational time.

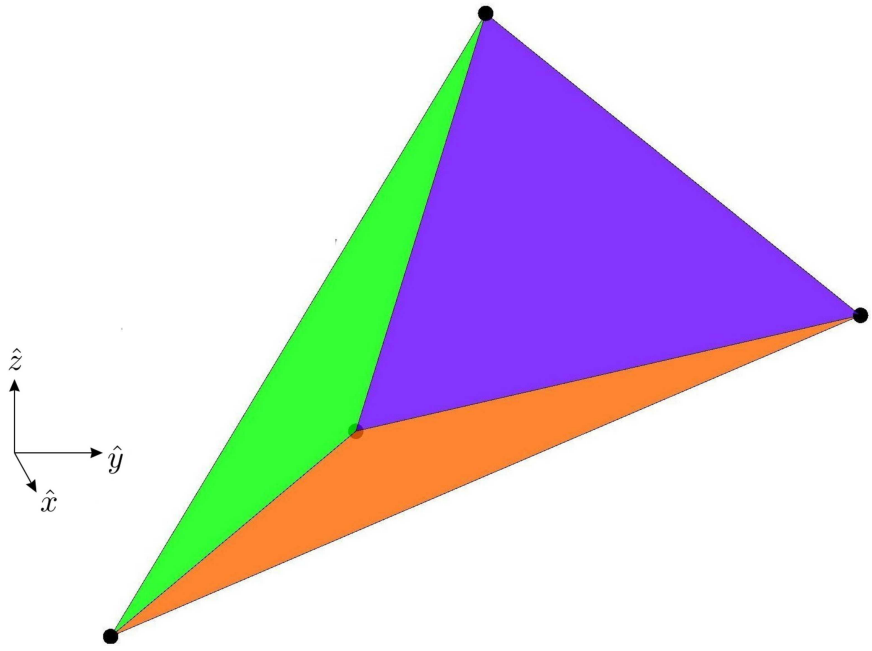


Fig. 3.3: First order element in Cartesian space. The face which corresponds to $r{=}{-}1$ is purple, $s{=}{-}1$ is green, and $t{=}{-}1$ is orange.

3.2.1 Mapping from the Canonical Element

Here we describe the transformation to map any position in the canonical element B_n to the corresponding Cartesian coordinates in V_n. The location of elements and their vertices in Cartesian space is defined by the physical geometry, which is meshed using Gmsh. From the coordinates $\vec{r}$ in B_n we use shape functions to determine the transformation to $\vec{x}$ in V_n. These shape functions linear interpolate the Cartestian coordinates using interpolating Lagrange polynomials. Since we know that r, s, and t in the canonical element must lie in the range [-1, 1] we use this information to construct our shape functions. The mapping is element dependent so each element has a unique transformation, which is written in scalar and vector form as:

$$\begin{aligned} x &= \frac{1}{2}(r+1)v_{1,x} + \frac{1}{2}(s+1)v_{2,x} + \frac{1}{2}(t+1)v_{3,x} + \frac{1}{2}(-r-s-t-1)v_{0,x} \\ y &= \frac{1}{2}(r+1)v_{1,y} + \frac{1}{2}(s+1)v_{2,y} + \frac{1}{2}(t+1)v_{3,y} + \frac{1}{2}(-r-s-t-1)v_{0,y} \\ z &= \frac{1}{2}(r+1)v_{1,z} + \frac{1}{2}(s+1)v_{2,z} + \frac{1}{2}(t+1)v_{3,z} + \frac{1}{2}(-r-s-t-1)v_{0,z} \end{aligned} \tag{3.1}$$

$$\vec{x} = \frac{1}{2}(r+1)\vec{v}_1 + \frac{1}{2}(s+1)\vec{v}_2 + \frac{1}{2}(t+1)\vec{v}_3 + \frac{1}{2}(-r-s-t-1)\vec{v}_0 \tag{3.2}$$

where $v_{i,j}$ is the j^{th} component of the i^{th} vertex. Note that if $r = s = t = -1$ then $\vec{x} = \vec{v}_0$; if $s = t = -1$ and $r = 1$ then $\vec{x} = \vec{v}_1$; if $r = t = -1$ and $s = 1$ then $\vec{x} = \vec{v}_2$; and if $r = s = -1$ and $t = 1$ then $\vec{x} = \vec{v}_3$. Any other linear combination of r, s, and t, each in the range $[-1, 1]$ results in a linear combination of the vertices and gives a $\vec{x}$ coordinate inside V_n. We must constrain this, however, as $r = s = t = 1$ is outside the canonical element which can be

confirmed by examining Figure 3.2. To prevent this we have the constraint $r + s + t \leq -1$ which defines the plane of the face which does not lie on axes $\hat{r}$, $\hat{s}$, or $\hat{t}$, and prevents $\vec{r}$ from being outside the canonical element [17].

3.2.2 The Jacobian of the Transformation

As previously mentioned we can evaluate the required values from (2.17) in the canonical element coordinate space to avoid the computational cost of computing an orthonormal basis over each element. Of interest is how to map from one coordinate space to another to change the volumetric and surface integrals present in (2.17) from $\vec{x}$ to $\vec{r}$. To do so we must introduce the Jacobian matrix:

$$\boldsymbol{\mathcal{J}} = \begin{bmatrix} \frac{\partial x}{\partial r} & \frac{\partial x}{\partial s} & \frac{\partial x}{\partial t} \\ \frac{\partial y}{\partial r} & \frac{\partial y}{\partial s} & \frac{\partial y}{\partial t} \\ \frac{\partial z}{\partial r} & \frac{\partial z}{\partial s} & \frac{\partial z}{\partial t} \end{bmatrix} \tag{3.3}$$

This matrix defines the rate of change of each Cartesian component with respect to each coordinate in barycentric space. Using the shape functions from (3.1) we can compute the Jacobian matrix by analytically computing the derivatives:

$$\boldsymbol{\mathcal{J}} = \begin{bmatrix} \frac{1}{2}v_{1,x} - \frac{1}{2}v_{0,x} & \frac{1}{2}v_{2,x} - \frac{1}{2}v_{0,x} & \frac{1}{2}v_{3,x} - \frac{1}{2}v_{0,x} \\ \frac{1}{2}v_{1,y} - \frac{1}{2}v_{0,y} & \frac{1}{2}v_{2,y} - \frac{1}{2}v_{0,y} & \frac{1}{2}v_{3,y} - \frac{1}{2}v_{0,y} \\ \frac{1}{2}v_{1,z} - \frac{1}{2}v_{0,z} & \frac{1}{2}v_{2,z} - \frac{1}{2}v_{0,z} & \frac{1}{2}v_{3,z} - \frac{1}{2}v_{0,z} \end{bmatrix} \tag{3.4}$$

We see from these equation that in any first order element the Jacobian of the transformation is constant in that element as it has no positional dependence. This is because we use first order shape functions with a single derivative. We will see how this changes for high-order geometries later in this chapter.

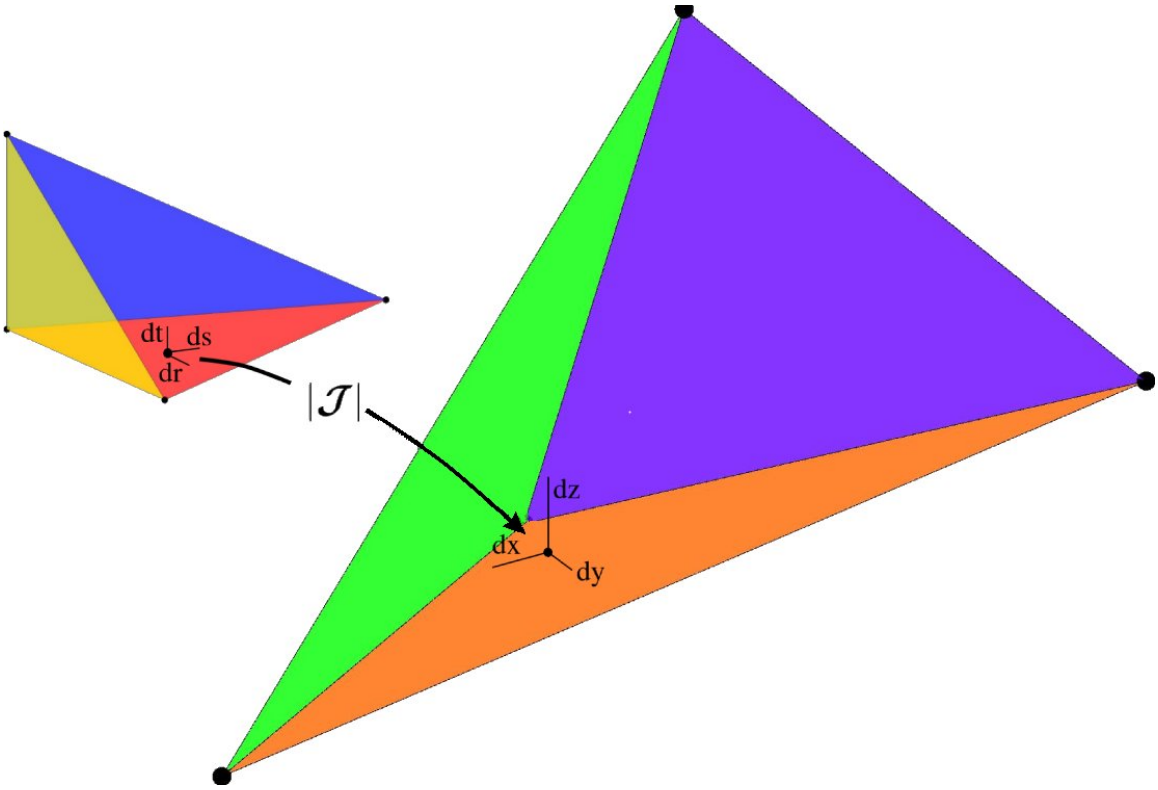


Fig. 3.4: The determinant of the Jacobian is the change in volume between the canonical element and the Cartesian element.

The determinant of the Jacobian matrix, $|\mathcal{J}|$, hereby refered to as the Jacobian, gives the relationship between an infinitesimal volume at a point in Cartesian space and barycentric space which is shown in Figure 3.4 and proven in Appendix A. The determinant is required to change integration variables between the two spaces, as it allows us to account for changes in volume between them. This is shown in (3.5) and (3.6) where again we note that since the Jacobian is constant within a first order element, it is positionally independent and can be separated from the integral.

$$V = \int_z \int_y \int_x dxdydz = \int_t \int_s \int_r |\mathcal{J}| drdsdt = |\mathcal{J}| \int_t \int_s \int_r drdsdt \tag{3.5}$$

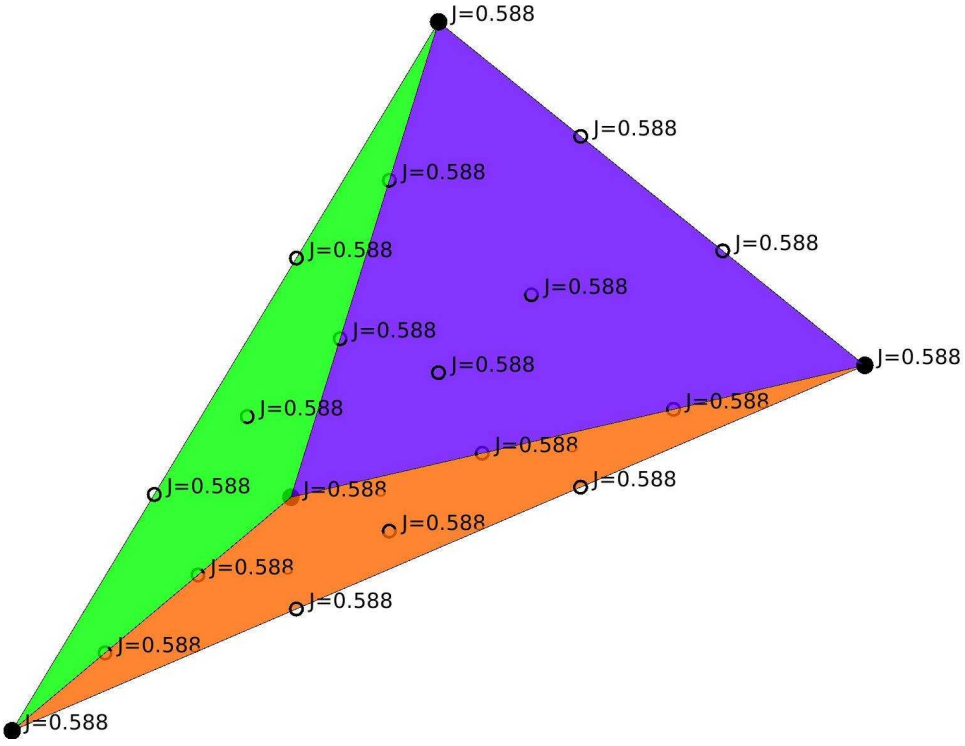


Fig. 3.5: The Jacobian computed at a number of points within a linear tetrahedral. Linear elements have a uniform Jacobian. A Jacobain less than one implies the canonical element has a larger volume.

$$\int_{V_n} f(\vec{x})d\vec{x} = |\boldsymbol{\mathcal{J}}| \int_{B_n} f(\vec{r})d\vec{r} \tag{3.6}$$

3.2.3 Mapping to the Canonical Element

Now that we know how to account for change in volume we are ready to introduce the mapping from Cartesian to barycentric space. This plays an important role in DGM and HDGM as computing the values of the fields at a given coordinate requires determining which element the coordinate lies in and accurately mapping the coordinate from $\vec{x}$ in V_n to $\vec{r}$ in B_n. Luckily, for linear elements the equations simplify. As we will see in Section

3.5.3 it becomes more complex when using high-order elements. We begin the derivation with (3.2) and re-write the equation to isolate r, s, and t:

$$\begin{aligned}
\vec{x} &= \frac{1}{2}(r+1)\vec{v}_1 + \frac{1}{2}(s+1)\vec{v}_2 + \frac{1}{2}(t+1)\vec{v}_3 + \frac{1}{2}(-r-s-t-1)\vec{v}_0 \\
\Rightarrow 2\vec{x} &= (r+1)\vec{v}_1 + (s+1)\vec{v}_2 + (t+1)\vec{v}_3 + (-r-s-t-1)\vec{v}_0 \\
\Rightarrow 2\vec{x} &= r(\vec{v}_1 - \vec{v}_0) + s(\vec{v}_2 - \vec{v}_0) + t(\vec{v}_3 - \vec{v}_0) + \vec{v}_1 + \vec{v}_2 + \vec{v}_3 - \vec{v}_0 \\
\Rightarrow 2\vec{x} - \vec{v}_1 - \vec{v}_2 - \vec{v}_3 + \vec{v}_0 &= r(\vec{v}_1 - \vec{v}_0) + s(\vec{v}_2 - \vec{v}_0) + t(\vec{v}_3 - \vec{v}_0)
\end{aligned} \tag{3.7}$$

We then convert the scalar equations to matrix form:

$$\begin{aligned}
\begin{bmatrix} 2x - v_{1,x} - v_{2,x} - v_{3,x} + v_{0,x} \\ 2y - v_{1,y} - v_{2,y} - v_{3,y} + v_{0,y} \\ 2z - v_{1,z} - v_{2,z} - v_{3,z} + v_{0,z} \end{bmatrix} &= r \begin{bmatrix} v_{1,x} - v_{0,x} \\ v_{1,y} - v_{0,y} \\ v_{1,z} - v_{0,z} \end{bmatrix} + s \begin{bmatrix} v_{2,x} - v_{0,x} \\ v_{2,y} - v_{0,y} \\ v_{2,z} - v_{0,z} \end{bmatrix} + t \begin{bmatrix} v_{3,x} - v_{0,x} \\ v_{3,y} - v_{0,y} \\ v_{3,z} - v_{0,z} \end{bmatrix} \\
\begin{bmatrix} 2x - v_{1,x} - v_{2,x} - v_{3,x} + v_{0,x} \\ 2y - v_{1,y} - v_{2,y} - v_{3,y} + v_{0,y} \\ 2z - v_{1,z} - v_{2,z} - v_{3,z} + v_{0,z} \end{bmatrix} &= \begin{bmatrix} v_{1,x} - v_{0,x} & v_{2,x} - v_{0,x} & v_{3,x} - v_{0,x} \\ v_{1,y} - v_{0,y} & v_{2,y} - v_{0,y} & v_{3,y} - v_{0,y} \\ v_{1,z} - v_{0,z} & v_{2,z} - v_{0,z} & v_{3,z} - v_{0,z} \end{bmatrix} \begin{bmatrix} r \\ s \\ t \end{bmatrix}
\end{aligned} \tag{3.8}$$

Note that the matrix term on the right hand side of (3.8) is identical to $2\boldsymbol{\mathcal{J}}$. We invert this to obtain $\vec{r}$ as a function of $\vec{x}$ which allows us to determine $\vec{r}$ in B_n from $\vec{x}$ in V_n:

$$\vec{r} = \frac{1}{2}\boldsymbol{\mathcal{J}}^{-1}(2\vec{x} - \vec{v_1} - \vec{v_2} - \vec{v_3} + \vec{v_4}) \tag{3.9}$$

3.3 The Nodal Basis

Before we can introduce important operators for linear elements, namely the mass and stiffness matrices, we must have an understanding of the basis. This section uses the previously discussed mappings between coordinate systems in order to evaluate the basis. HDGM requires an additional basis set which is described in Chapter 5.

In order to numerically compute a solution to (2.17) we create a set of discrete points within each element on which we test the solution in order to formulate a system of equations. One of the benefits of DGM and HDGM is the ability to represent unknowns using polynomial functions. This allows polynomial accurate constitutives and fields, to the selected order p. Since p can be selected independently for each element the solution accuracy can be locally refined. Common p^{th} order bases include orthogonal Legendre polynomials, known as a modal basis, and orthogonal Lagrange interpolating polynomials, known as a nodal basis which is used herein. These basis polynomials are created such that they contain all polynomial combinations of r, s, and t complete to order p. The number of independent functions required to represent the polynomial in 3D space can be computed using

$$N_p = \frac{(p+1)(p+2)(p+3)}{6} \tag{3.10}$$

where N_p is the number of terms required to represent a p^{th} order polynomial [13].

The basis points are created using equidistant points in the canonical element and mapping each point to Cartesian space using the shape functions for the element. Figure 3.6 shows the basis points for first, second, and third order polynomials in both barycentric and Cartesian coordinates.

Lagrange polynomials $\ell_i(\vec{r})$ are created for each $i \in [1, N_p]$ basis functions. These polynomi-

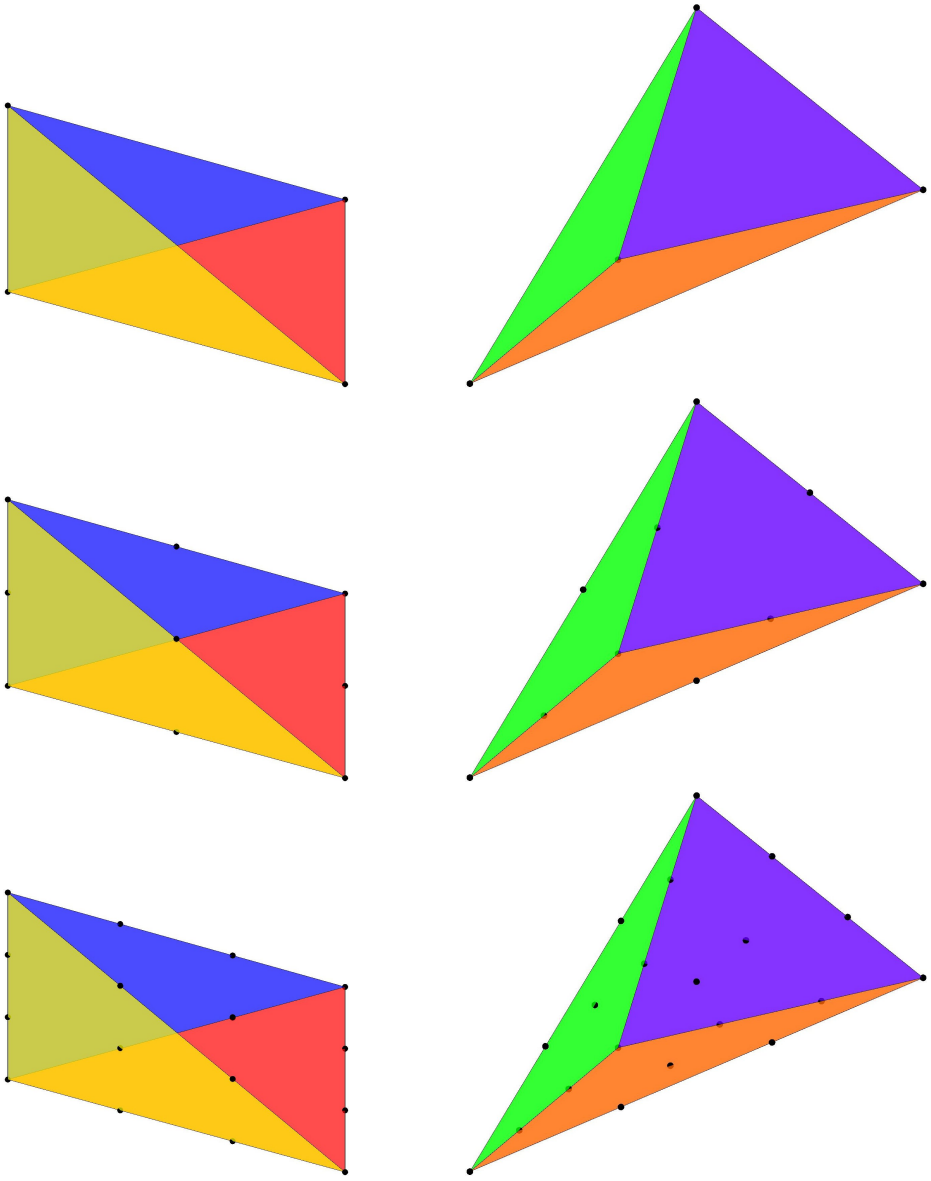


Fig. 3.6: Basis points, marked with a black dot, for the canonical element and corresponding basis nodes in the Cartesian element for $p = 1, 2,$ and 3.

als can easily be computed as a function of polynomial order desired. They can be evaluated at any position $\vec{r}$ in the barycentric element and have the Kronecker delta property δ_{ij} for polynomial i evaluated as basis point j [13].

Visualization of 3D Lagrange polynomials is difficult in 3D as it requires 4 dimensions; r, s, t, and value. Instead we simplify this by showing the set of 2D Lagrange polynomials for the canonical triangle which is easily visualized as a function or r, s, and value. Figure 3.7 shows the Lagrange polynomials for 1^{st}, 2^{nd}, and 3^{rd} order triangles. These plots are similar to the $4th$ order polynomials shown in Figure 6.7 of [13]. We can verify the Kronecker delta property as at each basis point location, shown in black, all polynomials are 0 except for a single polynomial which has a value of 1. The Lagrange polynomials for orders 4 through 6 are shown in Appendix B.

If we wish to evaluate the basis at any point $\vec{r} \in B_n$ we can use the sum of the basis polynomials evaluated at $\vec{r}$. This allows representation of functions as p^{th} order polynomials. An example approximation of the x component of the electric field using Lagrange polynomials is shown in (3.11) where $\ell_i(\vec{r})$ is the value of the i^{th} Lagrange polynomial evaluated at $\vec{r}$, and $E_{i,x}$ is the x component of the electric field at basis point i.

$$E_x(\vec{r}) = \sum_{i=1}^{Np} \ell_i(\vec{r}) E_{i,x} \tag{3.11}$$

We require the ability to evaluate basis polynomials at any set of points within a given element. Interpolation matrices, $\boldsymbol{\mathcal{P}}(\vec{r}^{\,'})$, are constructed to compute the fields at each point $\vec{r}^{\,'}$ of M points as a function of the field values at the elements' basis points:

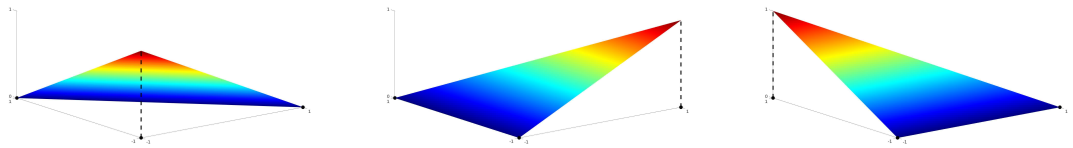
$$\boldsymbol{\mathcal{P}}(\vec{r}^{\,'}) = \begin{bmatrix} \ell_1(\vec{r}_1^{\,'}) & \ell_2(\vec{r}_1^{\,'}) & \ell_3(\vec{r}_1^{\,'}) & \dots & \ell_{Np}(\vec{r}_1^{\,'}) \\ \ell_1(\vec{r}_2^{\,'}) & \ell_2(\vec{r}_2^{\,'}) & \ell_3(\vec{r}_2^{\,'}) & \dots & \ell_{Np}(\vec{r}_2^{\,'}) \\ \vdots & & & \ddots & \vdots \\ \ell_1(\vec{r}_M^{\,'}) & \ell_2(\vec{r}_M^{\,'}) & \ell_3(\vec{r}_M^{\,'}) & \dots & \ell_{Np}(\vec{r}_M^{\,'}) \end{bmatrix} \tag{3.12}$$

The computation of this operator can be simplified using modal orthogonal polynomials on the canonical element, the details of which can be found in [13]. The application of this interpolation matrix is shown below, where $\underline{E}_x$ is the x component of the electric field at all basis points:

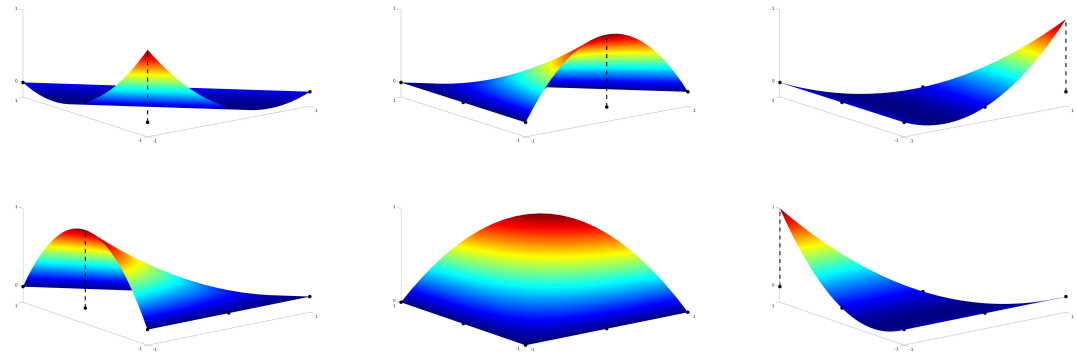
$$E_x(\vec{r}^{\,'}) = \begin{bmatrix} E_x(\vec{r}_1^{\,'}) \\ E_x(\vec{r}_2^{\,'}) \\ \vdots \\ E_x(\vec{r}_M^{\,'}) \end{bmatrix} = \boldsymbol{\mathcal{P}}(\vec{r}^{\,'}) \underline{E}_x \tag{3.13}$$

The ability to evaluate fields anywhere within an element is necessary, as fields within the element vary, and we require the values precisely at antenna locations. It is also needed

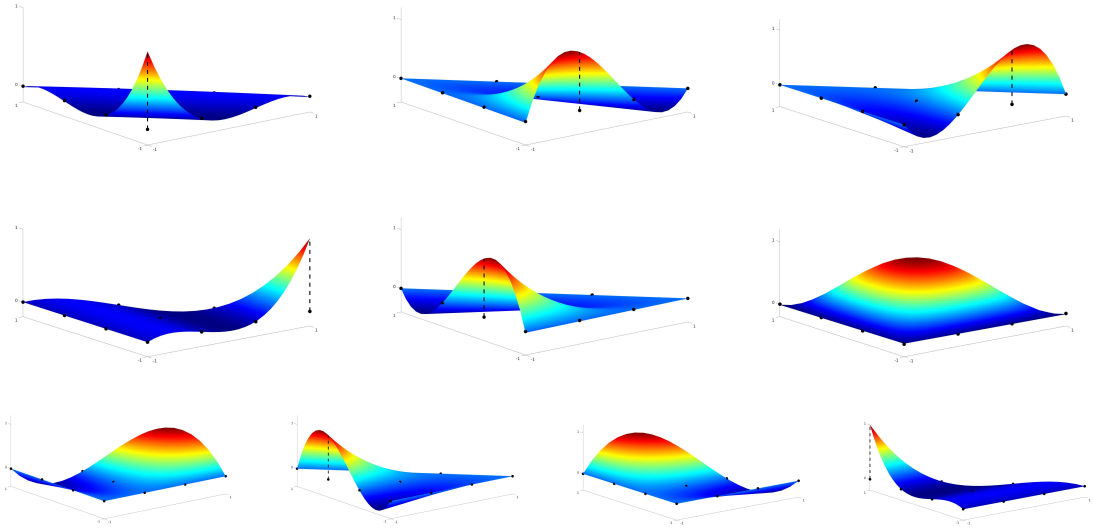
when evaluating DGM matrix terms when the basis order changes between elements. Figure 3.8 shows plots of example polynomial approximations, shown in 2D for simplification. We see that as the order of the polynomial increases so does the accuracy of the representation of the function.



(a) p=1

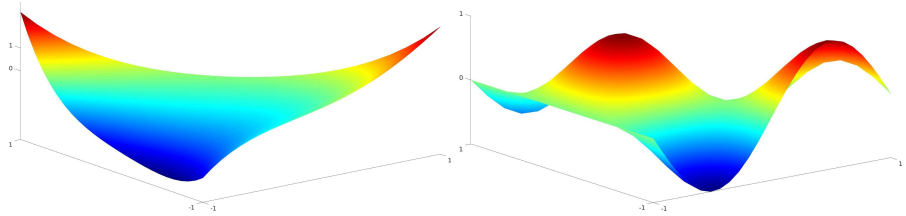


(b) p=2

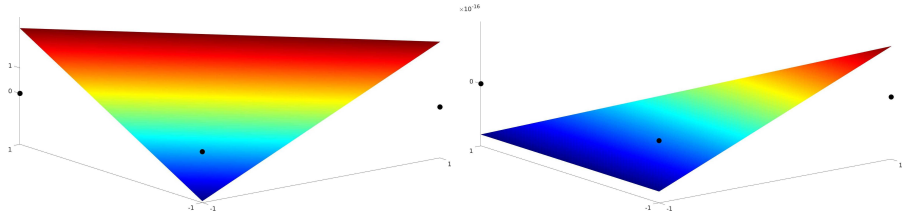


(c) p=3

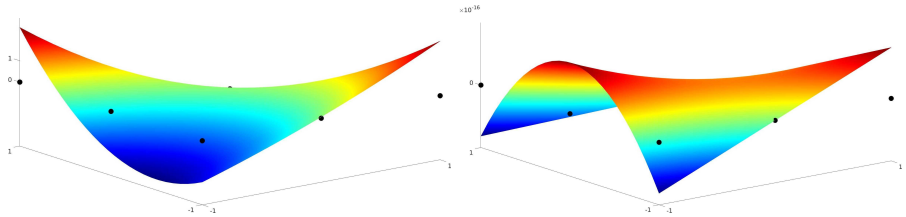
Fig. 3.7: 2D Lagrange basis polynomials for orders $p = 1, 2$, and 3. These polynomials are similar to those presented in Figure 6.7 in [13]



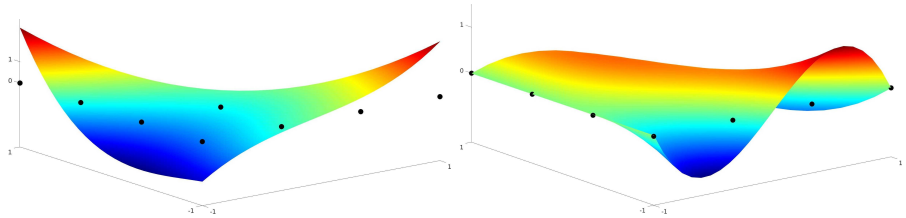
(a) True function values



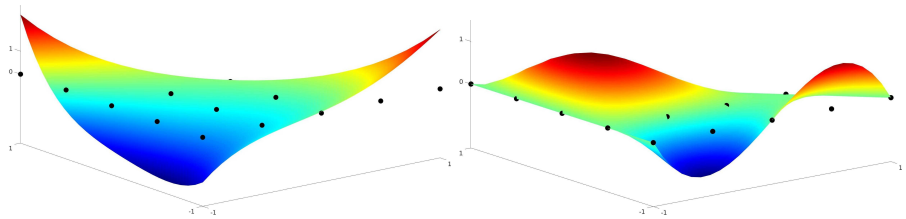
(b) $p = 1$



(c) $p = 2$



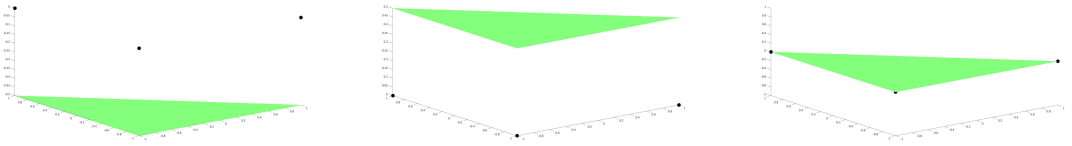
(d) $p = 3$



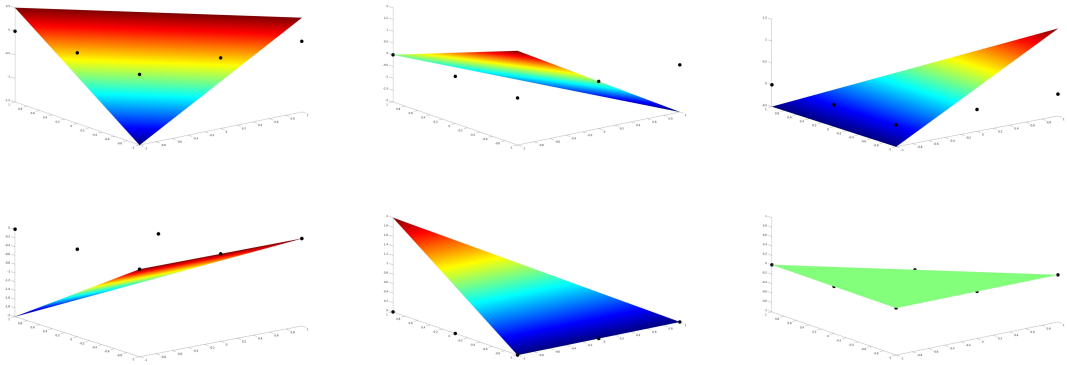
(e) $p = 4$

Fig. 3.8: Polynomials representation of a polynomial function (left) and sinusoidal function (right) as order increases.

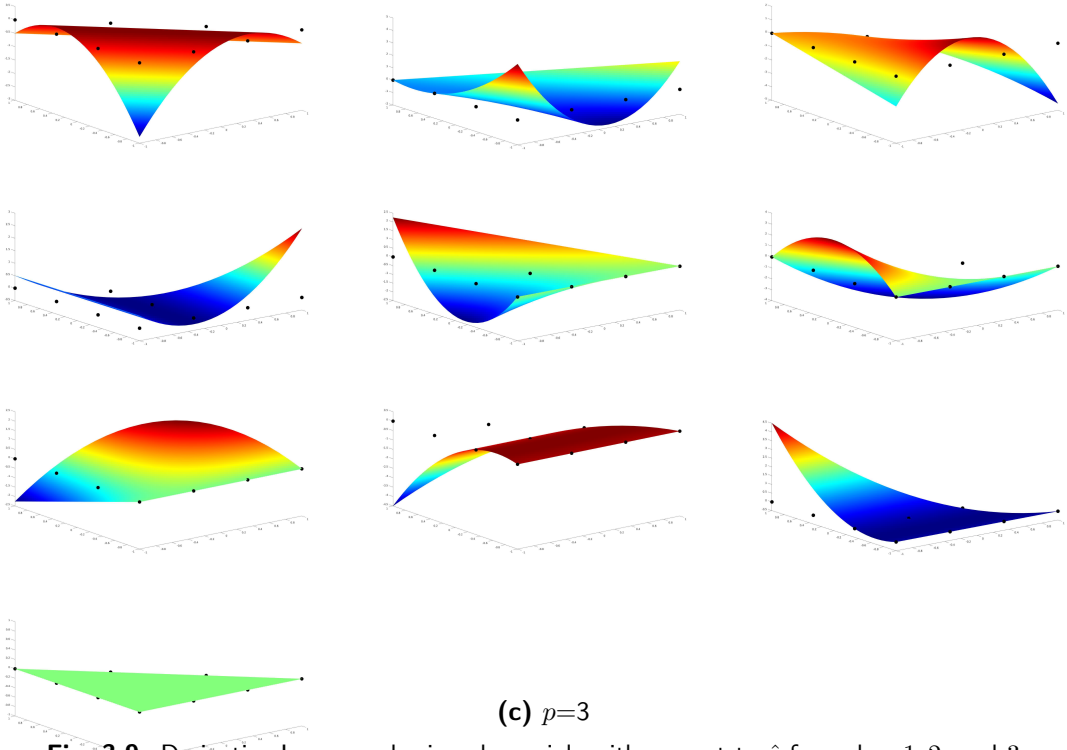
Important to the formulation of PDE's is the ability to compute the derivative of the polynomial functions using the basis. This is discussed further in Section 3.4, but here we show the derivative polynomials which can be computed from the Lagrange polynomials. Figure 3.9 and Figure 3.10 show the derivative polynomials with respect to $\hat{r}$ and $\hat{s}$ respectively, each with 1^{st}, 2^{nd}, and 3^{rd} order triangles. We can verify the order of each derivative polynomial is $p-1$. The derivative polynomials for orders 4 through 6 are shown in Appendix B.



(a) p=1

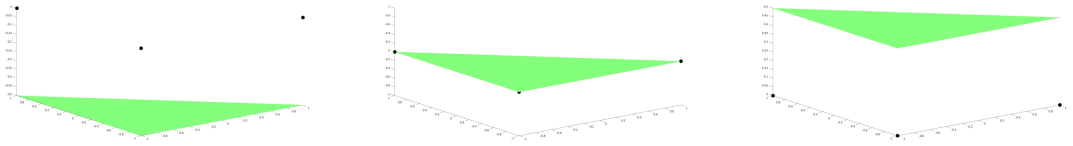


(b) p=2

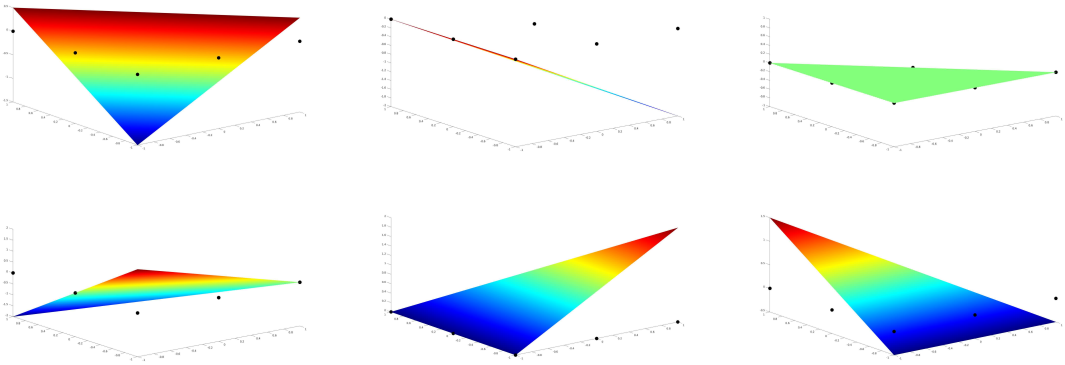


(c) p=3

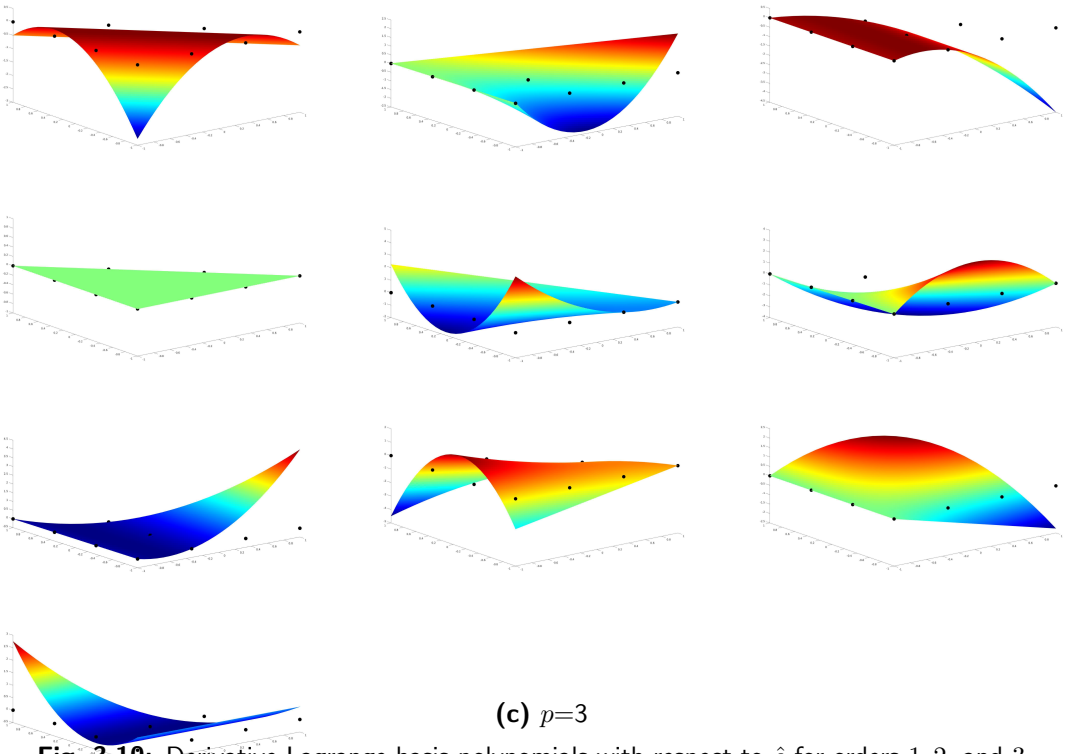
Fig. 3.9: Derivative Lagrange basis polynomials with respect to $\hat{r}$ for orders $1, 2$, and 3.



(a) p=1



(b) p=2



(c) p=3

Fig. 3.10: Derivative Lagrange basis polynomials with respect to $\hat{s}$ for orders $1, 2$, and 3.

3.4 First Order Tetrahedral Mass and Stiffness Matrices

Here we introduce the mass matrix which is used to compute the inner product of any two functions within a linear element using our basis. Below, the desired functionality from this operator is shown as a continuous operator in both Cartesian and barycentric space:

$$< f(\vec{x}), g(\vec{x}) >_{V_n} = \int_{V_n} f(\vec{x}) g(\vec{x})^* d\vec{x} = |\boldsymbol{\mathcal{J}}| \int_{B_n} f(\vec{r}) g(\vec{r})^* d\vec{r} \tag{3.14}$$

where $(\cdot)^*$ denotes complex conjugation. The discrete matrix, $\boldsymbol{\mathcal{M}}$, approximates this by computing the inner product between basis polynomials. The calculation of this operator also benefits from modal expansions on the canonical elements [13].

$$\boldsymbol{\mathcal{M}}_{V_n} = |\boldsymbol{\mathcal{J}}| \int_{B_n} f(\vec{r}) g(\vec{r})^* d\vec{r} = |\boldsymbol{\mathcal{J}}| \begin{bmatrix} < \ell_1, \ell_1 > & < \ell_1, \ell_2 > & \dots & < \ell_1, \ell_{Np} > \\ < \ell_2, \ell_1 > & < \ell_2, \ell_2 > & \dots & < \ell_2, \ell_{Np} > \\ \vdots & \vdots & \ddots & \vdots \\ < \ell_{Np}, \ell_1 > & < \ell_{Np}, \ell_2 > & \dots & < \ell_{Np}, \ell_{Np} > \end{bmatrix} \quad \ell_i \in V_n \tag{3.15}$$

We give an example of integration of the x components of the electric field times magnetic field below where $\underline{E}_x$ and $\underline{H}_x$ are the x components at each of the nodal points in V_n, and $(\cdot)^H$ denotes the complex conjugate transposition.

$$\int_{V_n} E_x(\vec{x}) H_x(\vec{x}) d\vec{x} \approx \underline{H}_x^H \boldsymbol{\mathcal{M}}_{V_n} \underline{E}_x \tag{3.16}$$

We require another operator to evaluate the volumetric integral of a function with the derivative of another function. This operator is called the stiffness matrix and the functionality we desire is:

$$< f(\vec{x}), \frac{\partial g(\vec{x})}{\partial \zeta} >_{V_n} = \int_{V_n} f(\vec{x}) \frac{\partial g(\vec{x})^*}{\partial \zeta} \, d\vec{x} \qquad \zeta \in \{x, y, z\} \tag{3.17}$$

We begin the derivation with (3.18) which shows the computation of the derivative of a polynomial function $f(\vec{r})$ with respect to barycentric coordinates r, s, and t.

$$\begin{aligned} \frac{\partial f(\vec{r})}{\partial r} &= \sum_{i=1}^{Np} \frac{\partial \ell_i(\vec{r})}{\partial r} f(\vec{r}_i) \\ \frac{\partial f(\vec{r})}{\partial s} &= \sum_{i=1}^{Np} \frac{\partial \ell_i(\vec{r})}{\partial s} f(\vec{r}_i) \\ \frac{\partial f(\vec{r})}{\partial t} &= \sum_{i=1}^{Np} \frac{\partial \ell_i(\vec{r})}{\partial t} f(\vec{r}_i) \end{aligned} \tag{3.18}$$

Using this we construct the discrete matrix $\boldsymbol{\mathcal{D}}_r$ in (3.19) which consists of terms that are the derivatives of the Lagrange basis polynomials.

$$\boldsymbol{\mathcal{D}}_r = \begin{bmatrix} \partial_r \ell_1(\vec{r}_1) & \partial_r \ell_2(\vec{r}_1) & \dots & \partial_r \ell_{Np}(\vec{r}_1) \\ \partial_r \ell_1(\vec{r}_2) & \partial_r \ell_2(\vec{r}_2) & \dots & \partial_r \ell_{Np}(\vec{r}_2) \\ \vdots & \vdots & \ddots & \vdots \\ \partial_r \ell_1(\vec{r}_{N_p}) & \partial_r \ell_2(\vec{r}_{N_p}) & \dots & \partial_r \ell_{Np}(\vec{r}_{N_p}) \end{bmatrix} \tag{3.19}$$

Equation (3.20) shows an example of how this operator can be used to take derivatives of a field component. Again while this function computes the derivative with respect to r we can compute matrices $\boldsymbol{\mathcal{D}}_s$ and $\boldsymbol{\mathcal{D}}_t$ which take derivatives with respect to s and t in a similar manner.

$$\begin{aligned} E(\vec{r}) &\approx \left[\ell_1(\vec{r}) \dots \ell_{Np}(\vec{r})\right] \underline{E}_x \\ \frac{\partial E(\vec{r})}{\partial \vec{r}} &\approx \boldsymbol{\mathcal{D}}_r \underline{E}_x \end{aligned} \tag{3.20}$$

Ultimately we require taking derivatives with respect to Cartesian coordinates. We do this by using the chain rule shown in (3.21). Note that the partial derivatives of the Lagrange basis functions are trivial to compute as it consists of only polynomial terms. Again we only show construction for a single scalar component as other components are computed similarily.

$$\begin{aligned} \frac{\partial \ell}{\partial x} &= \frac{\partial r}{\partial x}\frac{\partial \ell}{\partial r} + \frac{\partial s}{\partial x}\frac{\partial \ell}{\partial s} + \frac{\partial t}{\partial x}\frac{\partial \ell}{\partial t} \\ \boldsymbol{\mathcal{D}}_x &= \frac{\partial r}{\partial x}\boldsymbol{\mathcal{D}}_r + \frac{\partial s}{\partial x}\boldsymbol{\mathcal{D}}_s + \frac{\partial t}{\partial x}\boldsymbol{\mathcal{D}}_t \end{aligned} \tag{3.21}$$

Now that we can compute $\boldsymbol{\mathcal{D}}_x$, $\boldsymbol{\mathcal{D}}_y$, and $\boldsymbol{\mathcal{D}}_z$ we are ready to construct the stiffness matrices. The required integrals for $\boldsymbol{\mathcal{S}}_x$ are shown in (3.22) and construction of the discrete operator is given in (3.23). For this discrete operator we use the required $\boldsymbol{\mathcal{D}}$ matrix to compute derivatives using the chain rule.

$$< f(\vec{x}), \frac{\partial g(\vec{x})}{\partial x} >_{V_n} = \int_{V_n} f(\vec{x}) \frac{\partial g(\vec{x})^*}{\partial x} d\vec{x} = |\boldsymbol{\mathcal{J}}| \int_{B_n} f(\vec{r}) \frac{\partial g(\vec{r})^*}{\partial \vec{r}} \frac{\partial \vec{r}}{\partial x} d\vec{r} \tag{3.22}$$

$$\boldsymbol{\mathcal{S}}_x = |\boldsymbol{\mathcal{J}}| \int_{B_n} \ell(\vec{r}) \frac{\partial \ell^T(\vec{r})}{\partial x} d\vec{r} = |\boldsymbol{\mathcal{J}}| \begin{bmatrix} < \frac{\partial \ell_1}{\partial x}, \ell_1 > < \frac{\partial \ell_1}{\partial x}, \ell_2 > \ \dots \ < \frac{\partial \ell_1}{\partial x}, \ell_{Np} > \\ < \frac{\partial \ell_2}{\partial x}, \ell_1 > < \frac{\partial \ell_2}{\partial x}, \ell_2 > \ \dots \ < \frac{\partial \ell_2}{\partial x}, \ell_{Np} > \\ \vdots \quad \vdots \quad \ddots \quad \vdots \\ < \frac{\partial \ell_{Np}}{\partial x}, \ell_1 > < \frac{\partial \ell_{Np}}{\partial x}, \ell_2 > \ \dots \ < \frac{\partial \ell_{Np}}{\partial x}, \ell_{Np} > \end{bmatrix} \tag{3.23}$$

$$= \boldsymbol{\mathcal{M}} \boldsymbol{\mathcal{D}}_x$$

We give an example of integration of the x component of the electric field times the derivative of the x component of the magnetic field with respect to z in (3.24) where again $\underline{E}_x$ and $\underline{H}_x$ are the x components at each of the nodal points in V_n.

$$\int_{V_n} E_x(\vec{x}) \frac{\partial H_x(\vec{x})}{\partial z} d\vec{x} \approx \underline{H}_x^H \boldsymbol{\mathcal{S}}_z \, \underline{E}_x \tag{3.24}$$

3.5 High-Order Tetrahedral Elements

In this section we examine the formulation of the basis and required operators for high-order elements. While only 2^{nd} order geometries were implemented, the theory introduced in this section extends a general polynomial geometric expansion of any order. Figure 3.11 shows a second order tetrahedral element in Cartesian coordinates. We see that it is similar to a linear element in that it is constructed with 4 vertices and 6 edges. However, unlike linear elements, second order elements contain 6 additional DOF which lie on the mid point of each edge. These additional nodes capture the quadratic nature of second order geometries, and allow curvature in Cartesian coordinates. High-order elements use the same linear canonical element, but use the Jacobian of the transformation to capture the curvature in Cartesian space.

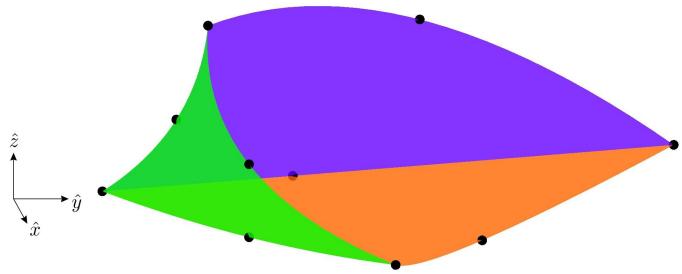


Fig. 3.11: A second order element in Cartesian coordinates.

In this section it is important to distinguish the difference between the geometric expansion order and solution expansion order. Similar to how we used high-order expansions for the fields in a linear element in Section 3.3, we can use a first order expansion of fields in a high-order geometric element. That is, we allow independent selection of the solution order p and geometric order g.

3.5.1 Mapping from the Canonical Element

Similar to linear tetrahedra, we require a mapping which takes points in barycentric coordinates and computes the corresponding Cartesian coordinates. Again shape functions are used, however more shape functions are required when using high-order elements as the relationship between $\vec{x}$ and $\vec{r}$ is a second order function. While linear elements required 4 shape functions, second order elements require 10 shape functions to represent the geometry, which are shown in (3.25) and can be found in literature [18]. Note the following similarities to the linear element: if $(r, s, t) = (-1, -1, -1)$ then $\vec{x} = \vec{v}_0$; if $(r, s, t) = (1, -1, -1)$ then $\vec{x} = \vec{v}_1$; if $(r, s, t) = (-1, 1, -1)$ then $\vec{x} = \vec{v}_2$; and if $(r, s, t) = (-1, -1, 1)$ then $\vec{x} = \vec{v}_3$. Basis points in the middle of each edge give us 6 additional equalities; for example if $(r, s, t) = (0, -1, -1)$ then $\vec{x} = \vec{v}_4$. Each of these points also satisfy that a single shape function is 1 at the basis point and 0 at all others.

$$
\begin{aligned}
\vec{x} &= \sum_{i=0}^{9} N_i(\vec{r})\ \vec{v}_i \\
N_0 &= \frac{1}{2}(-2-r-s-t)((-2-r-s-t)+1) \\
N_1 &= \frac{1}{2}r(r+1) \\
N_2 &= \frac{1}{2}s(s+1) \\
N_3 &= \frac{1}{2}t(t+1) \\
N_4 &= (r+1)((-2-r-s-t)+1) \\
N_5 &= (r+1)(s+1) \\
N_6 &= (s+1)((-2-r-s-t)+1) \\
N_7 &= (t+1)((-2-r-s-t)+1) \\
N_8 &= (s+1)(t+1) \\
N_9 &= (t+1)(r+1)
\end{aligned}
\tag{3.25}
$$

3.5.2 The Jacobian of the Transformation

Once again we require a mapping from one coordinate space to another. While this was simple for the linear element we show herein that high-order geometries result in a more complex formulation as the Jacobian becomes position dependent. Using the shape functions for the selected geometric order we can analytically compute the partial derivatives

required. Below we give the combined position function for a 2^{nd} order element:

$$\begin{aligned}\vec{x} = &\frac{1}{2}(-2-r-s-t)((-2-r-s-t)+1)\vec{v}_0 + \frac{1}{2}r(r+1)\vec{v}_1 \\ &+ \frac{1}{2}s(s+1)\vec{v}_2 + \frac{1}{2}t(t+1)\vec{v}_3 + (r+1)((-2-r-s-t)+1)\vec{v}_4 \\ &+ (r+1)(s+1)\vec{v}_5 + (s+1)((-2-r-s-t)+1)\vec{v}_6 + (t+1)((-2-r-s-t)+1)\vec{v}_7 \\ &+ (s+1)(t+1)\vec{v}_8 + (t+1)(r+1)\vec{v}_9\end{aligned} \tag{3.26}$$

Next, similar polynomial terms are grouped:

$$\begin{aligned}\vec{x} = &r^2\left(\frac{\vec{v}_0+\vec{v}_1}{2} - \vec{v}_4\right) + s^2\left(\frac{\vec{v}_2+\vec{v}_0}{2} - \vec{v}_6\right) + t^2\left(\frac{\vec{v}_0+\vec{v}_3}{2} - \vec{v}_7\right) \\ &+ rs(\vec{v}_0 - \vec{v}_4 + \vec{v}_5 - \vec{v}_6) + rt(\vec{v}_0 - \vec{v}_4 - \vec{v}_7 + \vec{v}_9) + st(\vec{v}_0 - \vec{v}_6 - \vec{v}_7 + \vec{v}_8) \\ &+ r(\frac{3\vec{v}_0}{2} + \frac{\vec{v}_1}{2} - 2\vec{v}_4 + \vec{v}_5 - \vec{v}_6 - \vec{v}_7 + \vec{v}_9) \\ &+ s(\frac{3\vec{v}_0}{2} + \frac{\vec{v}_2}{2} - \vec{v}_4 + \vec{v}_5 - 2\vec{v}_6 - \vec{v}_7 + \vec{v}_8) \\ &+ t(\frac{3\vec{v}_0}{2} + \frac{\vec{v}_3}{2} - \vec{v}_4 - \vec{v}_6 - 2\vec{v}_7 + \vec{v}_8 + \vec{v}_9) \\ &+ 1(\vec{v}_0 - \vec{v}_4 + \vec{v}_5 - \vec{v}_6 - \vec{v}_7 + \vec{v}_8 + \vec{v}_9)\end{aligned} \tag{3.27}$$

This allows the Jacobian to be analytically calculated by taking partial derivatives, giving

the positionally dependent Jacobian:

$$
\begin{aligned}
J(1,1) &= v_{0,x}(r+s+t+\frac{3}{2}) + v_{1,x}(r+\frac{1}{2}) + v_{4,x}(-2r-s-t-2) + v_{5,x}(s+1) \\
&\quad + v_{6,x}(-s-1) + v_{7,x}(-t-1) + v_{9,x}(t+1) \\
J(1,2) &= v_{0,y}(r+s+t+\frac{3}{2}) + v_{1,y}(r+\frac{1}{2}) + v_{4,y}(-2r-s-t-2) + v_{5,y}(s+1) \\
&\quad + v_{6,y}(-s-1) + v_{7,y}(-t-1) + v_{9,y}(t+1) \\
J(1,3) &= v_{0,z}(r+s+t+\frac{3}{2}) + v_{1,z}(r+\frac{1}{2}) + v_{4,z}(-2r-s-t-2) + v_{5,z}(s+1) \\
&\quad + v_{6,z}(-s-1) + v_{7,z}(-t-1) + v_{9,z}(t+1) \\
J(2,1) &= v_{0,x}(r+s+t+\frac{3}{2}) + v_{2,x}(s+\frac{1}{2}) + v_{4,x}(-r-1) + v_{5,x}(r+1) \\
&\quad + v_{6,x}(-r-2s-t-2) + v_{7,x}(-t-1) + v_{8,x}(t+1) \\
J(2,2) &= v_{0,y}(r+s+t+\frac{3}{2}) + v_{2,y}(s+\frac{1}{2}) + v_{4,y}(-r-1) + v_{5,y}(r+1) \\
&\quad + v_{6,y}(-r-2s-t-2) + v_{7,y}(-t-1) + v_{8,y}(t+1) \\
J(2,3) &= v_{0,z}(r+s+t+\frac{3}{2}) + v_{2,z}(s+\frac{1}{2}) + v_{4,z}(-r-1) + v_{5,z}(r+1) \\
&\quad + v_{6,z}(-r-2s-t-2) + v_{7,z}(-t-1) + v_{8,z}(t+1) \\
J(3,1) &= v_{0,x}(r+s+t+\frac{3}{2}) + v_{3,x}(t+\frac{1}{2}) + v_{4,x}(-r-1) + v_{6,x}(-s-1) \\
&\quad + v_{7,x}(-r-s-2t-2) + v_{8,x}(s+1) + v_{9,x}(r+1) \\
J(3,2) &= v_{0,y}(r+s+t+\frac{3}{2}) + v_{3,y}(t+\frac{1}{2}) + v_{4,y}(-r-1) + v_{6,y}(-s-1) \\
&\quad + v_{7,y}(-r-s-2t-2) + v_{8,y}(s+1) + v_{9,y}(r+1) \\
J(3,3) &= v_{0,z}(r+s+t+\frac{3}{2}) + v_{3,z}(t+\frac{1}{2}) + v_{4,z}(-r-1) + v_{6,z}(-s-1) \\
&\quad + v_{7,z}(-r-s-2t-2) + v_{8,z}(s+1) + v_{9,z}(r+1)
\end{aligned}
$$

$$
\boldsymbol{\mathcal{J}} = \begin{bmatrix} J(1,1) & J(1,2) & J(1,3) \\ J(2,1) & J(2,2) & J(2,3) \\ J(3,1) & J(3,2) & J(3,3) \end{bmatrix}
\tag{3.28}
$$

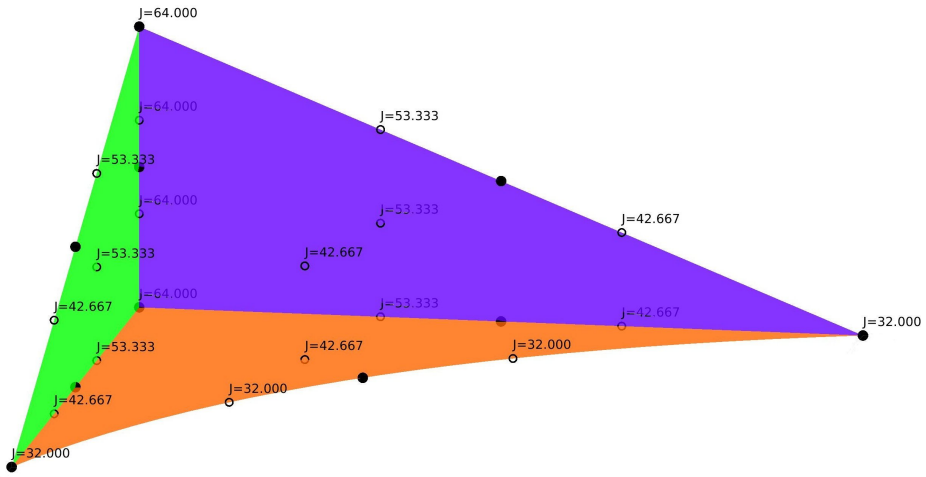
In (3.28) we see that for high-order geometries the Jacobian is positionally dependent, as it is computed by taking the partial derivatives of a 2^{nd} order function. Because the Jacobian is no longer constant within each element, it cannot be removed from the volume and surface integrals unlike the linear element case in (3.6).

$$\int_{V_n} f(\vec{x})d\vec{x} = \int_{B_n} f(\vec{r})|J(\vec{r})|d\vec{r} \tag{3.29}$$

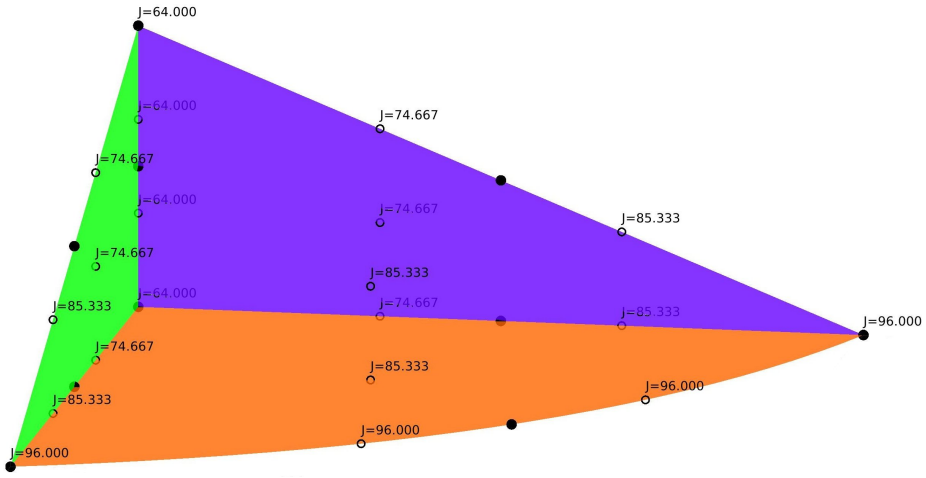
Figure 3.12 shows the non-constant Jacobian in a high-order element. 3.12 a) shows an element with a single basis point moved inward. Near this point the Jacobian is lower than the rest of the element as the volumetric expansion referenced to the canonical element is lower near this point. Alternatively, 3.12 b) shows the same element but moves the basis point outwards. This results in larger values in the Jacobian as the volumetric expansion referenced to the canonical element is higher near this point. Figure 3.13 shows the ability of high-order elements to accurately capture geometric features with its ability to shape each face.

3.5.3 Mapping to the Canonical Element

In Section 3.2.3 we showed for linear elements that if we require the value of a field at a specific position $\vec{x}$ we can map from the desired Cartesian coordinates to barycentric coordinates for the element that contains the point, and use the Lagrange basis polynomials to evaluate the field. While the mapping was simple for linear elements, high-order elements require a more complex formulation as the position can not be determined analytically. This is because for linear elements we had 3 unknowns, namely the coefficients of r, s, and t, and 3 scalar equations which mapped r, s, and t to Cartesian coordinates given in (3.1). Using this we could uniquely determine the unknowns. Now, for high-order geometries, we still



(a) A basis point is moved inward resulting in a lower Jacobian.



(b) A basis point moved outward resulting in a higher Jacobian.

Fig. 3.12: Cartesian plots showing the non-constant Jacobian within a high-order element.

desire the coefficients of r, s, and t but have shape functions with $\frac{(g+1)(g+2)(g+3)}{6}$ terms. Since we require knowledge of all 10 coefficients to map from barycentric space to Cartesian space we cannot simply invert the transformation. Instead, this problem is solved by using an iterative algorithm to determine the location.

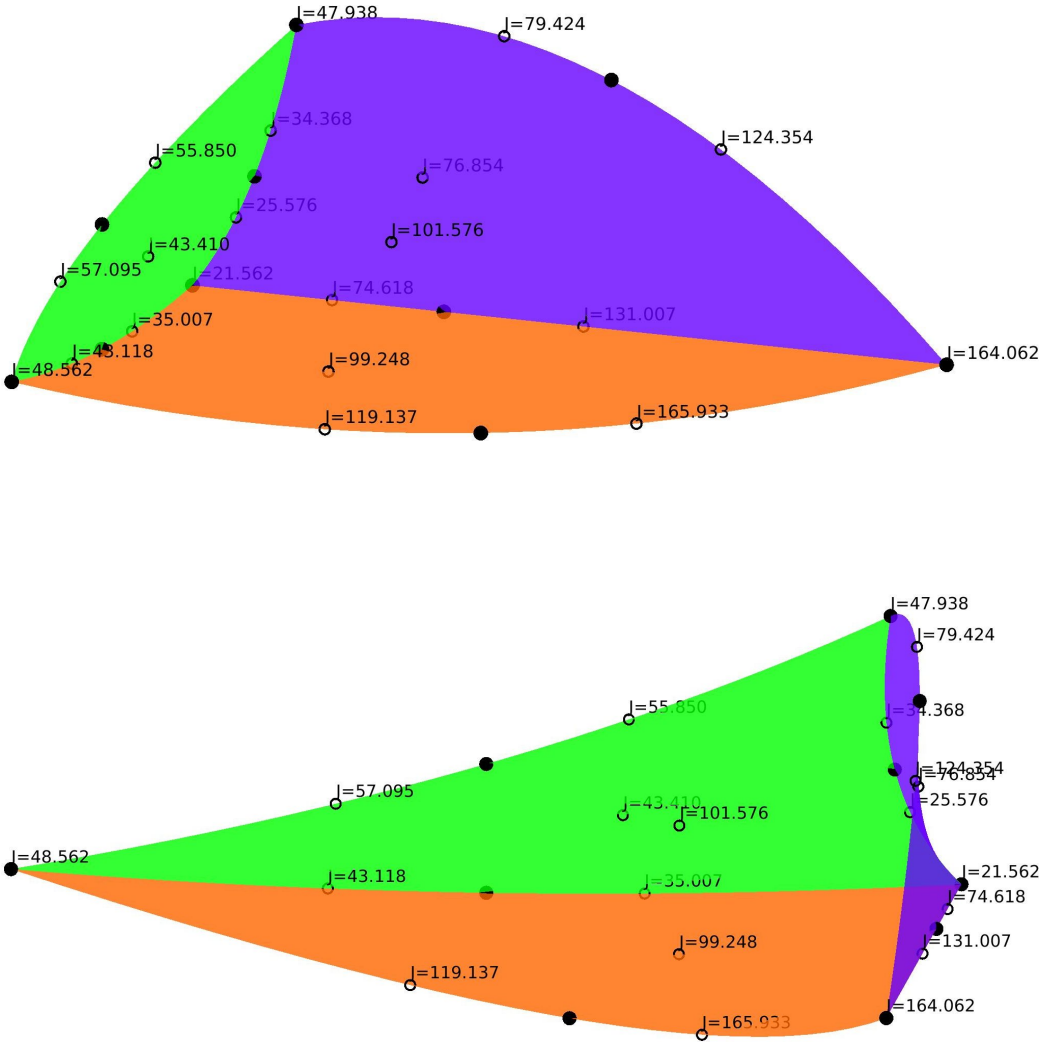


Fig. 3.13: Jacobians at various points of two elements with highly varying curvature.

Herein we choose Newton's method since it converges quadratically [19]. We create barycentric position estimates r_n, s_n, and t_n where n is the algorithm iteration. With these estimates we create Cartesian position estimates x_n, y_n, and z_n using (3.26). To obtain the equation used in the algorithm, we begin with (3.30) which formulates the Cartesian positions $\vec{x}_n$ first as a function of estimates $\vec{r}_n$, as well as by initial guesses $\vec{r}_0$ and offset $[\vec{r}_n - \vec{r}_0]$. We use mapping function f which is our standard shape functions given in (3.26).

$$\begin{bmatrix} x_n \\ y_n \\ z_n \end{bmatrix} = \begin{bmatrix} f_x(\vec{r}_n) \\ f_y(\vec{r}_n) \\ f_z(\vec{r}_n) \end{bmatrix} = \begin{bmatrix} f_x(\vec{r}_0) + \left(\frac{\partial f_x}{\partial r}(r_n - r_0) + \frac{\partial f_x}{\partial s}(s_n - s_0) + \frac{\partial f_x}{\partial t}(t_n - t_0)\right) \\ f_y(\vec{r}_0) + \left(\frac{\partial f_y}{\partial r}(r_n - r_0) + \frac{\partial f_y}{\partial s}(s_n - s_0) + \frac{\partial f_y}{\partial t}(t_n - t_0)\right) \\ f_z(\vec{r}_0) + \left(\frac{\partial f_z}{\partial r}(r_n - r_0) + \frac{\partial f_z}{\partial s}(s_n - s_0) + \frac{\partial f_z}{\partial t}(t_n - t_0)\right) \end{bmatrix} \tag{3.30}$$

Next, Cartesian terms are positioned on one side of the equation and barycentric terms on the other:

$$\begin{bmatrix} x_n - f_x(\vec{r}_0) \\ y_n - f_y(\vec{r}_0) \\ z_n - f_z(\vec{r}_0) \end{bmatrix} = \begin{bmatrix} \frac{\partial f_x}{\partial r} & \frac{\partial f_x}{\partial s} & \frac{\partial f_x}{\partial t} \\ \frac{\partial f_y}{\partial r} & \frac{\partial f_y}{\partial s} & \frac{\partial f_y}{\partial t} \\ \frac{\partial f_z}{\partial r} & \frac{\partial f_z}{\partial s} & \frac{\partial f_z}{\partial t} \end{bmatrix} \begin{bmatrix} r_n - r_0 \\ s_n - s_0 \\ t_n - t_0 \end{bmatrix} \tag{3.31}$$

The barycentric coordinates are isolated for a given iteration:

$$\begin{bmatrix} r_n \\ s_n \\ t_n \end{bmatrix} = \begin{bmatrix} r_0 \\ s_0 \\ t_0 \end{bmatrix} + \begin{bmatrix} \frac{\partial f_x}{\partial r} & \frac{\partial f_x}{\partial s} & \frac{\partial f_x}{\partial t} \\ \frac{\partial f_y}{\partial r} & \frac{\partial f_y}{\partial s} & \frac{\partial f_y}{\partial t} \\ \frac{\partial f_z}{\partial r} & \frac{\partial f_z}{\partial s} & \frac{\partial f_z}{\partial t} \end{bmatrix}^{-1} \begin{bmatrix} x_n - f_x(\vec{r}_0) \\ y_n - f_y(\vec{r}_0) \\ z_n - f_z(\vec{r}_0) \end{bmatrix} \tag{3.32}$$

This can be compactly written in vector form to give the update equation:

$$\boxed{\vec{r}_{n+1} = \vec{r}_n + \boldsymbol{\mathcal{J}}(\vec{r}_n)^{-1} \left(\vec{x}_n - \vec{x}(\vec{r}_n)\right)} \tag{3.33}$$

This equation is used to update the barycentric position estimate while minimizing the corresponding Cartesian position error. Iterations proceed until a pre-selected tolerance level is obtained.

3.6 High-Order Tetrahedral Mass and Stiffness Matrices

We now examine how the mass and stiffness matrices can be constructed for high-order elements following the same type of derivations used for linear elements in Section 3.2.3. Computing integrals for linear elements was simpler than it is for high-order elements, as the mass and stiffness matrices could be computed by scaling the integral over B_n by the Jacobian. High-order elements cannot evaluate integrals this way as the Jacobian varies over V_n and must be included in the integral. Therefore, it is necessary to evaluate the inner products directly using quadrature rules. Each quadrature rule is defined for a specific polynomial order q for which the rule will exactly integrate any polynomial of order q or less. Each rule contains N_q pairs of quadrature points $\vec{\xi_q} \in B_n$ and corresponding quadrature weights w_q. These quadrature rules and weights for tetrahedral elements are available from existing works [20].

Selection of quadrature rule order is on an element-by-element basis as we know the order of field representation, p. Since the mass and stiffness matrices multiply two p^{th} order polynomials we require a rule of order $2p$ or higher to capture the behaviour of the basis polynomials. Examining equation (3.29) we see this must be modified to take into account change in the Jacobian within the element. For 2^{nd} order elements the determinant is a 3^{rd} order function, thus we can obtain the required quadrature rule order of $2p + 3$.

In (3.34) we repeat the example of integrating the products of functions $f(\vec{x})$ and $g(\vec{x})$ over V_n, which was previously given for linear elements in (3.14). Similar to before we compute the integral over B_n as it is the easiest to define the basis functions on. Quadrature rules are defined on B_n, and account for volume discrepancy using the Jacobian. Put simply, we evaluate the integral by taking the sum over all quadrature points, of $f(\vec{\xi_q})$ times $g(\vec{\xi_q})$ times the corresponding quadrature weight w_q times the Jacobian at point $\vec{\xi_q}$.

$$\int_{V_n} f(\vec{x})g(\vec{x})d\vec{x} = \int_{B_n} f(\vec{r})g(\vec{r})|\boldsymbol{\mathcal{J}}(\vec{r})|d\vec{r} \approx \sum_{q=1}^{N_q} f(\vec{\xi}_q)g(\vec{\xi}_q)w_q|\boldsymbol{\mathcal{J}}(\vec{\xi}_q)| \tag{3.34}$$

Note that the approximation in equation (3.34) is exact if f and g are p^{th} order polynomials and a quadrature rule of order $2p+3$ or higher is used. To simplify formulation of the mass matrix, weight matrix $\boldsymbol{\mathcal{W}}$ and interpolation matrix $\boldsymbol{\mathcal{P}}(\vec{\xi_q})$ are introduced:

$$\boldsymbol{\mathcal{W}} = \begin{bmatrix} w_1|\boldsymbol{\mathcal{J}}(\vec{\xi}_1)| & & & \\ & w_2|\boldsymbol{\mathcal{J}}(\vec{\xi}_2)| & & \\ & & \ddots & \\ & & & w_{N_q}|\boldsymbol{\mathcal{J}}(\vec{\xi}_{N_q})|) \end{bmatrix} \tag{3.35}$$

$$\boldsymbol{\mathcal{P}}(\vec{\xi_q}) = \begin{bmatrix} \ell_1(\vec{\xi}_1) & \ell_2(\vec{\xi}_1) & \dots & \ell_{Np}(\vec{\xi}_1) \\ \ell_1(\vec{\xi}_2) & \ell_2(\vec{\xi}_2) & \dots & \ell_{Np}(\vec{\xi}_2) \\ \vdots & \vdots & \ddots & \vdots \\ \ell_1(\vec{\xi}_{N_q}) & \ell_2(\vec{\xi}_{N_q}) & \dots & \ell_{Np}(\vec{\xi}_{N_q}) \end{bmatrix} \tag{3.36}$$

Note that $\boldsymbol{\mathcal{P}}(\vec{\xi_q})$ simply interpolates values from the elements' basis points to the quadrature points. Using these matrices the discrete mass matrix $\boldsymbol{\mathcal{M}}_{V_n}$ is computed as:

$$\boldsymbol{\mathcal{M}}_{V_n} = \boldsymbol{\mathcal{P}}^T(\vec{\xi_q})\ \boldsymbol{\mathcal{W}}\ \boldsymbol{\mathcal{P}}(\vec{\xi_q}) \tag{3.37}$$

where the interpolation matrix interpolates fields to the quadrature points, the weight matrix scales these values by the corresponding quadrature weight and Jacobian, and the transpose of the interpolation matrix computes the inner product of these values and projects it onto the original basis points. It can be proven that for linear elements which do not require quadrature, and have a constant Jacobian, this simplifies to the linear element mass

matrix shown in (3.15).

The same example used in (3.16) shows how the high order mass matrix numerically evaluates integrals:

$$\int_{V_n} E_x(\vec{x})H_x(\vec{x})d\vec{x} \approx \underline{H}_x^H \boldsymbol{\mathcal{M}}_{V_n} \underline{E}_x \tag{3.38}$$

This is identical to the linear element case, except for construction of the mass matrix. Both linear and high-order elements construct a mass matrix of size N_p by N_p. We choose not denote these differently as we can simply write $\boldsymbol{\mathcal{M}}_{V_n}$ in the coming equations and select the appropriate matrix based on the element geometry.

Next we construct the stiffness matrices for high-order geometries which are used to evaluate the integral of a function times the derivative of another function. We note that the derivation from (3.18) which computes the derivative of our polyonmial expansions with respect to r, s, and t holds for high-order geometries. We use this to compute $\boldsymbol{\mathcal{D}}_r$, $\boldsymbol{\mathcal{D}}_s$, and $\boldsymbol{\mathcal{D}}_t$ but note two changes from the linear element derivation. First, the underlying basis polynomials from which we compute derivatives are different as they are constructed for 2^{nd} order elements. Second, the functions are evaluated at the quadrature points instead of the element's basis points as show in (3.39).

$$\boldsymbol{\mathcal{D}}_r = \begin{bmatrix} \partial_r \ell_1(\vec{\xi}_1) & \partial_r \ell_2(\vec{\xi}_1) & \dots & \partial_r \ell_{Np}(\vec{\xi}_1) \\ \partial_r \ell_1(\vec{\xi}_2) & \partial_r \ell_2(\vec{\xi}_2) & \dots & \partial_r \ell_{Np}(\vec{\xi}_2) \\ \vdots & \vdots & \ddots & \vdots \\ \partial_r \ell_1(\vec{\xi}_{N_q}) & \partial_r \ell_2(\vec{\xi}_{N_q}) & \dots & \partial_r \ell_{Np}(\vec{\xi}_{N_q}) \end{bmatrix} \tag{3.39}$$

Once again we use the chain rule in (3.21) to compute $\boldsymbol{\mathcal{D}}_x$, $\boldsymbol{\mathcal{D}}_y$, and $\boldsymbol{\mathcal{D}}_z$. We write the

stiffnes matrix for the x component, $\boldsymbol{\mathcal{S}}_x$, as:

$$\boldsymbol{\mathcal{S}}_x = \boldsymbol{\mathcal{D}}_x \, \boldsymbol{\mathcal{W}} \, \boldsymbol{\mathcal{P}}(\vec{\xi_q}) \tag{3.40}$$

Again the theory can be extended to $\boldsymbol{\mathcal{S}}_y$ and $\boldsymbol{\mathcal{S}}_z$. These can be used the same way as linear element stiffness matrices in (3.24):

$$\int_{V_n} E_x(\vec{x}) \frac{\partial H_x(\vec{x})}{\partial z} d\vec{x} \approx \underline{H}_x^H \boldsymbol{\mathcal{S}}_z \, \underline{E}_x \tag{3.41}$$

Once again both linear and high-order elements require a matrix of size N_p by N_p which we choose not to denote differently as we can select the appropriate matrix based on the element geometry. At this point we have defined the concepts required to derive DGM and HDGM. The following chapters combine these concepts, in order to create an algortihm capable of simulating electromagnetic phenomena.

4

The Discontinuous Galerkin Method

The purpose of this chapter is to introduce the DGM. While this is not the focus of this thesis, and has been done in many other works [5,13,21,22], it is necessary in order to provide meaningful discussion and point out key differences with HDGM in Chapter 5. The previous chapter focused on representing fields within an element using its basis. We now examine the conditions enforced between each set of neighbouring elements in the mesh. Combined, these are used to formulate a global system of equations which can be numerically solved to approximate the electric and magnetic fields in the computational domain. This system of equations is based on (2.17) which we repeat below for convenience.

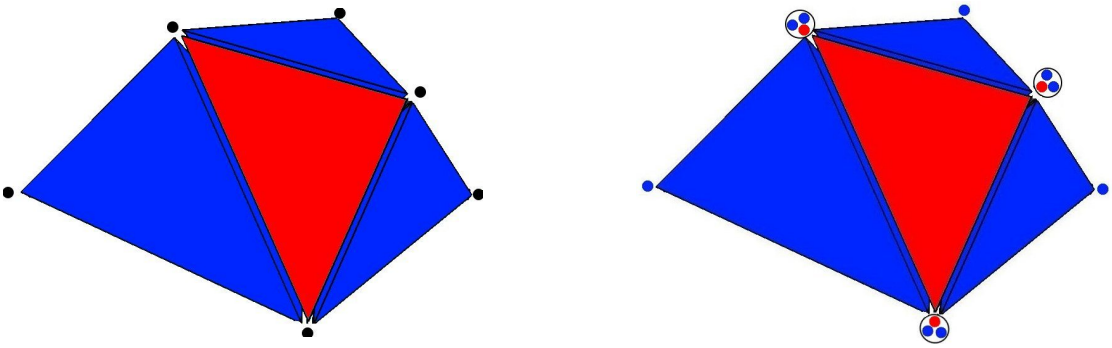
$$\int_{V_n} \boldsymbol{\psi} j\omega\varepsilon\vec{E}^{-,sct} dV_n - \int_{V_n} \boldsymbol{\psi}\nabla\times\vec{H}^{-,sct} dV_n - \oiint_{S_n} \boldsymbol{\psi}\hat{n}\times(\vec{H}^{\wedge,sct} - \vec{H}^{-,sct})\, dS_n = -\int_{V_n} \boldsymbol{\psi} j\omega(\varepsilon - \varepsilon_{bkg})\vec{E}^{-,inc} dV_n$$

$$\int_{V_n} \boldsymbol{\psi} j\omega\mu\vec{H}^{-,sct} dV_n + \int_{V_n} \boldsymbol{\psi}\nabla\times\vec{E}^{-,sct} dV_n + \oiint_{S_n} \boldsymbol{\psi}\hat{n}\times(\vec{E}^{\wedge,sct} - \vec{E}^{-,sct})\, dS_n = -\int_{V_n} \boldsymbol{\psi} j\omega(\mu - \mu_{bkg})\vec{H}^{-,inc} dV_n$$

(2.17 revisited)

The volumetric integrals in this equation can be computed using the mass and stiffness matrices previously introduced. As a Galerkin testing procedure is used, the testing functions $\boldsymbol{\psi}$ ranges over the expansion functions [21]. Evaluating the surface integrals requires computing $\hat{n}\times\vec{E}^{\wedge,sct}$ and $\hat{n}\times\vec{H}^{\wedge,sct}$, which are a function of neighbouring elements' fields. To examine how the surface integral is evaluated we investigate a small subsection of a mesh. Figure 4.1 a) shows the Cartesian coordinates and the set of vertices generated by Gmsh for 1^{st} order geometric elements. Vertices on each face of the red element are shared with a neighbouring blue element. We show the basis points for these elements in Figure 4.1 b). Since field values are discontinuous across element boundaries in the DGM, each element requires distinct values at the basis points despite the fact that vertices of neighbouring elements are collocated.

In order to evaluate the surface integral around a given element, we sum the integral over each face. Figure 4.2 examines this integral for a single face, and shows the 3^{rd} order field basis in both coordinate spaces for two neighbouring elements. The elements are aligned such that the yellow faces of their canonical element corresponds to the neighbouring faces in

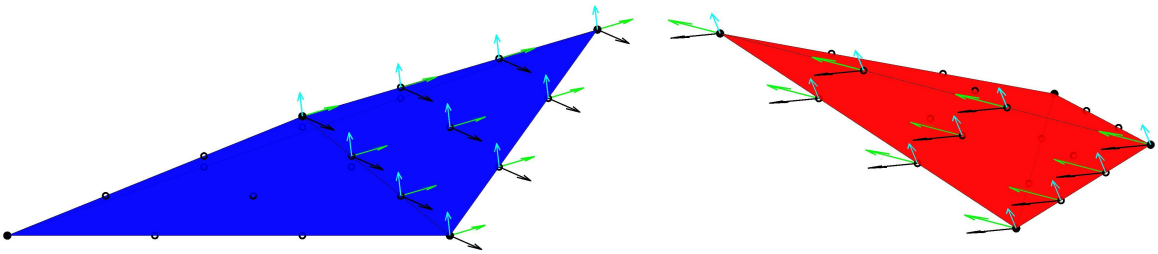


(a) Vertices of the mesh which are shared between neighbouring elements

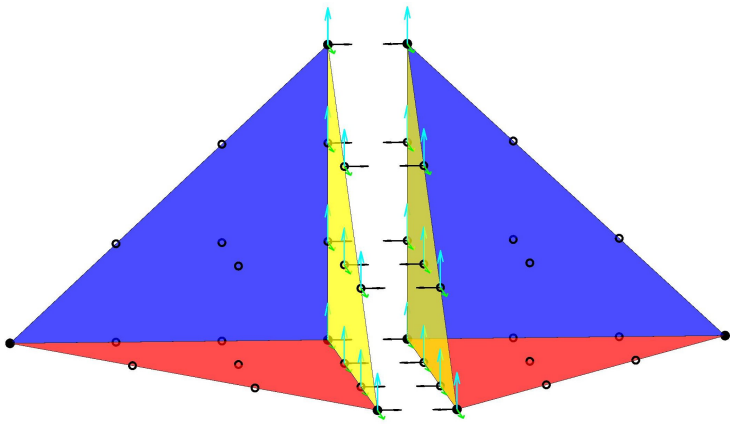
(b) Basis points coloured by the owning element. Points in each black circle are collocated

Fig. 4.1: Neighbouring elements for a small subsection of a mesh showing their vertices and basis points. Note that elements have been separated slightly for aesthetics.

Cartesian coordinates. Using the basis vectors along the face for both electric and magnetic fields, we compute $\hat{n} \times \vec{E}^{\wedge,sct}$ and $\hat{n} \times \vec{H}^{\wedge,sct}$ according to (2.14). DGM allows different field values between neighbouring faces but minimizes field discontinuity between $\hat{n} \times \vec{E}^{\wedge,sct}$ and $\hat{n} \times \vec{E}^{-,sct}$ as well as $\hat{n} \times \vec{H}^{\wedge,sct}$ and $\hat{n} \times \vec{H}^{-,sct}$ using the surface integral in the previous equation.



(a) Neighbouring elements with their DGM basis vectors used to represent fields in Cartesian coordinates on a face



(b) DGM basis vectors used to represent fields in barycentric coordinates

Fig. 4.2: Neighbouring elements with their basis vectors for each of the 3 scalar components on a shared face.

4.1 DGM Boundary Conditions

In addition to coupling the solution between neighbouring elements, we require boundary conditions for elements which terminate the computational domain Ω. The faces of these elements are used to terminate the domain using artificial numerical boundaries. Three types of field termination conditions were implemented; Silver-Müller Absorbing Boundary Conditions (ABC), Perfect Electrical Conductor (PEC) conditions, and Impedance Boundary Conditions (IBC). Each of these conditions are formulated using the neighbour boundary flux from (2.14) under different assumptions. The complete derivation of each boundary condition is shown in Appendix C.

The ABC assumes all energy propagates outward from the computational domain and attempts to eliminate reflections at the boundary. PEC conditions are assumed for metallic objects on which we have the electric field condition $\hat{n}\times\vec{E}=0$ since perfect conductors must have a constant voltage throughout. Lastly impedance conditions are used on boundaries which have an unmodeled neighbouring medium, and introduce an impedance mismatch. On these surfaces we enforce $\hat{n}\times\vec{E}=Z_s\hat{n}\times\hat{n}\times\vec{H}$ which is well known from electromagnetic theory as it states the tangential electric and magnetic fields are orthogonal and scaled by the wave impedance Z_s [23]. We show the final equations from the derivation of ABC boundaries in (4.1), PEC boundaries in (4.2), and IBC boundaries in (4.3). Note that these equations do not involve electric or magnetic fields from a neighbouring element since there are no elements outside Ω.

$$\hat{n}\times\vec{E}^{-,sct}-Z^{-}\hat{n}\times\hat{n}\times\vec{H}^{-,sct}+2Z^{-}\hat{n}\times\hat{n}\times\vec{H}^{\wedge,sct}=0 \tag{4.1}$$

$$\hat{n}\times\vec{E}^{-,sct}-Z^{-}\hat{n}\times\hat{n}\times(\vec{H}^{-,sct}-\vec{H}^{\wedge,sct})=-\hat{n}\times\vec{E}^{-,inc} \tag{4.2}$$

$$-(Z^- + Z_s)\hat{n}\times\hat{n}\times\vec{H}^{\wedge,sct} + Z^-\hat{n}\times\hat{n}\times\vec{H}^{-,sct} - \hat{n}\times\vec{E}^{-,sct} = Z_s\hat{n}\times\hat{n}\times\vec{H}^{-,inc} + \hat{n}\times\vec{E}^{-,inc} \quad (4.3)$$

4.2 Global System of Equations

The most time consuming step of DGM is solving the global system of equations. These equations numerically approximate (2.17) to couple the p^{th} order electric and magnetic fields within every element, enforce continuity in fields between neighbouring elements, and enforce boundary conditions. The system of equations takes the form $\boldsymbol{\mathcal{A}x} = \boldsymbol{b}$, where $\boldsymbol{x}$ is a concatenation of the unknown fields in each element, $\boldsymbol{b}$ is the numerical approximation to the right hand side of (2.17), and $\boldsymbol{\mathcal{A}}$ approximates the left hand side of (2.17) as a function of the unknown fields. Matrix $\boldsymbol{\mathcal{A}}$ is a block matrix where each block represents the p^{th} order fields within an element, and the block structure is determined by the connectivity of mesh elements.

We can compute the matrx structure and DOF in the matrix system using the number of elements and order of the basis within each element. We begin with the fact that a single p^{th} order element requires $(p+1)(p+2)(p+3)/6$ DOF to represent a single p^{th} order scalar variable. Since each element represents both electric and magnetic field unknowns, each with 3 components, a total of $(p+1)(p+2)(p+3)$ DOF are required for a p^{th} order element. Assuming a constant basis order throughout the mesh, and that there are many more non-boundary elements than boundary elements, DGM requires approximately $(p+1)(p+2)(p+3)N_e$ DOF for the DGM global system of equations. Figure 4.3 re-visits the small mesh used in Figure 4.1, and labels each element with a letter. These letters correspond to those used in equation (4.4) which displays the global block-matrix structure.

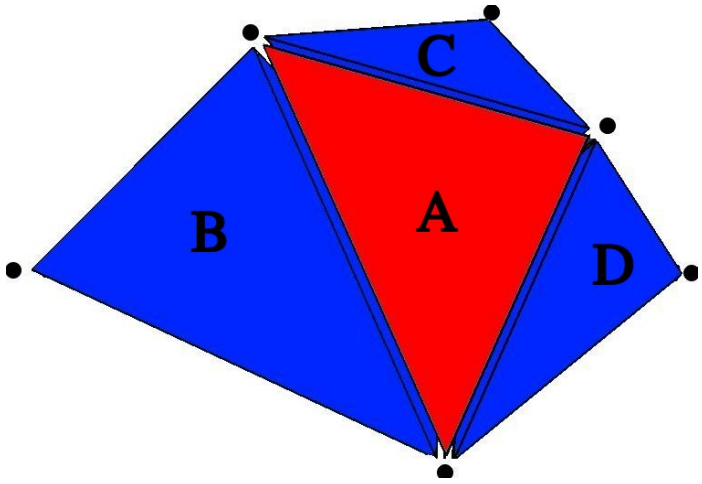
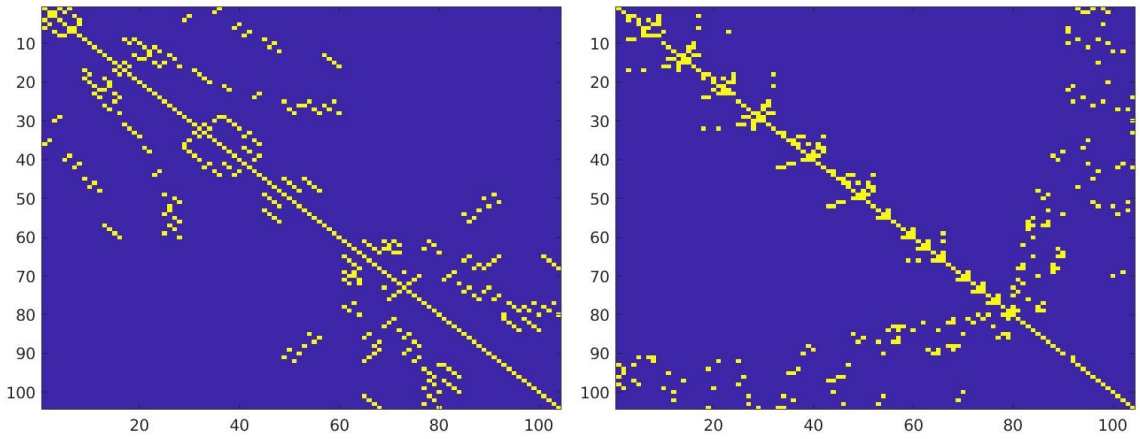


Fig. 4.3: Example mesh with labeled elements, resulting in the block-matrix structure shown in equation (4.4).

$$\left[\begin{array}{c|c|c|c} \boldsymbol{\mathcal{A}}_{AA} & \boldsymbol{\mathcal{A}}_{AB} & \boldsymbol{\mathcal{A}}_{AC} & \boldsymbol{\mathcal{A}}_{AD} \\ \hline \boldsymbol{\mathcal{A}}_{BA} & \boldsymbol{\mathcal{A}}_{BB} & & \\ \hline \boldsymbol{\mathcal{A}}_{CA} & & \boldsymbol{\mathcal{A}}_{CC} & \\ \hline \boldsymbol{\mathcal{A}}_{DA} & & & \boldsymbol{\mathcal{A}}_{DD} \end{array}\right] \left[\begin{array}{c} \underline{E}/\underline{H}_A^{sct} \\ \underline{E}/\underline{H}_B^{sct} \\ \underline{E}/\underline{H}_C^{sct} \\ \underline{E}/\underline{H}_D^{sct} \end{array}\right] = \left[\begin{array}{c} \underline{R}_A \\ \underline{R}_B \\ \underline{R}_C \\ \underline{R}_D \end{array}\right] \tag{4.4}$$

Assuming a uniform basis of order p, each block term of $\boldsymbol{\mathcal{A}}$ has $(p+1)(p+2)(p+3)$ rows and columns. Self terms $\boldsymbol{\mathcal{A}}_{ii}$ couple the electric and magnetic fields within element, while coupling terms $\boldsymbol{\mathcal{A}}_{ij}, i \neq j$ couple fields to a neighbouring element.

While Figure 4.3 is useful for demonstration purposes, it is not representative of a typical problem due to the small mesh size. In a standard mesh each non-boundary tetrahedral element will couple to four neighbouring elements resulting in four off-diagonal block terms. Boundary elements, assuming a single boundary face, will couple to three neighbouring elements and include additional terms in $\boldsymbol{\mathcal{A}}_{ii}$ to enforce boundary conditions. Figure 4.4 shows the matrix structure for a slightly larger mesh consisting of 104 elements. The matrix contains 104 block rows and columns, and each block term in the matrix has size



(a) The block-matrix structure for DGM where each block contains the unknowns for a single element.

(b) Block-matrix structure after fill-in reducing row and column permutations.

Fig. 4.4: The block-matrix structure of the DGM matrix for a mesh with 104 elements.

$(p+1)(p+2)(p+3)$. This matrix system will be compared to the matrix system created for HDGM in Section 5.5 where we see the reduction in DOF due to the HDGM formulation.

5

The Hybridizable Discontinuous Galerkin Method

The HDGM was introduced by Bernardo Cockburn in 2008 and was originally designed for convection-diffusion problems [24]. Its ability to accelerate the numerical computation has made it a popular algorithm with applications ranging from fluid dynamics, heat transfer, acoustics, electromagnetics, continuum mechanics, and more [25–28]. While some of these applications have existing publications on the use of high-order elements within the HDGM framework, this is the first to use high-order elements for electromagnetic problems.

HDGM improves performance over DGM by using a two-step solve procedure. In one step, discussed in Section 5.1, we show HDGM can compute fields within each element using only tangential fields on each face. In the other step, examined in Section 5.3, we formulate

the equations required to couple neighbouring elements that can be solved to obtain the tangential values on each face. Thus, by first constructing a system of equations for the tangential fields between elements we solve a smaller global system. Additional benefits are shown in Section 5.4 where we show the system of equations can be further reduced in size. This chapter concludes by examing the HDGM global matrix structure, and discussing benefits and drawbacks in Section 5.5.

5.1 Solving the Local System

The local system is one of the most fundamental concepts to HDGM. It is by means of the local system that we are able to express all 3 components of both the scattered electric and magnetic fields within each element, using only a single field tangential to the surface of the element. We begin by proving this and refrain from discussing the computation of these surface values until Section 5.3. The proof begins by rearranging the boundary flux equations which are repeated below.

$$\hat{n} \times \vec{E}^{\wedge,sct} \triangleq \frac{Y^-}{Y^- + Y^+}\hat{n} \times \vec{E}^{-,sct} + \frac{Y^+}{Y^- + Y^+}\hat{n} \times \vec{E}^{+,sct} - \frac{1}{Y^- + Y^+}\alpha\hat{n} \times \hat{n} \times (\vec{H}^{-,sct} - \vec{H}^{+,sct})$$

$$\hat{n} \times \vec{H}^{\wedge,sct} \triangleq \frac{Z^-}{Z^- + Z^+}\hat{n} \times \vec{H}^{-,sct} + \frac{Z^+}{Z^- + Z^+}\hat{n} \times \vec{H}^{+,sct} + \frac{1}{Z^- + Z^+}\alpha\hat{n} \times \hat{n} \times (\vec{E}^{-,sct} - \vec{E}^{+,sct})$$

(2.14 revisited)

First, $\hat{n} \times \vec{E}^{+,sct}$ is isolated from the first boundary flux equation:

$$\hat{n} \times \vec{E}^{+,sct} = \frac{Y^- + Y^+}{Y^+}\hat{n} \times \vec{E}^{\wedge,sct} - \frac{Y^-}{Y^+}\hat{n} \times \vec{E}^{-,sct} + \frac{1}{Y^+}\hat{n} \times \hat{n} \times (\vec{H}^{-,sct} - \vec{H}^{+,sct}) \quad (5.1)$$

Next, $\hat{n} \times \vec{E}^{+,sct}$ is substituted into the second boundary flux equation. We replace Y^- with

$\frac{1}{Z^-}$ as well as Y^+ with $\frac{1}{Z^+}$ and simplify:

$$\hat{n} \times \vec{H}^{\wedge,sct} = \frac{Z^-}{Z^- + Z^+}\hat{n} \times \vec{H}^{-,sct} + \frac{Z^+}{Z^- + Z^+}\hat{n} \times \vec{H}^{+,sct} + \frac{1}{Z^- + Z^+}\hat{n} \times \hat{n} \times \vec{E}^{-,sct}$$
$$- \frac{1}{Z^- + Z^+}\hat{n} \times \Big((\frac{Z^- + Z^+}{Z^-})\hat{n} \times \vec{E}^{\wedge,sct} - \frac{Z^+}{Z^-}\hat{n} \times \vec{E}^{-,sct} + Z^+\hat{n} \times \hat{n} \times (\vec{H}^{-,sct} - \vec{H}^{+,sct})\Big) \tag{5.2}$$

Equations (5.3) and (5.4) simplify this. We note that all terms involving $\vec{H}^{+,sct}$ cancel as $\vec{A}^t = -\hat{n} \times \hat{n} \times \vec{A}^t$ for any tangential surface value A^t.

$$\hat{n} \times \vec{H}^{\wedge,sct} = \frac{Z^-}{Z^- + Z^+}\hat{n} \times \vec{H}^{-,sct} + \frac{1}{Z^- + Z^+}\hat{n} \times \hat{n} \times \vec{E}^{-,sct}$$
$$- \frac{1}{Z^- + Z^+}\hat{n} \times \Big((\frac{Z^- + Z^+}{Z^-})\hat{n} \times \vec{E}^{\wedge,sct} - \frac{Z^+}{Z^-}\hat{n} \times \vec{E}^{-,sct} + Z^+\hat{n} \times \hat{n} \times \vec{H}^{-,sct}\Big) \tag{5.3}$$

$$\hat{n} \times \vec{H}^{\wedge,sct} = \hat{n} \times \vec{H}^{-,sct} + \frac{1}{Z^-}\hat{n} \times \hat{n} \times \vec{E}^{-,sct} - \frac{1}{Z^-}\hat{n} \times \hat{n} \times \vec{E}^{\wedge,sct} \tag{5.4}$$

Lastly we rearrange and simplify for the desired values:

$$\hat{n} \times (\vec{E}^{\wedge,sct} - \vec{E}^{-,sct}) = -Z^-\hat{n} \times \hat{n} \times (\vec{H}^{-,sct} - \vec{H}^{\wedge,sct}) \tag{5.5}$$

We briefly note the implication of this equation. It tells us that the error in magnetic field due to a discrepancy between the boundary flux value and a given element, taken in the plane of a face, should be equal to the admittance of the surface times the perpendicular error in electric field. We expect this from an electromagnetic point of view as this is the

basic property of electromagnetic waves [23]. Additionally, it shows us that while we enforce continuity of the magnetic field along the faces of each element, discussed further in Section 5.3, we simultaneously enforce continuity of electric field terms.

We now finish the derivation by using (5.5) and substituting it into the strong-form elemental system in (2.17). $\hat{n} \times \vec{H}^{\wedge,sct}$ is placed on the right hand side to obtain:

$$\begin{aligned}
&\int_{V_n} \boldsymbol{\psi}\ j\omega\varepsilon\vec{E}^{-,sct}dV_n - \int_{V_n} \boldsymbol{\psi}\nabla\times\vec{H}^{-,sct}dV_n + \oiint_{S_n} \boldsymbol{\psi}\hat{n}\times\vec{H}^{-,sct}\ dS_n \\
&\qquad = -\int_{V_n} \boldsymbol{\psi}\ j\omega(\varepsilon-\varepsilon_{bkg})\vec{E}^{-,inc}dV_n + \oiint_{S_n} \boldsymbol{\psi}\hat{n}\times\vec{H}^{\wedge,sct}dS_n \\
&\int_{V_n} \boldsymbol{\psi}\ j\omega\mu\vec{H}^{-,sct}dV_n + \int_{V_n} \boldsymbol{\psi}\nabla\times\vec{E}^{-,sct}dV_n - \oiint_{S_n} \boldsymbol{\psi}Z^{-}\hat{n}\times\hat{n}\times\vec{H}^{-,sct}\ dS_n \\
&\qquad = -\int_{V_n} \boldsymbol{\psi}\ j\omega(\mu-\mu_{bkg})\vec{H}^{-,inc}dV_n + \oiint_{S_n} \boldsymbol{\psi}Z^{-}\vec{H}^{\wedge,sct}dS_n
\end{aligned} \tag{5.6}$$

This form allows us to separate the equation conveniently; the left hand side containing the six field terms within V_n, and the right hand side containing a volume integral of incident fields and a surface integral of tangential magnetic fields. Under the assumption that $\vec{H}^{\wedge,sct}$ can be computed, which is proven in 5.3, we can numerically compute all terms on the right hand side of both equations since incident fields and physics are known.

The integrals on the left hand side can be numerically evaluated which is discussed further in Section 6.1. There, matrices $\boldsymbol{\mathcal{K}}_{V_n}$, $\boldsymbol{\mathcal{L}}_{V_n}$, and $\boldsymbol{\mathcal{R}}_{V_n}$ are constructed to numerically approximate

(5.6) as:

$$\boldsymbol{\mathcal{K}}_{V_n} \begin{bmatrix} \underline{E}^{-,sct} \\ \underline{H}^{-,sct} \end{bmatrix} = \boldsymbol{\mathcal{L}}_{V_n} \left[\underline{H}^{\wedge,sct} \right] + \boldsymbol{\mathcal{R}}_{V_n} \begin{bmatrix} \underline{E}^{-,inc} \\ \underline{H}^{-,inc} \end{bmatrix} \tag{5.7}$$

Since $\boldsymbol{\mathcal{K}}_{V_n}$ is invertible, we represent the scattered electric and magnetic fields within V_n as a function of the known incident fields and physics, and the magnetic fields values surrounding the element.

$$\begin{bmatrix} \underline{E}^{-,sct} \\ \underline{H}^{-,sct} \end{bmatrix} = \boldsymbol{\mathcal{K}}_{V_n}^{-1} \boldsymbol{\mathcal{L}}_{V_n} \left[\underline{H}^{\wedge,sct} \right] + \boldsymbol{\mathcal{K}}_{V_n}^{-1} \boldsymbol{\mathcal{R}}_{V_n} \begin{bmatrix} \underline{E}^{-,inc} \\ \underline{H}^{-,inc} \end{bmatrix} \tag{5.8}$$

The ability to represent all fields within each element as a function of known values and unknown magnetic surface values is what allows HDGM to construct a global system of equations with significantly fewer DOF than DGM. This is described further in Section 5.3 which also discusses computation of the magnetic surface values.

5.2 Representation of Fields

In this section we examine how the nodal basis is used by the HDGM solver to represent fields. We take advantage of the local system as it allows us to compute both electric and magnetic fields within the element using only a single field tangential to the elements' surfaces. The reduction in DOF contributes to the performance benefits of HDGM. We begin by introducing a new p^{th} order vector, $\vec{\Lambda}^t$, which is used to represent values along a surface. This is defined by (5.9) in which we use the normal to the surface to ensure tangential fields.

$$\vec{\Lambda}^t = -\hat{n} \times \hat{n} \times \vec{H}^{\wedge,sct} \tag{5.9}$$

Figure 5.1 shows the basis used to represent $\vec{\Lambda}^t$ with polynomial order $p = 6$ on a curvilinear element. We note that values along the surface at any point can be represented using two vectors tangential to the face. Since the tangent to the surface changes for a high-order element, the direction of the basis vectors also change. For linear elements the direction is constant. Using these basis vectors on all faces of the element we are able to compute the surface integral of magnetic field values in (5.6) in order to reconstruct electric and magnetic fields within the element to order p.

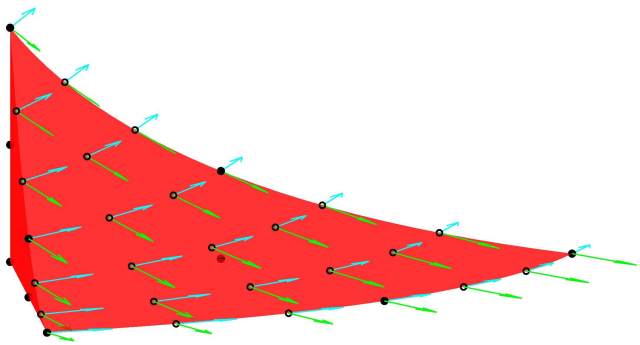


Fig. 5.1: $\vec{\Lambda}^t$ on a face of V_n to order p. Black circles represent node locations, and the green and blue vectors are the basis vectors.

We now consider the neighbour element to the element in Figure 5.1. This element is shown in Figure 5.2, and the basis vectors required for this element are shown in Figure 5.3.

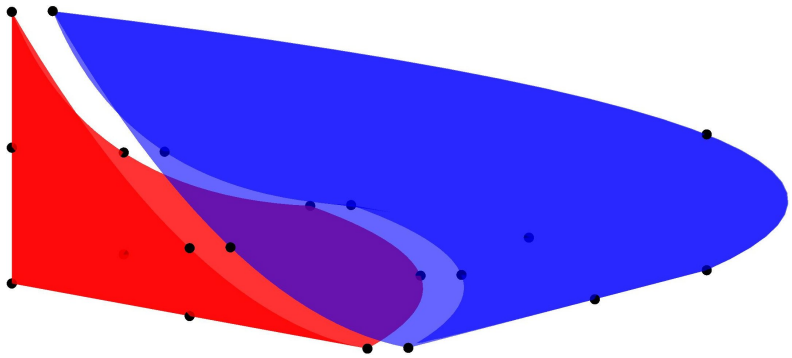


Fig. 5.2: Neighbouring curvilinear elements.

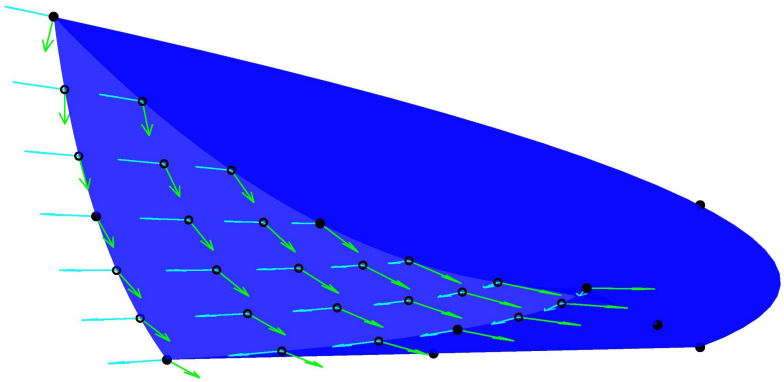


Fig. 5.3: Basis vectors required by the neighbouring element.

We can see that the vectors along the face are identical for neighbouring elements, which is expected as they share a common face. Unlike in the DGM, each element does not require its own set of basis vectors. Instead, HDGM shares basis vectors between neighbours. This is possible because the basis uses the fact that tangential fields across element boundaries are continuous.

Figure 5.4 shows the shared basis used. The vectors are shown on the numerical trace which can be thought of as *in-between* elements despite the fact that vertices are collocated. From

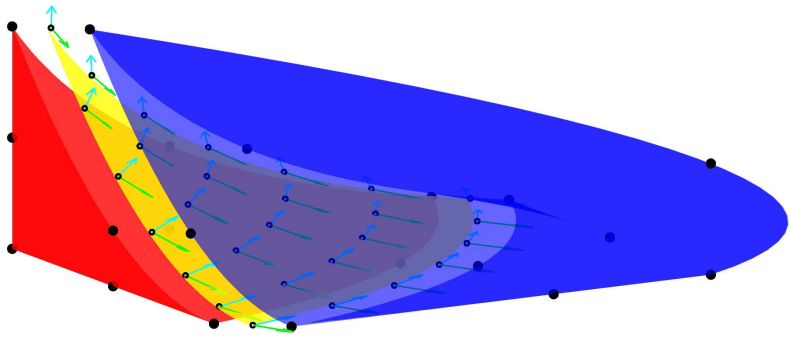


Fig. 5.4: Numerical trace with both elements' basis vectors.

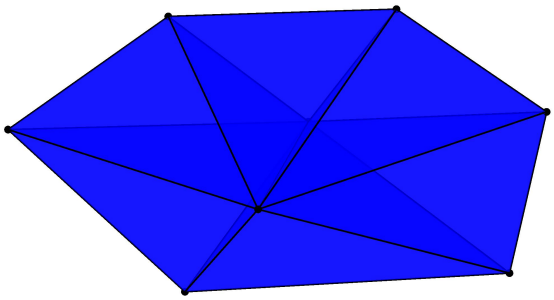
these basis vectors we are able to perform integration along the surface of either element.

5.3 Solving the Global System

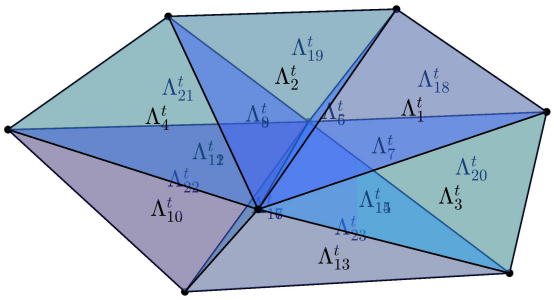
Similar to DGM, HDGM creates a global system of equations which couples the solutions between neighbouring elements. HDGM reduces the DOF in this system of equations by only using the p^{th} order tangential fields introduced in Section 5.2. After these unknowns are computed we obtain both electric and magnetic fields within each element using the local system discussed in Section 5.1. We now aim to describe the formulation of the global system of equations which is written in the form:

$$\boldsymbol{\mathcal{A}}_{\Omega}\underline{\Lambda}_{\Omega}^{t} = \underline{b}_{\Omega} \tag{5.10}$$

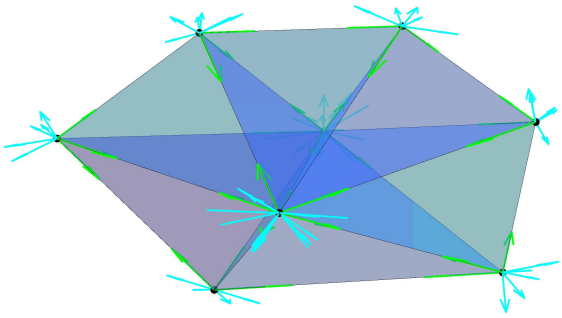
where $\boldsymbol{\mathcal{A}}_{\Omega}$ is the global HDGM matrix, $\underline{\Lambda}_{\Omega}^{t}$ is a vector of the concatenation of all tangential basis vector coefficients, and $\underline{b}_{\Omega}$ is the global right hand side vector. We examine the numerical trace, Ω_N, which consists of the union of all tangential basis vectors in Figure 5.5 [25]. This shows the mesh, the unique surfaces $\underline{\Lambda}_{\Omega}^{t}$, and the numerical trace which consists of all basis vectors.



(a) Example mesh with 1^{st} order tetrahedral elements



(b) Numerical trace of the mesh with global numbering of traces



(c) Global set of unknowns, $\underline{\Lambda}^t_\Omega$, which consists of $\underline{\Lambda}^t$ for each trace and assumes order $p = 1$

Fig. 5.5: Mesh showing the numerical trace and basis vectors.

The global system of equations is based on enforcing continuity of the tangential electric field between neighbouring elements. We use (5.5) but replace $\vec{H}^{\wedge,sct}$ with the equivalent $\vec{\Lambda}^t$ and isolate the tangential electric boundary component:

$$\hat{n} \times \vec{E}^{\wedge,sct} = \hat{n} \times \vec{E}^{-,sct} - Z^- \hat{n} \times \hat{n} \times (\vec{H}^{-,sct} - \vec{\Lambda}^t) \tag{5.11}$$

Enforcing continuity of $\hat{n} \times \vec{E}^{\wedge,sct}$ is achieved by formulating it in terms of both neighbouring elements' fields values, and setting them equal. Equation (5.12) shows $\hat{n} \times \vec{E}^{\wedge,sct}$ constructed from each element. We note that $\hat{n}$ from one elements' perspective is $-\hat{n}$ for the other.

$$\begin{aligned} &\text{+'ve element: } \hat{n} \times \vec{E}^{\wedge,sct} = \hat{n} \times \vec{E}^{-,sct} - Z^- \hat{n} \times \hat{n} \times (\vec{H}^{-,sct} - \vec{\Lambda}^t) \\ &\text{-'ve element: } -\hat{n} \times \vec{E}^{\wedge,sct} = -\hat{n} \times \vec{E}^{+,sct} - Z^+ \hat{n} \times \hat{n} \times (\vec{H}^{+,sct} - \vec{\Lambda}^t) \end{aligned} \tag{5.12}$$

By adding these equations we eliminate $\hat{n} \times \vec{E}^{\wedge,sct}$:

$$\begin{aligned} &\hat{n} \times \vec{E}^{-,sct} - Z^- \hat{n} \times \hat{n} \times (\vec{H}^{-,sct} - \vec{\Lambda}^t) \\ &- \hat{n} \times \vec{E}^{+,sct} - Z^+ \hat{n} \times \hat{n} \times (\vec{H}^{+,sct} - \vec{\Lambda}^t) = 0 \end{aligned} \tag{5.13}$$

While this equation eliminated $\hat{n} \times \vec{E}^{\wedge,sct}$, it still contains terms other than the tangential magnetic fields. The local system is used within the construction of the global system to represent internal fields as a function of only boundary field values. We show this in (5.14) where operators $f^-(\underline{\Lambda}^S_{V^-})$ and $f^+(\underline{\Lambda}^S_{V^+})$ are introduced as a function of $\underline{\Lambda}^S_{V^\pm}$, a concatenation of tangential basis vectors $\underline{\Lambda}^t$ for all faces neighbouring elements V^+ or V^-. This equation is created using the local element equation (5.8) and serves to simplify the

upcoming equations.

$$
\begin{aligned}
f^{-}(\underline{\Lambda}^{S}_{V^{-}}) &= \begin{bmatrix} \underline{E}^{-,sct} \\ \underline{H}^{-,sct} \end{bmatrix} = \boldsymbol{\mathcal{K}}^{-1}_{V^{-}} \boldsymbol{\mathcal{L}}_{V^{-}} \underline{\Lambda}^{S}_{V^{-}} + \boldsymbol{\mathcal{K}}^{-1}_{V^{-}} \boldsymbol{\mathcal{R}}_{V^{-}} \begin{bmatrix} \underline{E}^{-,inc} \\ \underline{H}^{-,inc} \end{bmatrix} \\
f^{+}(\underline{\Lambda}^{S}_{V^{+}}) &= \begin{bmatrix} \underline{E}^{+,sct} \\ \underline{H}^{+,sct} \end{bmatrix} = \boldsymbol{\mathcal{K}}^{-1}_{V^{+}} \boldsymbol{\mathcal{L}}_{V^{+}} \underline{\Lambda}^{S}_{V^{+}} + \boldsymbol{\mathcal{K}}^{-1}_{V^{+}} \boldsymbol{\mathcal{R}}_{V^{+}} \begin{bmatrix} \underline{E}^{+,inc} \\ \underline{H}^{+,inc} \end{bmatrix}
\end{aligned}
\tag{5.14}
$$

We replace the unknown scattered electric and magnetic fields from (5.13) with the values from this relation. We additionally require operators $\boldsymbol{\mathcal{S}}_E$ and $\boldsymbol{\mathcal{S}}_H$ defined in (5.15) which use identity matrix $\boldsymbol{\mathcal{I}}$ and zero matrix $\boldsymbol{\mathcal{O}}$ to select either the electric or magnetic field terms. We write the new equation in (5.16).

$$
\boldsymbol{\mathcal{S}}_E = \begin{bmatrix} \boldsymbol{\mathcal{I}} & \boldsymbol{\mathcal{O}} \end{bmatrix}, \boldsymbol{\mathcal{S}}_H = \begin{bmatrix} \boldsymbol{\mathcal{O}} & \boldsymbol{\mathcal{I}} \end{bmatrix}
\tag{5.15}
$$

$$
\begin{aligned}
&\hat{n} \times \boldsymbol{\mathcal{S}}_E f^{-}(\underline{\Lambda}^{S}_{V^{-}}) - Z^{-} \hat{n} \times \hat{n} \times \boldsymbol{\mathcal{S}}_H f^{-}(\underline{\Lambda}^{S}_{V^{-}}) - \hat{n} \times \boldsymbol{\mathcal{S}}_E f^{+}(\underline{\Lambda}^{S}_{V^{+}}) \\
&- Z^{+} \hat{n} \times \hat{n} \times \boldsymbol{\mathcal{S}}_H f^{+}(\underline{\Lambda}^{S}_{V^{+}}) + (Z^{-} + Z^{+}) \hat{n} \times \hat{n} \times \underline{\Lambda}^{t} = 0
\end{aligned}
\tag{5.16}
$$

This is a complicated equation with details that should be noted. First, we see that the equation contains only known incident fields, known physics, matrices and operators which can be computed, and values from the global set of unknown tangential magnetic fields. Next, we note a subtle distinction since there are 3 different tangential magnetic field quantities in this equation: $\underline{\Lambda}^{S}_{V^{-}}$, $\underline{\Lambda}^{S}_{V^{+}}$, and $\underline{\Lambda}^{t}$. Here, $\underline{\Lambda}^{S}_{V^{-}}$ is the set of basis vector coefficients along the surfaces of V_{-} which is required to compute the surface integral in (5.6). Similarly

$\underline{\Lambda}^S_{V^+}$ does this for the neighbouring element. $\underline{\Lambda}^t$ is the set of basis vector values along the shared trace on which we are enforcing continuous electric field conditions.

Equation (5.16) is enforced on each trace in the mesh for which there are 2 neighbours. We see that (5.16) produces values that contain all components. These values are projected onto the trace's basis vectors. Traces that contain only a single neighbour require field terminating conditions which are described below.

5.3.1 HDGM Boundary Conditions

Similar to DGM, three types of field termination conditions were implemented; ABC, PEC, and IBC. This section shows the modifications from the DGM conditions derived in Appendix C which are required for a HDGM formulation. For each of these conditions the local system is used to formulate fields as a function of $\vec{\Lambda}^t$. Similar to neighbouring element conditions, these equations produce values in full Cartesian components. Once again we project these values onto the basis vectors of the trace. We begin with the ABC condition for which we had the DGM condition below and use operators to obtain (5.17).

$$\hat{n} \times \vec{E}^{-,sct} - Z^- \hat{n} \times \hat{n} \times \vec{H}^{-,sct} + 2Z^- \hat{n} \times \hat{n} \times \vec{H}^{\wedge,sct} = 0 \qquad \text{(4.1 revisited)}$$

$$\hat{n} \times \boldsymbol{\mathcal{S}}_E f^-(\underline{\Lambda}^S_{V^-}) - Z^- \hat{n} \times \hat{n} \times \boldsymbol{\mathcal{S}}_H f^-(\underline{\Lambda}^S_{V^-}) + 2Z^- \hat{n} \times \hat{n} \times \underline{\Lambda}^t = 0 \qquad (5.17)$$

Next we formulate PEC's which had the DGM formulation shown below, and from it formulate (5.18).

$$\hat{n} \times \vec{E}^{-,sct} - Z^- \hat{n} \times \hat{n} \times (\vec{H}^{-,sct} - \vec{H}^{\wedge,sct}) = -\hat{n} \times \vec{E}^{-,inc} \qquad \text{(4.2 revisited)}$$

$$\hat{n} \times \boldsymbol{\mathcal{S}}_E f^-(\underline{\Lambda}^S_{V^-}) - Z^- \hat{n} \times \hat{n} \times \boldsymbol{\mathcal{S}}_H f^-(\underline{\Lambda}^S_{V^-}) + Z^- \hat{n} \times \hat{n} \times \underline{\Lambda}^t = -\hat{n} \times \underline{E}^{-,inc} \qquad (5.18)$$

Lastly we formulate the IBC in (5.18) from the DGM equation below.

$$-(Z^- + Z_s)\hat{n} \times \hat{n} \times \vec{H}^{\wedge,sct} + Z^- \hat{n} \times \hat{n} \times \vec{H}^{-,sct} - \hat{n} \times \vec{E}^{-,sct} = Z_s \hat{n} \times \hat{n} \times \vec{H}^{-,inc} + \hat{n} \times \vec{E}^{-,inc} \tag{4.3 revisited}$$

$$\begin{aligned} Z^- \hat{n} \times \hat{n} \times \boldsymbol{\mathcal{S}}_H f^-(\underline{\Lambda}^S_{V^-}) - \hat{n} \times \boldsymbol{\mathcal{S}}_E f^-(\underline{\Lambda}^S_{V^-}) - (Z^- + Z_s)\hat{n} \times \hat{n} \times \underline{\Lambda}^t \\ = Z_s \hat{n} \times \hat{n} \times \underline{H}^{-,inc} + \hat{n} \times \underline{E}^{-,inc} \end{aligned} \tag{5.19}$$

Using the inter-element and boundary conditions, we can formulate the global system of equations in terms of the unique tangential fields defined on the element traces. Terms involving $\underline{\Lambda}^t$ or $\underline{\Lambda}^S_{V^\pm}$ are placed in global matrix $\boldsymbol{\mathcal{A}}_\Omega$. All other terms are moved to the right hand side of the system of equations to form $\underline{b}_\Omega$. As we will see in Chapter 8 this matrix system leads to significant computational advantages. Following computation of the global tangential unknowns, we use the local system in (5.14) to compute the solution within each element.

5.4 Electric Field Formulation

So far, this thesis has focused on HDGM using tangential magnetic field unknowns as that was how it was originally implemented. However, electric field unknowns can alternatively be used. This section examines the required operators, which can be derived or determined using the electromagnetic concept of duality [23]. This section starts with the local system where we show surface electric fields on an element can be used to obtain both electric and magnetic fields within the element. Following this we formulate the global system of equations, examine boundary conditions, and discuss benefits.

5.4.1 Solving the Electric Local System

We begin by defining the p^{th} order electric field variable, $\vec{\Lambda}_E^t$, which is used to represent tangential electric field values along a surface:

$$\vec{\Lambda}_E^t = -\hat{n} \times \hat{n} \times \vec{E}^{\wedge,sct} \tag{5.20}$$

Similar to the magnetic field basis, we aim to formulate the strong-form elemental system in terms of the basis variable. This requires the dual of (5.5):

$$\hat{n} \times (\vec{H}^{\wedge,sct} - \vec{H}^{-,sct}) = -Y^- \hat{n} \times \hat{n} \times (\vec{E}^{\wedge,sct} - \vec{E}^{-,sct}) \tag{5.21}$$

This is substituted into the strong-form elemental system in (2.17) to produce (5.22), the dual of (5.6). We move $\hat{n} \times \vec{E}^{\wedge,sct}$ to the right hand side, where it is defined in terms of $\vec{\Lambda}_E^t$.

$$\begin{aligned}
&\int_{V_n} \boldsymbol{\psi} j\omega\varepsilon\vec{E}^{-,sct} dV_n - \int_{V_n} \boldsymbol{\psi}\nabla\times\vec{H}^{-,sct} dV_n - \oiint_{S_n} \boldsymbol{\psi} Y^- \hat{n}\times\hat{n}\times\vec{E}^{-,sct}\, dS_n \\
&\qquad = -\int_{V_n} \boldsymbol{\psi} j\omega(\varepsilon-\varepsilon_{bkg})\vec{E}^{-,inc} dV_n + \oiint_{S_n} \boldsymbol{\psi} Y^- \vec{\Lambda}_E^t\, dS_n \\
&\int_{V_n} \boldsymbol{\psi} j\omega\mu\vec{H}^{-,sct} dV_n + \int_{V_n} \boldsymbol{\psi}\nabla\times\vec{E}^{-,sct} dV_n - \oiint_{S_n} \boldsymbol{\psi}\hat{n}\times\vec{E}^{-,sct}\, dS_n \\
&\qquad = -\int_{V_n} \boldsymbol{\psi} j\omega(\mu-\mu_{bkg})\vec{H}^{-,inc} dV_n - \oiint_{S_n} \boldsymbol{\psi}\hat{n}\times\vec{\Lambda}_E^t\, dS_n
\end{aligned} \tag{5.22}$$

Once again this separates the equation conveniently. The left hand side contains the six field terms within V_n, and the right hand side contains a volume integral of incident fields and a surface integral of tangential electric fields. Similar to before we aim to create an operator which allows us to represent fields within the element as a function of tangential surface values. We begin with (5.23) which introduces matrix operators $\boldsymbol{\mathcal{K}}_{E,V_n}$ and $\boldsymbol{\mathcal{L}}_{E,V_n}$, as well as $\underline{\Lambda}_E^S$ which is a concatenation of tangential electric basis vectors $\underline{\Lambda}_E^t$ for each face of element V_n. The operators differ slightly from their magnetic field equivalent since the surface integrals in (5.22) are different.

$$\boldsymbol{\mathcal{K}}_{E,V_n}\begin{bmatrix}\underline{E}^{-,sct}\\ \underline{H}^{-,sct}\end{bmatrix} = \boldsymbol{\mathcal{L}}_{E,V_n}\underline{\Lambda}_E^S + \boldsymbol{\mathcal{R}}_{V_n}\begin{bmatrix}\underline{E}^{-,inc}\\ \underline{H}^{-,inc}\end{bmatrix} \tag{5.23}$$

Again we create operators to represent the scattered electric and magnetic fields within V_n as a function of known incident fields and physics, and the surface electric values. These

operators are the electric field equivalent of (5.14):

$$\begin{aligned} f^{-}(\underline{\Lambda}^{S}_{E,V^-}) &= \begin{bmatrix} \underline{E}^{-,sct} \\ \underline{H}^{-,sct} \end{bmatrix} = \boldsymbol{\mathcal{K}}^{-1}_{E,V_-} \boldsymbol{\mathcal{L}}_{E,V_-} \underline{\Lambda}^{S}_{E,V_-} + \boldsymbol{\mathcal{K}}^{-1}_{E,V_-} \boldsymbol{\mathcal{R}}_{V^-} \begin{bmatrix} \underline{E}^{-,inc} \\ \underline{H}^{-,inc} \end{bmatrix} \\ f^{+}(\underline{\Lambda}^{S}_{E,V^+}) &= \begin{bmatrix} \underline{E}^{+,sct} \\ \underline{H}^{+,sct} \end{bmatrix} = \boldsymbol{\mathcal{K}}^{-1}_{E,V_+} \boldsymbol{\mathcal{L}}_{E,V_+} \underline{\Lambda}^{S}_{E,V_+} + \boldsymbol{\mathcal{K}}^{-1}_{E,V_+} \boldsymbol{\mathcal{R}}_{V^+} \begin{bmatrix} \underline{E}^{+,inc} \\ \underline{H}^{+,inc} \end{bmatrix} \end{aligned} \tag{5.24}$$

5.4.2 Solving the Electric Global System

In this section we follow the same procedure used in Section 5.3 to formulate a global system of equations. Previously this was accomplished by using tangential magnetic fields to represent both electric and magnetic fields within the element, and continuity of the electric boundary value was enforced by evaluating it from the two adjacent elements. Now, we use tangential electric fields to represent both electric and magnetic fields within the element, and enforce continuity of the magnetic boundary value. We begin with (5.21) and isolate the tangential magnetic boundary value:

$$\hat{n} \times \vec{H}^{\wedge,sct} = \hat{n} \times \vec{H}^{-,sct} - Y^{-} \hat{n} \times \hat{n} \times (\vec{\Lambda}^{t}_{E} - \vec{E}^{-,sct}) \tag{5.25}$$

Similar to (5.12), the boundary value is defined using neighbouring elements' values:

$$\begin{aligned} &\text{+'ve element: } \hat{n} \times \vec{H}^{\wedge,sct} = \hat{n} \times \vec{H}^{-,sct} - Y^{-} \hat{n} \times \hat{n} \times (\vec{\Lambda}^{t}_{E} - \vec{E}^{-,sct}) \\ &\text{-'ve element: } -\hat{n} \times \vec{H}^{\wedge,sct} = -\hat{n} \times \vec{H}^{+,sct} - Y^{+} \hat{n} \times \hat{n} \times (\vec{\Lambda}^{t}_{E} - \vec{E}^{+,sct}) \end{aligned} \tag{5.26}$$

These equations are added to eliminate $\hat{n} \times \vec{H}^{\wedge,sct}$:

$$\begin{aligned} &\hat{n} \times \vec{H}^{-,sct} - Y^{-} \hat{n} \times \hat{n} \times (\vec{\Lambda}^{t}_{E} - \vec{E}^{-,sct}) \\ &- \hat{n} \times \vec{H}^{+,sct} - Y^{+} \hat{n} \times \hat{n} \times (\vec{\Lambda}^{t}_{E} - \vec{E}^{+,sct}) = 0 \end{aligned} \tag{5.27}$$

Next, we use the local operators in (5.24) to represent internal electric and magnetic fields as a function of $\underline{\Lambda}_E^S$. We substitute values into (5.27) and use the selection matrices from (5.15) to obtain the dual of (5.16):

$$\begin{aligned}&\hat{n}\times\boldsymbol{\mathcal{S}}_H f^-(\underline{\Lambda}^S_{E,V^-})+Y^-\hat{n}\times\hat{n}\times\boldsymbol{\mathcal{S}}_E f^-(\underline{\Lambda}^S_{E,V^-})-\hat{n}\times\boldsymbol{\mathcal{S}}_H f^+(\underline{\Lambda}^S_{E,V^+})\\&+Y^+\hat{n}\times\hat{n}\times\boldsymbol{\mathcal{S}}_E f^+(\underline{\Lambda}^S_{E,V^+})-(Y^-+Y^+)\hat{n}\times\hat{n}\times\underline{\Lambda}^t_E=0\end{aligned}\tag{5.28}$$

A global system of equations is created by enforcing (5.28) on each trace in the mesh for which there are 2 neighbours and projecting the value onto the trace's basis vectors. Field terminating conditions require modification which are described in the following section.

Modification to Boundary Conditions

Once again the boundary conditions are derived in Appendix C. From these derivations we obtain the ABC in (5.29) and the IBC in (5.30).

$$\hat{n}\times\vec{H}^{-,sct}+Y^-\hat{n}\times\hat{n}\times\vec{E}^{-,sct}+2Y^-\vec{\Lambda}^t_E=0\tag{5.29}$$

$$\begin{aligned}(Y^-+Y_{boundary})\vec{\Lambda}^t_E+Y^-\hat{n}\times\hat{n}\times\vec{E}^{-,sct}+\hat{n}\times\vec{H}^{-,sct}&\\=Y_{boundary}\hat{n}\times\hat{n}\times\vec{E}^{-,inc}-\hat{n}\times\vec{H}^{-,inc}&\end{aligned}\tag{5.30}$$

Unlike DGM and HDGM using a magnetic field basis, HDGM using an electric field basis does not require enforcing PEC conditions. This is because on any PEC surface we enforce $\hat{n}\times\vec{E}^{tot}=\hat{n}\times\vec{E}^{inc}+\hat{n}\times\vec{E}^{sct}=0$ for tangential field components. Since the basis vectors are tangential to the PEC surface, and the incident fields are known, we can compute scattered field values prior to solving the global system of equations. This eliminates these unknowns, further reducing the size of the global system of equations. We conclude this section by using the local system for electric fields in (5.24) to represent all variables as a function of

tangential electric fields for ABC and IBC in (5.31) and (5.32) respectively.

$$\hat{n} \times \boldsymbol{\mathcal{S}}_H f^-(\underline{\Lambda}^S_{V^-}) + Y^- \hat{n} \times \hat{n} \times \boldsymbol{\mathcal{S}}_E f^-(\underline{\Lambda}^S_{V^-}) + 2Y^- \underline{\Lambda}^t_E = 0 \tag{5.31}$$

$$\begin{aligned}(Y^- + Y_{boundary})\underline{\Lambda}^t_E + Y^- \hat{n} \times \hat{n} \times \boldsymbol{\mathcal{S}}_E f^-(\underline{\Lambda}^S_{V^-}) + \hat{n} \times \boldsymbol{\mathcal{S}}_H f^-(\underline{\Lambda}^S_{V^-}) \\ = Y_{boundary} \hat{n} \times \hat{n} \times \underline{E}^{-,inc} - \hat{n} \times \underline{H}^{-,inc}\end{aligned} \tag{5.32}$$

5.5 A Comparison of DGM and HDGM

We begin this section by comparing the equations enforced by DGM, HDGM using a magnetic field basis, and HDGM using an electric field basis. These can be easily compared by removing the mass, stiffness, and incident field terms as these are common between all 3 formulations. This leaves only the flux terms which couple neighbouring elements. We begin with the original DGM equation introduced in (2.17) which enforces the flux below.

$$\begin{aligned} \oiint_{S_n} \boldsymbol{\psi}\hat{n} \times \vec{H}^{sct}\, dS_n &= \oiint_{S_n} \boldsymbol{\psi}\hat{n} \times \vec{H}^{\wedge,sct} dS_n \\ -\oiint_{S_n} \boldsymbol{\psi}\hat{n} \times \vec{E}^{sct}\, dS_n &= -\oiint_{S_n} \boldsymbol{\psi}\hat{n} \times \vec{E}^{\wedge,sct}\, dS_n \end{aligned} \tag{5.33}$$

DGM formulates equations using all 3 components of both electric and magnetic fields, and can therefore represent both electric and magnetic boundary values. It simply enforces that each elements' values match boundary values, but requires a higher number of DOF to do so. Next, we see the flux enforced by HDGM using a magnetic field basis which comes from (5.6).

$$\begin{aligned} \oiint_{S_n} \boldsymbol{\psi}\hat{n} \times \vec{H}^{sct}\, dS_n &= \oiint_{S_n} \boldsymbol{\psi}\hat{n} \times \vec{\Lambda}^t dS_n \\ -\oiint_{S_n} \boldsymbol{\psi} Z^- \hat{n} \times \hat{n} \times \vec{H}^{sct}\, dS_n &= \oiint_{S_n} \boldsymbol{\psi} Z^- \vec{\Lambda}^t dS_n \end{aligned} \tag{5.34}$$

The first equation enforced by HDGM is the same as that used by DGM, but uses the tangential magnetic field basis to do so. The second equation differs from DGM, as it uses an integral of the impedance times the magnetic fields to compute the surface integral of electric fields. This shows HDGM can properly compute flux using the magnetic field basis to represent all unknowns. Last, we compre the flux representation when using an electric

field basis which was formulated in (5.22).

$$\begin{aligned} -\oiint_{S_n} \boldsymbol{\psi} Y^- \hat{n} \times \hat{n} \times \vec{E}^{-,sct}\, dS_n &= \oiint_{S_n} \boldsymbol{\psi} Y^- \vec{\Lambda}_E^t\, dS_n \\ -\oiint_{S_n} \boldsymbol{\psi} \hat{n} \times \vec{E}^{sct}\, dS_n &= -\oiint_{S_n} \boldsymbol{\psi} \hat{n} \times \vec{\Lambda}_E^t\, dS_n \end{aligned} \tag{5.35}$$

We see that when using an electric field basis the second equation is equivalent to the DGM flux formulation. The first equation now differs from DGM, as it uses an integral of the surface admittance times the electric fields to compute the surface integral of magnetic fields.

As we will see in the results, these formulations give identical solutions to machine precision. However, HDGM provides significant computational advantages over DGM. One way it does this is by use of a single tangential field component to represent both electric and magnetic fields, unlike DGM which represents both as unknowns. In addition HDGM couples neighbouring elements using a continuous variable on the trace between neighbouring elements as opposed to DGM which uses collocated nodal values. This has the added advantage that basis coefficients can be represented using 2 tangential vectors to the trace, unlike DGM which requires basis vectors for each of the 3 Cartesian components.

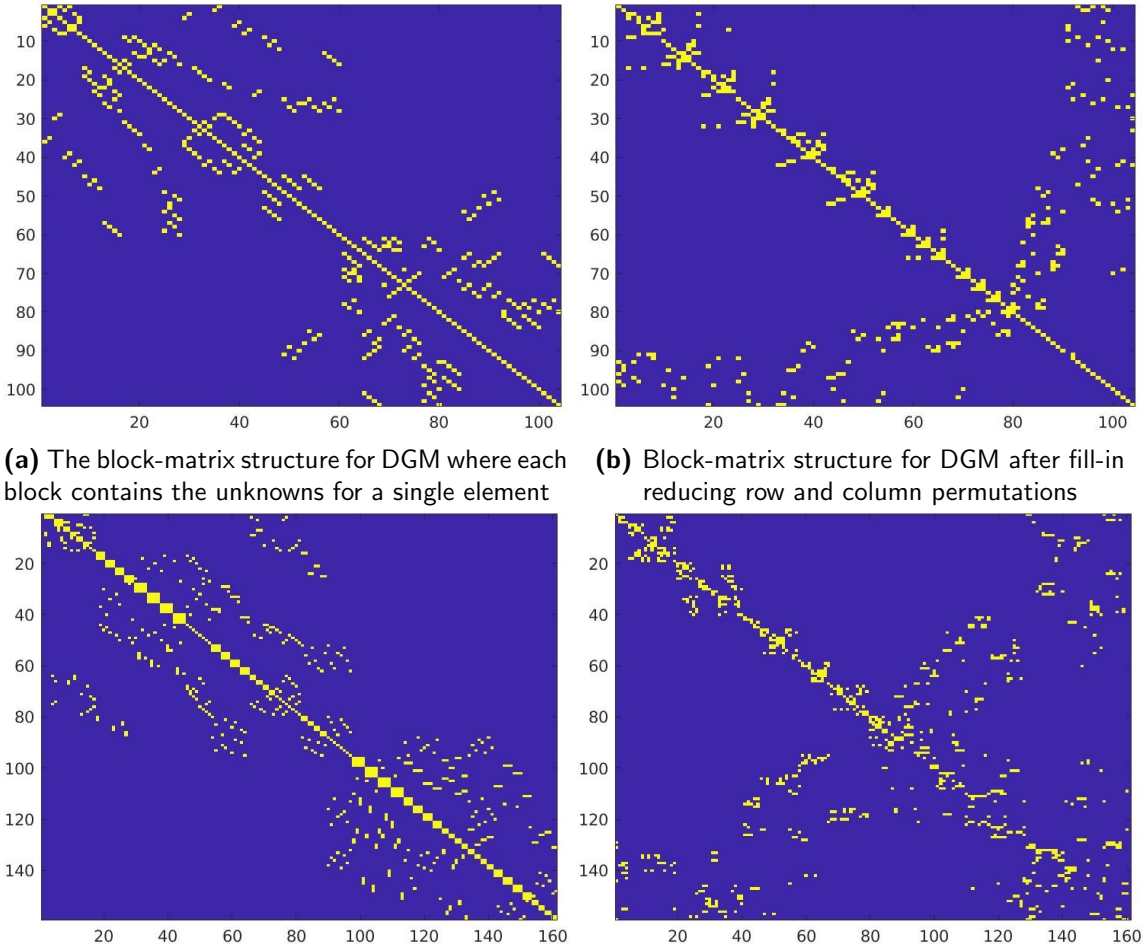
The basis used by HDGM changes the structure of the global matrix compared to DGM. The matrix still has a block structure, but each block now represents the fields on a given trace instead of a given element. This requires $(p+1)(p+2)$ unknowns since HDGM requires two p^{th} order vectors to represent the fields on a trace. Flux is enforced on each trace, and is computed using the fields of the 2 adjacent elements. These fields are computed as a function of the basis coefficients on each of the 4 faces. Combined, this results in 7 unique traces which contribute directly to the equations in the global system associated with a given trace. This results in a diagonal block and 6 off-diagonal blocks for each block-row

in the global matrix; more than the 4 off-diagonal terms required by DGM. While the additional off-diagonal terms are undesireable, the reduced size of each block allows much better overall performance.

We now compare the size and fill-in of the global matrix between DGM and HDGM using the example introduced in Figure 4.4. We show the matrix structure for both DGM and HDGM in Figure 5.6. HDGM requires more blocks terms, as the unknowns are now for each trace instead of each element. However while the number of blocks increased from 104 to 160, the size of each block has shrunk from $(p+1)(p+2)(p+3)$ unknowns to $(p+1)(p+2)$ unknowns. We also see HDGM has a worse fill-in, after row and column permutations, compared to DGM. This is attributed to the fact that block rows for HDGM couple to 6 columns, while those in DGM only couple to 4.

Assuming there are many more non-boundary elements in a mesh than boundary elements, we approximate the number of faces, N_f, to be two times the number of elements, N_e. This results in HDGM requiring a total of $2(p+1)(p+2)N_e$ DOF assuming uniform solution order throughout a mesh. We compare this to the $(p+1)(p+2)(p+3)N_e$ DOF required for DGM and see that HDGM reduces the DOF by a factor of $(p+3)/2$. Furthermore, when using the electric field basis all unknowns on PEC surfaces can be removed from the DOF. This is especially useful when imaging in metallic chambers.

One drawback of HDGM is an increase in global matrix construction time. This is largely due to the inversion of the local system matrix $\boldsymbol{\mathcal{K}}_{V_n}$ for every element, which is a dense matrix of size $(p+1)(p+2)(p+3)$. However, since the majority of computational time is used to solve the global matrix system, the reduced DOF provided by the HDGM formulation easily outweights this cost.



(a) The block-matrix structure for DGM where each block contains the unknowns for a single element

(b) Block-matrix structure for DGM after fill-in reducing row and column permutations

(c) The block-matrix structure for HDGM where each block contains the unknowns for a single trace

(d) Block-matrix structure after fill-in reducing row and column permutations

Fig. 5.6: The block-matrix structure for the global DGM and HDGM matrices when a mesh with 104 elements and 160 traces is selected.

6

Implementation Details

This chapter aims to provide missing details on the implementation of HDGM using the theory and operators introduced in previous chapters. It focuses on the operators required when using a magnetic field basis and omits discussion of the operators required for an electric field basis as they are nearly identical. Since HDGM is computationally demanding, steps were taken to ensure the algorithm ran as fast as possible. This section also includes details on the code-level implementation used, including an MPI parallel implementation discussed in Section 6.3 [29]. We see in the results that this implementation results in significantly improved performance. Global matrices were created using the Portable, Extensible Toolkit for Scientific Computation software package (PETSc) software package which utilizes a multifrontal massively parallel sparse direct solver software package (MUMPS) to perform matrix factorization [30, 31]. Representation of the matrix uses a block sparse format where each square block represents a trace, and the matrix uses a compressed sparse

row format to save the location of blocks. BLAS and LAPACK routines are used to construct smaller matrices, perform matrix-vector or matrix-matrix products, and to invert $\boldsymbol{\mathcal{K}}_{V_n}$ for each element [32]. We begin by describing implementation of the local system in Section 6.1 followed by the global system in Section 6.2.

6.1 Implementation of the Local System

In order to implement the local system of equations for each element we require 3 matrices; $\boldsymbol{\mathcal{K}}_{V_n}$, $\boldsymbol{\mathcal{L}}_{V_n}$, and $\boldsymbol{\mathcal{R}}_{V_n}$ which are used in (5.8). While these are constructed on an element-by-element basis, we introduce a global matrix form in Section 6.1.4 which is required for derivations in Chapter 7.

Implementation of the local system also benefits from further description of the scalar testing functions $\boldsymbol{\psi}$, which have been largely notational. We can compute the required number of testing functions as each element contains $6Np$ scalar unknowns, namely the three Cartesian components of the electric and magnetic fields. In order to sufficiently test, $6Np$ equations are generated from using Np testing functions for all three components in two equations. Each point $\vec{R}_i, i \in Np$ is tested:

$$\boldsymbol{\psi}(\vec{R}_i) = \begin{bmatrix} \psi_1(\vec{R}_i) & \psi_2(\vec{R}_i) & \dots & \psi_{Np}(\vec{R}_i) \end{bmatrix} = \begin{bmatrix} \ell_1(\vec{R}_i) & \ell_2(\vec{R}_i) & \dots & \ell_{Np}(\vec{R}_i) \end{bmatrix} \tag{6.1}$$

6.1.1 Construction of $\boldsymbol{\mathcal{K}}_{V_n}$

$\boldsymbol{\mathcal{K}}_{V_n}$ approximates the following terms from the HDGM global system:

$$\boldsymbol{\mathcal{K}}_{V_n} \begin{bmatrix} \underline{E}^{sct} \\ \underline{H}^{sct} \end{bmatrix} \approx \begin{bmatrix} \int\limits_{V_n} \boldsymbol{\psi}\ j\omega\varepsilon\vec{E}^{sct} dV_n & -\int\limits_{V_n} \boldsymbol{\psi}\nabla\times\vec{H}^{sct} dV_n & +\oiint\limits_{S_n} \boldsymbol{\psi}\hat{n}\times\vec{H}^{sct}\, dS_n \\ \int\limits_{V_n} \boldsymbol{\psi}\ j\omega\mu\vec{H}^{sct} dV_n & +\int\limits_{V_n} \boldsymbol{\psi}\nabla\times\vec{E}^{sct} dV_n & -\oiint\limits_{S_n} \boldsymbol{\psi}Z^{-}\hat{n}\times\hat{n}\times\vec{H}^{sct}\, dS_n \end{bmatrix} \tag{6.2}$$

We break (6.2) into parts beginning with the first term from both equations. The mass matrix is used to numerically compute the integrals component-by-component:

$$\begin{bmatrix} \int\limits_{V_n} \boldsymbol{\psi}\ j\omega\varepsilon\vec{E}^{sct}dV_n \\ \int\limits_{V_n} \boldsymbol{\psi}\ j\omega\mu\vec{H}^{sct}dV_n \end{bmatrix} \approx j\omega \begin{bmatrix} \varepsilon\boldsymbol{\mathcal{M}} & & & & & & \\ & \varepsilon\boldsymbol{\mathcal{M}} & & & & & \\ & & \varepsilon\boldsymbol{\mathcal{M}} & & & & \\ & & & \mu\boldsymbol{\mathcal{M}} & & & \\ & & & & \mu\boldsymbol{\mathcal{M}} & & \\ & & & & & \mu\boldsymbol{\mathcal{M}} & \\ & & & & & & \mu\boldsymbol{\mathcal{M}} \end{bmatrix} \begin{bmatrix} \underline{E}_x \\ \underline{E}_y \\ \underline{E}_z \\ \underline{H}_x \\ \underline{H}_y \\ \underline{H}_z \end{bmatrix} \tag{6.3}$$

Note that each block $\boldsymbol{\mathcal{M}}$ and corresponding $\underline{E}_i/\underline{H}_i$ are of size Np, which is due to the fact that it is built for all testing functions.

Next we look at the second terms in (6.2) in which we test the curl of the electric and magnetic fields. We break this into scalar components in (6.4) where stiffness matrices are used to compute derivatives and perform integration.

$$\begin{bmatrix} -\int\limits_{V_n} \boldsymbol{\psi}\nabla\times\vec{H}^{sct}dV_n \\ \int\limits_{V_n} \boldsymbol{\psi}\nabla\times\vec{E}^{sct}dV_n \end{bmatrix} \approx \begin{bmatrix} & & & & \boldsymbol{\mathcal{S}}_z & -\boldsymbol{\mathcal{S}}_y \\ & & & -\boldsymbol{\mathcal{S}}_z & & \boldsymbol{\mathcal{S}}_x \\ & & & \boldsymbol{\mathcal{S}}_y & -\boldsymbol{\mathcal{S}}_x & \\ & -\boldsymbol{\mathcal{S}}_z & \boldsymbol{\mathcal{S}}_y & & & \\ \boldsymbol{\mathcal{S}}_z & & -\boldsymbol{\mathcal{S}}_x & & & \\ -\boldsymbol{\mathcal{S}}_y & \boldsymbol{\mathcal{S}}_x & & & & \end{bmatrix} \begin{bmatrix} \underline{E}_x \\ \underline{E}_y \\ \underline{E}_z \\ \underline{H}_x \\ \underline{H}_y \\ \underline{H}_z \end{bmatrix} \tag{6.4}$$

The last component needed to construct $\boldsymbol{\mathcal{K}}_{V_n}$ are the flux matrices in the last term of (6.2). Here, a surface integral is performed by adding the integrals over each face. In order to

perform these integrals, we recall the high-order mass matrix:

$$\boldsymbol{\mathcal{M}}_{V_n} = \boldsymbol{\mathcal{P}}^T(\vec{\xi_q})\ \boldsymbol{\mathcal{W}}\ \boldsymbol{\mathcal{P}}(\vec{\xi_q}) \qquad \text{(3.37 revisited)}$$

While this matrix is constructed for 3D elements, the mass matrix required to integrate along a face is almost identical as it is simply a 2D mass matrix. We write the 2D mass matrix as $\boldsymbol{\mathcal{P}}_f^T\ \boldsymbol{\mathcal{W}}_f\ \boldsymbol{\mathcal{P}}_f$, where $\boldsymbol{\mathcal{W}}_f$ is a diagonal matrix of the integration weight times the Jacobian of the transformation at each quadrature point on the face, and $\boldsymbol{\mathcal{P}}_f$ is an interpolation matrix to the quadrature points on the given face. $\underline{\hat{n}}_i$ is used to simplify the equation and is $\boldsymbol{\mathcal{W}}_f$ times the Cartesian component i of the unit normal at each of the quadrature points on the face. Using this notation, we write the last term of $\boldsymbol{\mathcal{K}}_{V_n}$ as:

$$\begin{bmatrix} \oiint_{S_n} \boldsymbol{\psi}\hat{n} \times \vec{H}^{sct}\, dS_n \\ -\oiint_{S_n} \boldsymbol{\psi} Z^- \hat{n} \times \hat{n} \times \vec{H}^{sct}\, dS_n \end{bmatrix} \approx \sum_{Face=1}^{4}$$

$$\begin{bmatrix} -\boldsymbol{\mathcal{P}}_f^T \underline{\hat{n}}_z \boldsymbol{\mathcal{P}}_f & \boldsymbol{\mathcal{P}}_f^T \underline{\hat{n}}_y \boldsymbol{\mathcal{P}}_f & \\ \boldsymbol{\mathcal{P}}_f^T \underline{\hat{n}}_z \boldsymbol{\mathcal{P}}_f & & -\boldsymbol{\mathcal{P}}_f^T \underline{\hat{n}}_x \boldsymbol{\mathcal{P}}_f \\ -\boldsymbol{\mathcal{P}}_f^T \underline{\hat{n}}_y \boldsymbol{\mathcal{P}}_f & \boldsymbol{\mathcal{P}}_f^T \underline{\hat{n}}_x \boldsymbol{\mathcal{P}}_f & \\ & & \\ -\boldsymbol{\mathcal{P}}_f^T \underline{Z}^- (\underline{\hat{n}}_y \underline{\hat{n}}_y + \underline{\hat{n}}_z \underline{\hat{n}}_z) \boldsymbol{\mathcal{P}}_f & \boldsymbol{\mathcal{P}}_f^T \underline{Z}^- \underline{\hat{n}}_x \underline{\hat{n}}_y \boldsymbol{\mathcal{P}}_f & \boldsymbol{\mathcal{P}}_f^T \underline{Z}^- \underline{\hat{n}}_x \underline{\hat{n}}_z \boldsymbol{\mathcal{P}}_f \\ \boldsymbol{\mathcal{P}}_f^T \underline{Z}^- \underline{\hat{n}}_x \underline{\hat{n}}_y \boldsymbol{\mathcal{P}}_f & -\boldsymbol{\mathcal{P}}_f^T \underline{Z}^- (\underline{\hat{n}}_x \underline{\hat{n}}_x + \underline{\hat{n}}_z \underline{\hat{n}}_z) \boldsymbol{\mathcal{P}}_f & \boldsymbol{\mathcal{P}}_f^T \underline{Z}^- \underline{\hat{n}}_y \underline{\hat{n}}_z \boldsymbol{\mathcal{P}}_f \\ \boldsymbol{\mathcal{P}}_f^T \underline{Z}^- \underline{\hat{n}}_x \underline{\hat{n}}_z \boldsymbol{\mathcal{P}}_f & \boldsymbol{\mathcal{P}}_f^T \underline{Z}^- \underline{\hat{n}}_y \underline{\hat{n}}_z \boldsymbol{\mathcal{P}}_f & -\boldsymbol{\mathcal{P}}_f^T \underline{Z}^- (\underline{\hat{n}}_z \underline{\hat{n}}_z + \underline{\hat{n}}_y \underline{\hat{n}}_y) \boldsymbol{\mathcal{P}}_f \end{bmatrix} \begin{bmatrix} \underline{H}_x^{Face} \\ \underline{H}_y^{Face} \\ \underline{H}_z^{Face} \end{bmatrix} \tag{6.5}$$

Since every term between $\boldsymbol{\mathcal{P}}_f^T$ and $\boldsymbol{\mathcal{P}}_f$ in this matrix in purely diagonal, we can efficiently implement it by row-scaling $\boldsymbol{\mathcal{P}}_f$. Since these matrices are stored in row-major format this allows quick coalesced memory access, and minimal computational time. This leaves

the matrix-matrix multiplication of $\boldsymbol{\mathcal{P}}_f^T$ with this row-scaled matrix as the only major computational cost. Since 6 terms in this matrix appear twice, they can be reused and do not need not be re-computed. Furthermore, when physics are constant within each element, Z^- can be accounted for after the matrix-matrix product, allowing further acceleration by using real data types instead of complex data types for the matrix-matrix product.

We combine matrices (6.3), (6.4), and (6.5) to form $\boldsymbol{\mathcal{K}}_{V_n}$. In practice, this matrix is not built by adding 3 matrices, but instead by filling a single matrix with dense sub-blocks allowing much faster construction. Each of the 6 block terms in (6.3) only requires copying the element's mass matrix and scaling terms by the permittivity of permeability, and each of the 12 block terms in (6.4) requires coping the stiffness matrix and possibly flipping the sign. This leaves the only major costs of constructing the local system as the 9 matrix-matrix products of matrices size Np by Nqp times matrices size Nqp by Np for each face of each element, as well as the inversion of $\boldsymbol{\mathcal{K}}_{V_n}$ which is a square matrix of size $6Np$ by $6Np$ for each element.

6.1.2 Construction of $\boldsymbol{\mathcal{L}}_{V_n}$

Similar to the last term of $\boldsymbol{\mathcal{K}}_{V_n}$, $\boldsymbol{\mathcal{L}}_{V_n}$ computes a surface integral by adding the integral for each face of V_n. It approximates the following terms on the right hand side of (5.6).

$$\boldsymbol{\mathcal{L}}_{V_n}\underline{\Lambda}_S^t \approx \begin{bmatrix} \oiint_{S_n} \boldsymbol{\psi}\hat{n} \times \vec{\Lambda}_S^t \, dS_n \\ \oiint_{S_n} \boldsymbol{\psi} Z^- \vec{\Lambda}_S^t \, dS_n \end{bmatrix} \tag{6.6}$$

We begin by constructing this operator assuming it operates on Cartesian terms in (6.7).

$$\boldsymbol{\mathcal{L}}_{V_n}\begin{bmatrix}\underline{\Lambda}_x\\ \underline{\Lambda}_y\\ \underline{\Lambda}_z\end{bmatrix} = \sum_{Face=1}^{4}\begin{bmatrix} & \boldsymbol{\mathcal{P}}_f^T\underline{\hat{n}}_z\boldsymbol{\mathcal{P}}_f & -\boldsymbol{\mathcal{P}}_f^T\underline{\hat{n}}_y\boldsymbol{\mathcal{P}}_f\\ -\boldsymbol{\mathcal{P}}_f^T\underline{\hat{n}}_z\boldsymbol{\mathcal{P}}_f & & \boldsymbol{\mathcal{P}}_f^T\underline{\hat{n}}_x\boldsymbol{\mathcal{P}}_f\\ \boldsymbol{\mathcal{P}}_f^T\underline{\hat{n}}_y\boldsymbol{\mathcal{P}}_f & -\boldsymbol{\mathcal{P}}_f^T\underline{\hat{n}}_x\boldsymbol{\mathcal{P}}_f & \\ \boldsymbol{\mathcal{P}}_f^T\underline{Z}^-\boldsymbol{\mathcal{P}}_f & & \\ & \boldsymbol{\mathcal{P}}_f^T\underline{Z}^-\boldsymbol{\mathcal{P}}_f & \\ & & \boldsymbol{\mathcal{P}}_f^T\underline{Z}^-\boldsymbol{\mathcal{P}}_f\end{bmatrix}\begin{bmatrix}\underline{\Lambda}_x\\ \underline{\Lambda}_y\\ \underline{\Lambda}_z\end{bmatrix} \tag{6.7}$$

Of course, the basis used in HDGM does not represent $\vec{\Lambda}^t$ using Cartesian components. Instead it uses 2 vectors tangential to the surface of the face at each basis point. This is necessary to reduce the DOF and improve performance of the algorithm. However this results in $\boldsymbol{\mathcal{L}}_{V_n}$ requiring an additional term to account for converting basis vectors to Cartesian components.

To construct this operator we first examine the process to convert normalized basis vectors $\vec{a}_1$ and $\vec{a}_2$ to Cartesian components at a single point. We show the problem setup for this in Figure 6.1. From this we use the Cartesian components of basis vectors $\vec{a}_1^{\,i}$ and $\vec{a}_2^{\,i}$ for face point i to construct matrix $\boldsymbol{\mathcal{A}}^i$ as shown in (6.8). This matrix can be used to convert coefficients of the basis vectors to Cartesian components as shown in (6.9).

$$\boldsymbol{\mathcal{A}}^i = \begin{bmatrix} a^i_{1,x} & a^i_{2,x}\\ a^i_{1,y} & a^i_{2,y}\\ a^i_{1,z} & a^i_{2,z}\end{bmatrix} \tag{6.8}$$

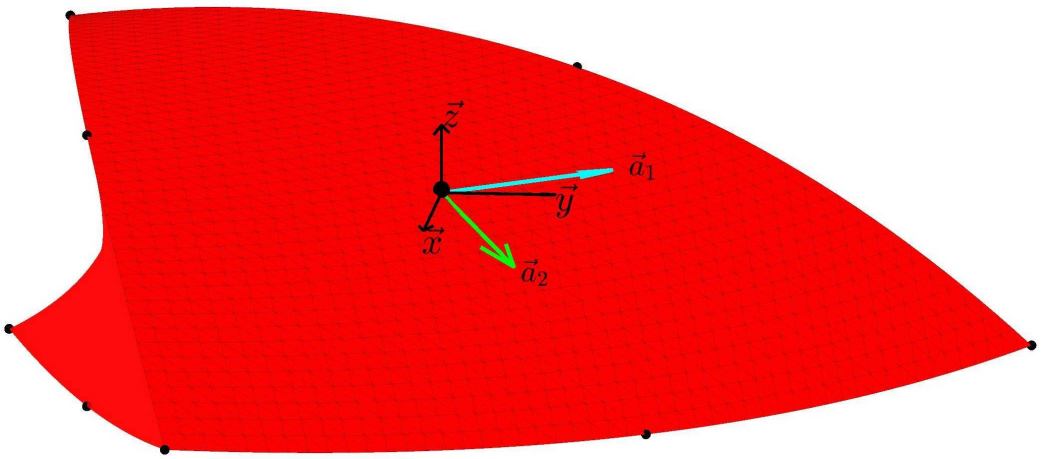


Fig. 6.1: HDGM basis vectors and Cartesian unit vectors at a point.

$$\begin{bmatrix} \Lambda_x^i \\ \Lambda_y^i \\ \Lambda_z^i \end{bmatrix} = \boldsymbol{\mathcal{A}}^i \begin{bmatrix} a_1^i \\ a_2^i \end{bmatrix} \tag{6.9}$$

When using elements with linear geometry $\boldsymbol{\mathcal{A}}^i$ is the same for all basis points and does not need to be re-computed. However, for high-order geometric elements vectors $\vec{a}_1^{\,i}$ and $\vec{a}_2^{\,i}$ may change at each basis point due to the curvature of the surface. We use $\boldsymbol{\mathcal{A}}^i$ at each face point to create matrices $\boldsymbol{\mathcal{A}}_x$, $\boldsymbol{\mathcal{A}}_y$, and $\boldsymbol{\mathcal{A}}_z$ which are used to recover the given Cartesian component. We show construction of $\boldsymbol{\mathcal{A}}_x$ in (6.10) and note $\boldsymbol{\mathcal{A}}_y$ and $\boldsymbol{\mathcal{A}}_z$ are created similarily.

$$\begin{bmatrix} \Lambda_x^1 \\ \Lambda_x^2 \\ \vdots \\ \Lambda_x^{Nfp} \end{bmatrix} = \boldsymbol{\mathcal{A}}_x \underline{\Lambda}^t = \begin{bmatrix} a_{1,x}^1 & a_{2,x}^1 & & & & \\ & & a_{1,x}^2 & a_{2,x}^2 & & \\ & & & & \ddots & \\ & & & & a_{1,x}^{Nfp} & a_{2,x}^{Nfp} \end{bmatrix} \begin{bmatrix} \vec{a}_1^1 \\ \vec{a}_2^1 \\ \vdots \\ \vec{a}_1^{Nfp} \\ \vec{a}_2^{Nfp} \end{bmatrix} \tag{6.10}$$

We use this and obtain the final equation for $\boldsymbol{\mathcal{L}}_{V_n}$ below.

$$\boldsymbol{\mathcal{L}}_{V_n} = \sum_{Face=1}^{4} \begin{bmatrix} & \boldsymbol{\mathcal{P}}_f^T \underline{\hat{n}}_z \boldsymbol{\mathcal{A}}_x \boldsymbol{\mathcal{P}}_f & -\boldsymbol{\mathcal{P}}_f^T \underline{\hat{n}}_y \boldsymbol{\mathcal{A}}_x \boldsymbol{\mathcal{P}}_f \\ -\boldsymbol{\mathcal{P}}_f^T \underline{\hat{n}}_z \boldsymbol{\mathcal{A}}_y \boldsymbol{\mathcal{P}}_f & & \boldsymbol{\mathcal{P}}_f^T \underline{\hat{n}}_x \boldsymbol{\mathcal{A}}_y \boldsymbol{\mathcal{P}}_f \\ \boldsymbol{\mathcal{P}}_f^T \underline{\hat{n}}_y \boldsymbol{\mathcal{A}}_z \boldsymbol{\mathcal{P}}_f & -\boldsymbol{\mathcal{P}}_f^T \underline{\hat{n}}_x \boldsymbol{\mathcal{A}}_z \boldsymbol{\mathcal{P}}_f & \\ \boldsymbol{\mathcal{P}}_f^T \underline{Z}^- \boldsymbol{\mathcal{A}}_x \boldsymbol{\mathcal{P}}_f & & \\ & \boldsymbol{\mathcal{P}}_f^T \underline{Z}^- \boldsymbol{\mathcal{A}}_y \boldsymbol{\mathcal{P}}_f & \\ & & \boldsymbol{\mathcal{P}}_f^T \underline{Z}^- \boldsymbol{\mathcal{A}}_z \boldsymbol{\mathcal{P}}_f \end{bmatrix} \tag{6.11}$$

Similar to $\boldsymbol{\mathcal{K}}_{V_n}$ this matrix can be efficiently implemented using row scaling, which applies to $\underline{\hat{n}}_i$, $\underline{Z}^-$, and $\boldsymbol{\mathcal{A}}_i$. Overall, this operator requires 18 matrix-matrix products, each of a matrix size Nfp by Nqp times a matrix size Nqp by Nfp for each face of each element, as in practice $\boldsymbol{\mathcal{A}}$ is separated into two matrices.

6.1.3 Construction of $\boldsymbol{\mathcal{R}}_{V_n}$

$\boldsymbol{\mathcal{R}}_{V_n}$ is used to compute the integral term involving the incident fields. This term is added to the right hand side of the global system of equations and approximates the volume integral on the right hand side of (5.6).

$$\boldsymbol{\mathcal{R}}_{V_n} \begin{bmatrix} \underline{E}^{inc} \\ \underline{H}^{inc} \end{bmatrix} \approx \begin{bmatrix} -\int\limits_{V_n} \boldsymbol{\psi}\ j\omega(\varepsilon - \varepsilon_{bkg}) \vec{E}^{inc} dV_n \\ -\int\limits_{V_n} \boldsymbol{\psi}\ j\omega(\mu - \mu_{bkg}) \vec{H}^{inc} dV_n \end{bmatrix} \tag{6.12}$$

$$\boldsymbol{\mathcal{R}}_{V_n}\begin{bmatrix}\underline{E}^{inc}\\ \underline{H}^{inc}\end{bmatrix} = -j\omega\begin{bmatrix}(\varepsilon-\varepsilon_{bkg})\boldsymbol{\mathcal{M}} & & & & & \\ & (\varepsilon-\varepsilon_{bkg})\boldsymbol{\mathcal{M}} & & & & \\ & & (\varepsilon-\varepsilon_{bkg})\boldsymbol{\mathcal{M}} & & & \\ & & & (\mu-\mu_{bkg})\boldsymbol{\mathcal{M}} & & \\ & & & & (\mu-\mu_{bkg})\boldsymbol{\mathcal{M}} & \\ & & & & & (\mu-\mu_{bkg})\boldsymbol{\mathcal{M}}\end{bmatrix}\begin{bmatrix}\underline{E}_x^{inc}\\ \underline{E}_y^{inc}\\ \underline{E}_z^{inc}\\ \underline{H}_x^{inc}\\ \underline{H}_y^{inc}\\ \underline{H}_z^{inc}\end{bmatrix} \tag{6.13}$$

This matrix requires almost no computational resources to compute as we can simply scale the mass matrix.

6.1.4 Global Operator Form

Typical use of HDGM as a forward solver wouldn't require a formulation using global operators. However, global operators simplify parts of the CSI derivation in Chapter 7. We briefly introduce compact operators below which are used to create a compact equation in (6.17). We begin with operator $\boldsymbol{\mathcal{K}}_{\Omega}^{-1}$ which is a block diagonal matrix consisting of $\boldsymbol{\mathcal{K}}_{V_n}^{-1}$ for each element in domain Ω.

$$\boldsymbol{\mathcal{K}}_{\Omega}^{-1} = \begin{bmatrix}\boldsymbol{\mathcal{K}}_1^{-1} & & & \\ & \boldsymbol{\mathcal{K}}_2^{-1} & & \\ & & \ddots & \\ & & & \boldsymbol{\mathcal{K}}_{Ne}^{-1}\end{bmatrix} \tag{6.14}$$

Next, $\boldsymbol{\mathcal{R}}_\Omega$ is a block diagonal matrix of $\boldsymbol{\mathcal{R}}_{V_n}$ for each element.

$$\boldsymbol{\mathcal{R}}_\Omega = \begin{bmatrix} \boldsymbol{\mathcal{R}}_1 & & & \\ & \boldsymbol{\mathcal{R}}_2 & & \\ & & \ddots & \\ & & & \boldsymbol{\mathcal{R}}_{Ne} \end{bmatrix} \tag{6.15}$$

Lastly, $\boldsymbol{\mathcal{L}}_\Omega$ is the global operator for all local systems. It is the effect on electric and magnetic fields in Ω due to fields at the basis vectors. It is constructed using $\boldsymbol{\mathcal{L}}_{V_n}^{F_i}$ where F_i refers to the face, as we require integration along all faces to compute the fields within the element. Each element corresponds to a block row of the matrix $\boldsymbol{\mathcal{L}}_\Omega$. Each block row contains four block column terms; one for each face. The column depends on the index into global vector $\underline{\Lambda}^t$ that corresponds to the trace.

$$\boldsymbol{\mathcal{L}}_\Omega = \begin{bmatrix} \boldsymbol{\mathcal{L}}_1^1 & \boldsymbol{\mathcal{L}}_1^2 & & & \boldsymbol{\mathcal{L}}_1^3 & & & \boldsymbol{\mathcal{L}}_1^4 & \\ & \boldsymbol{\mathcal{L}}_2^1 & \boldsymbol{\mathcal{L}}_2^2 & \boldsymbol{\mathcal{L}}_2^3 & & & \boldsymbol{\mathcal{L}}_2^4 & & \\ & & \ddots & & & & & & \\ & & & \ddots & & & & & \\ & & & & \ddots & & & & \\ & & & \boldsymbol{\mathcal{L}}_{Ne}^1 & \boldsymbol{\mathcal{L}}_{Ne}^2 & & & \boldsymbol{\mathcal{L}}_{Ne}^3 & \boldsymbol{\mathcal{L}}_{Ne}^4 \end{bmatrix} \tag{6.16}$$

We use these matrices to write the solution to all local systems in Ω using the global form shown below:

$$\begin{bmatrix} \underline{E}_1^{sct} \\ \underline{H}_1^{sct} \\ \vdots \\ \underline{E}_{Ne}^{sct} \\ \underline{H}_{Ne}^{sct} \end{bmatrix} = \boldsymbol{\mathcal{K}}_\Omega^{-1}\boldsymbol{\mathcal{R}}_\Omega \begin{bmatrix} \underline{E}_1^{inc} \\ \underline{H}_1^{inc} \\ \vdots \\ \underline{E}_{Ne}^{inc} \\ \underline{H}_{Ne}^{inc} \end{bmatrix} + \boldsymbol{\mathcal{K}}_\Omega^{-1}\boldsymbol{\mathcal{L}}_\Omega \begin{bmatrix} \underline{\Lambda}_1^t \\ \underline{\Lambda}_2^t \\ \vdots \\ \underline{\Lambda}_{Nf}^t \end{bmatrix} \tag{6.17}$$

6.2 Implementation of the Global System

The global system is constructed by enforcing continuity of fields between elements, and using boundary conditions where necessary. This section examines the construction of the global matrix $\boldsymbol{\mathcal{A}}_\Omega$ and global right hand side vector $\underline{b}_\Omega$ which are required to compute the global basis coefficient vector $\underline{\Lambda}_\Omega$. We do not actually compute the inverse of $\boldsymbol{\mathcal{A}}_\Omega$ as it is computationally expensive but instead use this notation to represent factorization of the matrix.

$$\underline{\Lambda}_\Omega = \boldsymbol{\mathcal{A}}_\Omega^{-1} \underline{b}_\Omega \tag{6.18}$$

We begin by making an observation of the local system which is repeated below.

$$f^{\pm}(\underline{\Lambda}^S_{V^\pm}) = \begin{bmatrix} \underline{E}^{\pm,sct} \\ \underline{H}^{\pm,sct} \end{bmatrix} = \boldsymbol{\mathcal{K}}^{-1}_{V^\pm} \boldsymbol{\mathcal{L}}_{V^\pm} \underline{\Lambda}^S_{V^\pm} + \boldsymbol{\mathcal{K}}^{-1}_{V^\pm} \boldsymbol{\mathcal{R}}_{V^\pm} \begin{bmatrix} \underline{E}^{\pm,inc} \\ \underline{H}^{\pm,inc} \end{bmatrix} \quad \text{(5.14 revisited)}$$

This equation is based upon two different terms. $\underline{\Lambda}^S_{V^\pm}$ which consists of values in the global set of unknowns, and $\underline{E}/\underline{H}^{\pm,inc}$ which are known prior to solving the global system. We must separate these terms in order to implement the system of equations. We begin with (5.13), the equation enforced between neighbouring elements derived in Section 5.3, which is repeated below.

$$\begin{aligned} &\hat{n} \times \boldsymbol{\mathcal{S}}_E f^-(\underline{\Lambda}^S_{V^-}) - Z^- \hat{n} \times \hat{n} \times \boldsymbol{\mathcal{S}}_H f^-(\underline{\Lambda}^S_{V^-}) - \hat{n} \times \boldsymbol{\mathcal{S}}_E f^+(\underline{\Lambda}^S_{V^+}) \\ &- Z^+ \hat{n} \times \hat{n} \times \boldsymbol{\mathcal{S}}_H f^+(\underline{\Lambda}^S_{V^+}) + (Z^- + Z^+) \hat{n} \times \hat{n} \times \underline{\Lambda}^t = 0 \end{aligned} \quad \text{(5.13 revisited)}$$

We separate terms from the local system and move terms involving incident fields to the

right hand side in (6.19).

$$\begin{aligned}
&\big(\hat{n}\times\boldsymbol{\mathcal{S}}_E - Z^-\hat{n}\times\hat{n}\times\boldsymbol{\mathcal{S}}_H\big)\boldsymbol{\mathcal{K}}_{V^-}^{-1}\boldsymbol{\mathcal{L}}_{V^-}\underline{\Lambda}^S_{V^-} - \big(\hat{n}\times\boldsymbol{\mathcal{S}}_E + Z^+\hat{n}\times\hat{n}\times\boldsymbol{\mathcal{S}}_H\big)\boldsymbol{\mathcal{K}}_{V^+}^{-1}\boldsymbol{\mathcal{L}}_{V^+}\underline{\Lambda}^S_{V^+} \\
&\qquad + (Z^- + Z^+)\hat{n}\times\hat{n}\times\underline{\Lambda}^t \\
&= \big(-\hat{n}\times\boldsymbol{\mathcal{S}}_E + Z^-\hat{n}\times\hat{n}\times\boldsymbol{\mathcal{S}}_H\big)\boldsymbol{\mathcal{K}}_{V^-}^{-1}\boldsymbol{\mathcal{R}}_{V^-}\begin{bmatrix}\underline{E}^{-,inc}\\ \underline{H}^{-,inc}\end{bmatrix} \\
&\qquad + \big(\hat{n}\times\boldsymbol{\mathcal{S}}_E + Z^+\hat{n}\times\hat{n}\times\boldsymbol{\mathcal{S}}_H\big)\boldsymbol{\mathcal{K}}_{V^+}^{-1}\boldsymbol{\mathcal{R}}_{V^+}\begin{bmatrix}\underline{E}^{+,inc}\\ \underline{H}^{+,inc}\end{bmatrix}
\end{aligned} \tag{6.19}$$

Note that most of these terms will have previously been computed. Since the element on either side of the trace adds values to the global matrix for the trace, we require computation of $\pm\hat{n}\times\boldsymbol{\mathcal{S}}_E$ and $\pm Z^{\pm}\hat{n}\times\hat{n}\times\boldsymbol{\mathcal{S}}_H$ for each face of the given element. The following matrix below shows a general case of this operator where the physics and signs should be modified accordingly

$$\hat{n}\times\boldsymbol{\mathcal{S}}_E + Z^-\hat{n}\times\hat{n}\times\boldsymbol{\mathcal{S}}_H = \sum_{Face=1}^{4}\begin{bmatrix}A_{00} & \boldsymbol{\mathcal{O}}\\ \boldsymbol{\mathcal{O}} & A_{11}\end{bmatrix}\begin{bmatrix}\underline{E}\\ \underline{H}\end{bmatrix},$$

$$A_{00} = \begin{bmatrix} & -\boldsymbol{\mathcal{P}}_f^T\underline{\hat{n}}_z\boldsymbol{\mathcal{P}}_f & \boldsymbol{\mathcal{P}}_f^T\underline{\hat{n}}_y\boldsymbol{\mathcal{P}}_f\\ \boldsymbol{\mathcal{P}}_f^T\underline{\hat{n}}_z\boldsymbol{\mathcal{P}}_f & & -\boldsymbol{\mathcal{P}}_f^T\underline{\hat{n}}_x\boldsymbol{\mathcal{P}}_f\\ -\boldsymbol{\mathcal{P}}_f^T\underline{\hat{n}}_y\boldsymbol{\mathcal{P}}_f & \boldsymbol{\mathcal{P}}_f^T\underline{\hat{n}}_x\boldsymbol{\mathcal{P}}_f & \end{bmatrix},$$

$$A_{11} = \begin{bmatrix} -\boldsymbol{\mathcal{P}}_f^T\underline{Z}^-(\underline{\hat{n}}_y\underline{\hat{n}}_y + \underline{\hat{n}}_z\underline{\hat{n}}_z)\boldsymbol{\mathcal{P}}_f & \boldsymbol{\mathcal{P}}_f^T\underline{Z}^-\underline{\hat{n}}_x\underline{\hat{n}}_y\boldsymbol{\mathcal{P}}_f & \boldsymbol{\mathcal{P}}_f^T\underline{Z}^-\underline{\hat{n}}_x\underline{\hat{n}}_z\boldsymbol{\mathcal{P}}_f\\ \boldsymbol{\mathcal{P}}_f^T\underline{Z}^-\underline{\hat{n}}_x\underline{\hat{n}}_y\boldsymbol{\mathcal{P}}_f & -\boldsymbol{\mathcal{P}}_f^T\underline{Z}^-(\underline{\hat{n}}_x\underline{\hat{n}}_x + \underline{\hat{n}}_z\underline{\hat{n}}_z)\boldsymbol{\mathcal{P}}_f & \boldsymbol{\mathcal{P}}_f^T\underline{Z}^-\underline{\hat{n}}_y\underline{\hat{n}}_z\boldsymbol{\mathcal{P}}_f\\ \boldsymbol{\mathcal{P}}_f^T\underline{Z}^-\underline{\hat{n}}_x\underline{\hat{n}}_z\boldsymbol{\mathcal{P}}_f & \boldsymbol{\mathcal{P}}_f^T\underline{Z}^-\underline{\hat{n}}_y\underline{\hat{n}}_z\boldsymbol{\mathcal{P}}_f & -\boldsymbol{\mathcal{P}}_f^T\underline{Z}^-(\underline{\hat{n}}_z\underline{\hat{n}}_z + \underline{\hat{n}}_y\underline{\hat{n}}_y)\boldsymbol{\mathcal{P}}_f\end{bmatrix}$$

(6.20)

This operator is used on the right hand side of the global system of equations where matrix-vector multiplication is used, and terms are added to $\underline{b}_\Omega$. On the left hand side we multiply this by the local system to enforce field conditions as a function of the underlying basis values. Construction of this operator requires 9 matrix-matrix products of matrices size Np by Nqp and Nqp by Np for each face of each element. We also require $Z^{\pm}\hat{n}\times\hat{n}\times\underline{\Lambda}^t$ on the left hand side which can be computed almost identically to $Z^-\hat{n}\times\hat{n}\times\boldsymbol{\mathcal{S}}_H$, but operates on $\underline{\Lambda}_\Omega$. Construction of this operator requires 6 matrix-matrix products of matrices size Np by Nqp and Nqp by Np, but only once for each trace.

In addition to the cost of constructing these operators, we must evaluate the interaction between faces of each element. This is also computationally demanding as it requires 32 matrix-matrix products of size $2Nfp$ by $3Np$ times $3Np$ by $2Nfp$ for each element. This is because each element computes the interaction of each of its four faces with each face, and does so for both $\pm\hat{n}\times\boldsymbol{\mathcal{S}}_E$ and $\pm Z^{\pm}\hat{n}\times\hat{n}\times\boldsymbol{\mathcal{S}}_H$

Similar operations are used for boundary conditions. Below each boundary condition is re-written with isolated unknown and incident field terms. The equation for ABC's is shown in (6.21), PEC's in (6.22), and IBC's in (6.23).

$$\begin{aligned}&\big(\hat{n}\times\boldsymbol{\mathcal{S}}_E - Z^-\hat{n}\times\hat{n}\times\boldsymbol{\mathcal{S}}_H\big)\boldsymbol{\mathcal{K}}_{V^-}^{-1}\boldsymbol{\mathcal{L}}_{V^-}\underline{\Lambda}_{V^-}^{S} + 2Z^-\hat{n}\times\hat{n}\times\underline{\Lambda}^t\\ &= \big(-\hat{n}\times\boldsymbol{\mathcal{S}}_E + Z^-\hat{n}\times\hat{n}\times\boldsymbol{\mathcal{S}}_H\big)\boldsymbol{\mathcal{K}}_{V^-}^{-1}\boldsymbol{\mathcal{R}}_{V^-}\begin{bmatrix}\underline{E}^{-,inc}\\ \underline{H}^{-,inc}\end{bmatrix}\end{aligned} \tag{6.21}$$

$$\begin{aligned}&\big(\hat{n}\times\boldsymbol{\mathcal{S}}_E - Z^-\hat{n}\times\hat{n}\times\boldsymbol{\mathcal{S}}_H\big)\boldsymbol{\mathcal{K}}_{V^-}^{-1}\boldsymbol{\mathcal{L}}_{V^-}\underline{\Lambda}_{V^-}^{S} + Z^-\hat{n}\times\hat{n}\times\underline{\Lambda}^t\\ &= \big(-\hat{n}\times\boldsymbol{\mathcal{S}}_E + Z^-\hat{n}\times\hat{n}\times\boldsymbol{\mathcal{S}}_H\big)\boldsymbol{\mathcal{K}}_{V^-}^{-1}\boldsymbol{\mathcal{R}}_{V^-}\begin{bmatrix}\underline{E}^{-,inc}\\ \underline{H}^{-,inc}\end{bmatrix} - \hat{n}\times\underline{E}^{-,inc}\end{aligned} \tag{6.22}$$

$$\begin{aligned}
&\big(-\hat{n}\times\boldsymbol{\mathcal{S}}_E + Z^-\hat{n}\times\hat{n}\times\boldsymbol{\mathcal{S}}_H\big)\boldsymbol{\mathcal{K}}_{V^-}^{-1}\boldsymbol{\mathcal{L}}_{V^-}\underline{\Lambda}_{V^-}^{S} - (Z^- + Z_s)\hat{n}\times\hat{n}\times\underline{\Lambda}^t \\
&= \big(\hat{n}\times\boldsymbol{\mathcal{S}}_E - Z^-\hat{n}\times\hat{n}\times\boldsymbol{\mathcal{S}}_H\big)\boldsymbol{\mathcal{K}}_{V^-}^{-1}\boldsymbol{\mathcal{R}}_{V^-}\begin{bmatrix}\underline{E}^{-,inc}\\ \underline{H}^{-,inc}\end{bmatrix} + Z_s\hat{n}\times\hat{n}\times\underline{H}^{-,inc} + \hat{n}\times\underline{E}^{-,inc}
\end{aligned} \tag{6.23}$$

These matrices can be constructed in a manner similar to (6.20). Note that IBC and PEC equations involve incident fields terms which are not multiplied by $\boldsymbol{\mathcal{K}}_{V_n}$ or $\boldsymbol{\mathcal{R}}_{V_n}$ as they are due to total field enforcements. These are also added to $\underline{b}_\Omega$.

6.3 Parallelism

Due to the large computational demand of HDGM, a parallel implementation which could take advantage of multiple central processing units was required. This section examines a number of details used by this implementation, beginning with domain partitioning. Domain partitioning is used to separate the mesh into a number of partitions equal to the number of processors being used. The Zoltan software library is used to partition the mesh, using orthogonal recursive bisection [33]. Each partition contains an approximately equal number of elements in order to balance computational work. Since terms in the global system couple neighbouring elements, each partition requires values from adjacent elements to the boundary of the partition. An MPI parallel implementation was used to communicate between processors [29]. Since each processor contains only a subset of the mesh, this implementation also reduced the memory required per processor.

Once the mesh is partitioned, each processor computes $\boldsymbol{\mathcal{K}}_{V_n}$, $\boldsymbol{\mathcal{L}}_{V_n}$, and $\boldsymbol{\mathcal{R}}_{V_n}$ for each element its partition. This allows processors to simultaneously evaluate terms in the global system of equations. Each processor computes terms for elements within its partition, which is possible because a global ordering of unknowns based on element numbering can be created and communicated between processors.

Once the factorization of $\boldsymbol{\mathcal{A}}_\Omega$ is computed, each processor computes terms in the global right hand side, $\underline{b}_\Omega$, due to elements in its partition. Using the factorization, the processors compute $\underline{\Lambda}^t_\Omega$, the global unknowns. Each processor obtains values for traces in within its partition, and uses the previously computed local system to evaluate the electric and magnetic field components where required.

7

Inverse Problems

So far, this thesis has focused on the design and implementation of the HDGM forward solver. An additional goal of this research was to implement and use the solver for Microwave Imaging (MWI) at the electromagnetic imaging lab. This chapter examines a few key pieces of the CSI algorithm, and derives the operators required for the CSI algorithm to take advantage of features HDGM offers. While this chapter only discusses CSI, the solver has also has been used for other imaging techniques such as neural-network-based imaging [34]. Many imaging techniques benefit from the ability to compute a forward model, and can use HDGM.

CSI is a well known algorithm for imaging [35]. It has previously been studied for a wide range of applications including biomedical imaging [36–38], geophysics [39], agriculture [40, 41], and more. In the context of MWI, CSI aims to reconstruct the permittivity and/or permeability of the object of interest. It does this by iteratively minimizing a cost

functional which consists of two terms. The first term accounts for field discrepancies between experimentally collected data surrounding the object and synthetically generated data based on an estimate of the constitutive parameters. The second term is used as a Maxwellian regularizer to ensure the reconstruction properly enforces Maxwell's equations [42]. Of interest to us is that the CSI algorithm requires two types of forward solves; one which inputs a term based on the estimate of the physics and outputs the scattered fields due to those estimates, and one which backpropagates error by computing the effect on constitutives due to mismatched field values at the antenna locations. The forward solve to compute scattering based on known constitutives is exactly what has been described for HDGM so far. We now focus on the derivation and theory required in order to compute the response of constitutives from mismatched fields.

Since this chapter focuses on using the HDGM forward solver in CSI, and does not go into detail on the CSI algorithm, readers are referred to the following for DGM-CSI and HDGM-CSI details [5, 21, 43–46]. In Section 7.1 we derive the operator required for CSI, and in Section 7.2 the modifications required for HDGM are shown. Results of CSI using the HDGM method are compared in Chapter 8.

7.1 Derivation of Required HDGM Operator

This section aims to derive the operator required to use HDGM in CSI. We begin with equation (7.1) which examines the effect on fields due to a small change of sources. To do this we use the Gateaux derivative of search direction $\underline{h}$ as step size ϵ approaches 0 [5, 42]. $\underline{E}$ are the measured field values on measurement surface S, $\boldsymbol{\mathcal{L}}(\underline{x})$ is the forward operator which computes the fields due to source vector $\underline{x}$, and α is a scalar coefficient.

$$\lim_{\epsilon \to 0} \alpha \frac{\|\underline{E} - \boldsymbol{\mathcal{L}}(\underline{w} + \epsilon \underline{h})\|_S^2 - \|\underline{E} - \boldsymbol{\mathcal{L}}(\underline{w})\|_S^2}{\epsilon} \tag{7.1}$$

The terms from the first norm in (7.1) are expanded using the identity in (7.2) to give (7.3) [5].

$$\|A + B\|^2 =< A + B, A + B >= \|A\|^2 + \|B\|^2 + 2Real < A, B > \tag{7.2}$$

$$\lim_{\epsilon \to 0} \alpha \frac{\|\underline{E} - \boldsymbol{\mathcal{L}}(\underline{w})\|_S^2 - 2\epsilon Real < \underline{E} - \boldsymbol{\mathcal{L}}(\underline{w}), \boldsymbol{\mathcal{L}}(\underline{h}) > -\epsilon^2 \|\boldsymbol{\mathcal{L}}(\underline{h})\|_S^2 - \|\underline{E} - \boldsymbol{\mathcal{L}}(\underline{w})\|_S^2}{\epsilon} \tag{7.3}$$

Note that all terms cancel except $-2\alpha Real < \underline{E} - \boldsymbol{\mathcal{L}}(\underline{w}), \boldsymbol{\mathcal{L}}(\underline{h}) >$. The search direction is isolated using the adjoint operator of $\boldsymbol{\mathcal{L}}$ which is precisely the operator CSI requires which we will derive in the following section:

$$\underline{h} = -2\alpha \boldsymbol{\mathcal{L}}^{adjoint}(\underline{E} - \boldsymbol{\mathcal{L}}(\underline{w})) \tag{7.4}$$

7.2 Formulation of Required HDGM Operator

We now know CSI requires the adjoint operator of $\boldsymbol{\mathcal{L}}(\underline{w})$. In a DGM formulation, this operator is easy to compute as the matrix system is set up to do exactly that. Since we require the adjoint of this operator we simply compute the hermitian of the operator. The following matrix system is used by DGM which maps $\underline{w}$ directly to $\underline{E}$:

$$\begin{aligned} \underline{E} &= \boldsymbol{\mathcal{L}}\underline{w} \\ \underline{w} &= \boldsymbol{\mathcal{L}}^H \underline{E} \end{aligned} \tag{7.5}$$

Some additional details related to non-uniform element size is omitted for simplicity but can be found in [5].

Since HDGM uses a two-stage solve procedure the formulation is not as simple. It requires both the solution to the global system of equations as well as local post-processing to compute the fields. We begin by formulating equations which will be equivalent to $\boldsymbol{\mathcal{L}}$ for HDGM. This requires the global, local post-processing operator from (6.17) as well as our global matrix system from (6.18). We modify (6.17) below by replacing the global vector of unknown basis coefficients with the solution to the global system of equations, and changing the incident fields to general sources $\underline{w}$.

$$\begin{bmatrix} \underline{\vec{E}}_1^{sct} \\ \underline{\vec{H}}_1^{sct} \\ \vdots \\ \underline{\vec{E}}_{Ne}^{sct} \\ \underline{\vec{H}}_{Ne}^{sct} \end{bmatrix} = \boldsymbol{\mathcal{K}}_\Omega^{-1}\boldsymbol{\mathcal{R}}_\Omega\underline{w} + \boldsymbol{\mathcal{K}}_\Omega^{-1}\boldsymbol{\mathcal{L}}_\Omega\big(\boldsymbol{\mathcal{A}}_\Omega^{-1}\boldsymbol{\mathcal{R}}_\Omega\underline{w}\big) \tag{7.6}$$

From this we compute the forward operator $\boldsymbol{\mathcal{L}}$, shown in (7.7), such that the equalities used for DGM in (7.5) hold for HDGM.

$$\boldsymbol{\mathcal{L}} = \boldsymbol{\mathcal{K}}_{\Omega}^{-1}\boldsymbol{\mathcal{R}}_{\Omega} + \boldsymbol{\mathcal{K}}_{\Omega}^{-1}\boldsymbol{\mathcal{L}}_{\Omega}\left(\boldsymbol{\mathcal{A}}_{\Omega}^{-1}\boldsymbol{\mathcal{R}}_{\Omega}\right) \tag{7.7}$$

In order to compute the hermitian operator we require the identities in (7.8).

$$\begin{aligned} (A+B)^H &= A^H + B^H \\ (AB)^H &= B^H A^H \end{aligned} \tag{7.8}$$

We compute the hermitian as shown in (7.9). We verify correctness of this formulation in the results, where we see HDGM and DGM achieve the exact same values to expected precision.

$$\boldsymbol{\mathcal{L}}^H = \boldsymbol{\mathcal{R}}_{\Omega}^H(\boldsymbol{\mathcal{K}}_{\Omega}^{-1})^H + \boldsymbol{\mathcal{R}}_{\Omega}^H(\boldsymbol{\mathcal{A}}_{\Omega}^{-1})^H\boldsymbol{\mathcal{L}}_{\Omega}^H(\boldsymbol{\mathcal{K}}_{\Omega}^{-1})^H \tag{7.9}$$

8

Results and Validation

In this section a number of examples are used in order to validate and benchmark HDGM. Previous work has examined the use of DGM both as a forward solver and for inverse problems [5,21]. Since HDGM maintains the same advantages of DGM, this chapter focuses on the improved performance of HDGM over DGM. We begin in Section 8.1 where HDGM is validated using a problem setup which has an analytic scattered field; the scattering of a planewave from a PEC sphere. Next, in Section 8.2 we compare the performance of DGM, HDGM, and an existing first-order Finite Element Method (FEM) solver [42] in terms of computational time, memory requirements, and accuracy. Last, Section 8.3 examines the benefits of HDGM for inverse problems. As previously mentioned Gmsh [11] is used for mesh generation, PETSc [30] is used to construct the matrix systems, and MUMPS [31] is used to solve the global matrix.

8.1 Scattering from a PEC Sphere

We begin with a simple example to validate the HDGM implementation. The scattering of a planewave from a PEC sphere has an analytic solution which can be computed using a Mie series [47]. Figure 8.1 shows a subsection of the numerical computational domain used by HDGM to simulate this problem. The mesh consists of approximately 3000 elements, each using 2^{nd} order polynomial representations for both geometry and fields. Coloured blue is a PEC sphere with a radius of 0.5 [m], in yellow is an ABC surface with a radius of 3.5 [m], and element edges are shown in green. Each element uses a permittivity of $1 - j0$ as the assumed background is free space. This problem uses an electric planewave incident field traveling in $\hat{z}$ with polarization $\hat{x}$ and a frequency of 300 MHz, resulting in a 1 [m] wavelength.

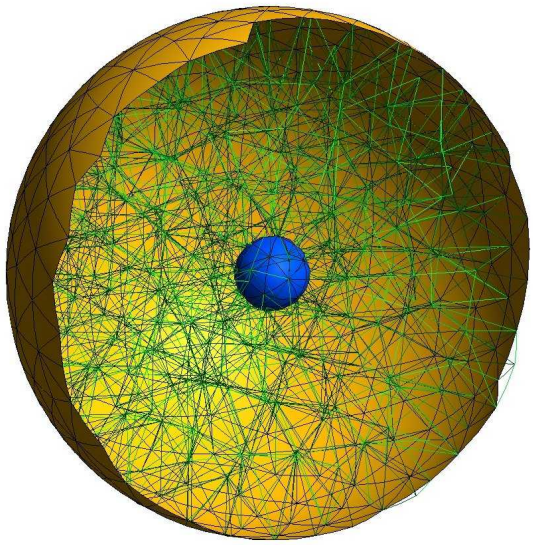
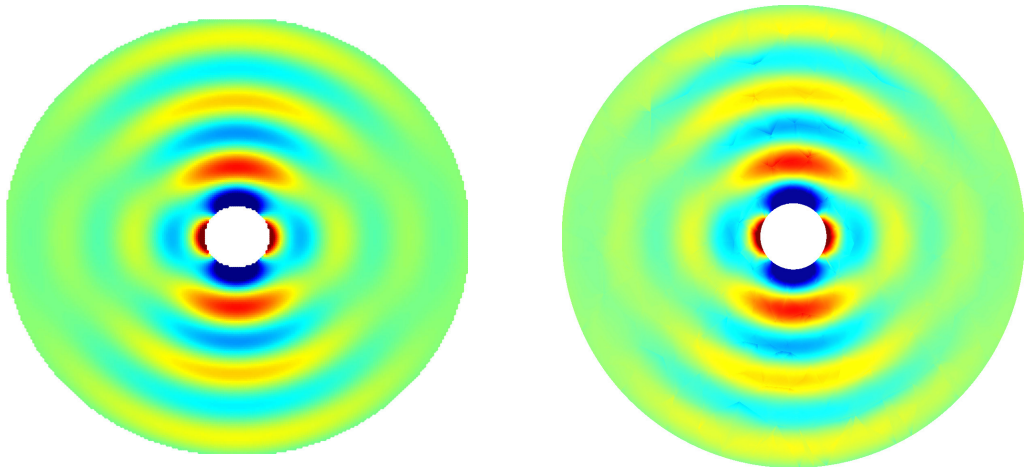


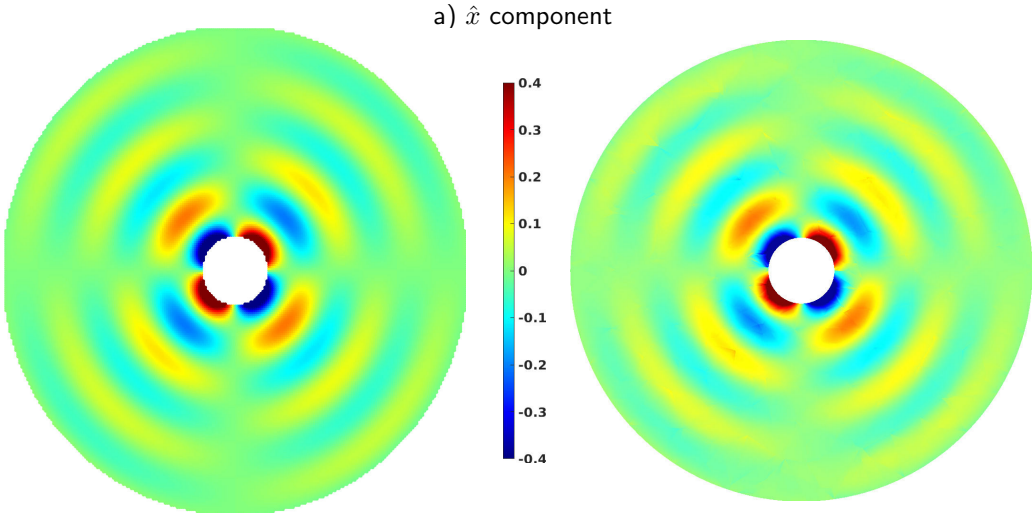
Fig. 8.1: Computational domain for the HDGM problem setup.

8.1.1 Qualitative Analysis

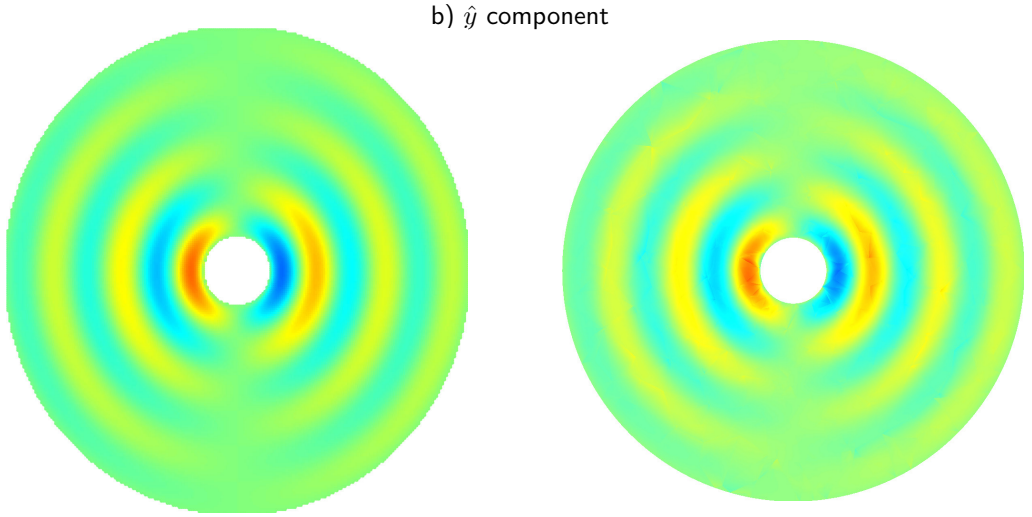
Figures 8.2 through 8.7 examine the real and imaginary components of the scattered electric field. Each figure displays the analytic solution on the left and the HDGM solution on the right, for each electric field vector component. Figures 8.2 and 8.3 show the XY plane, 8.4 and 8.5 show the XZ plane, and 8.6 and 8.7 show the YZ plane.



a) $\hat{x}$ component

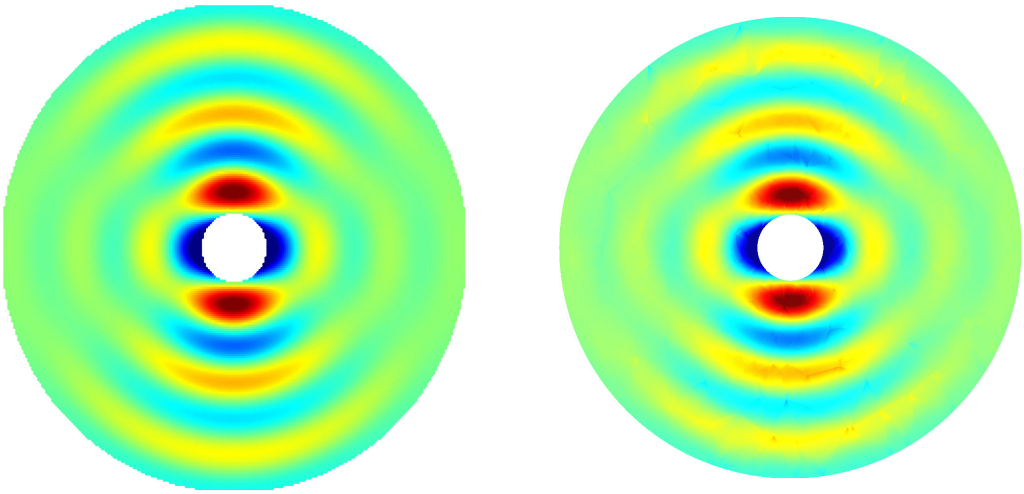


b) $\hat{y}$ component

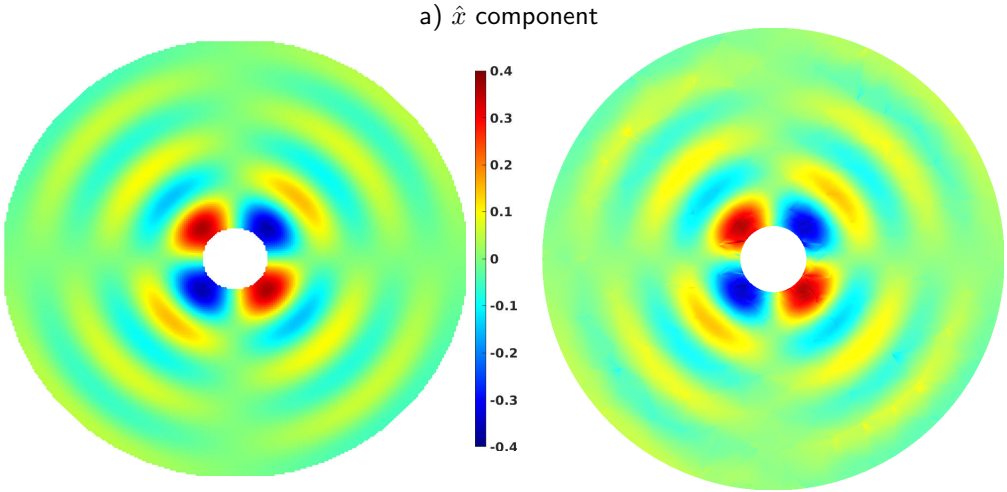


c) $\hat{z}$ component

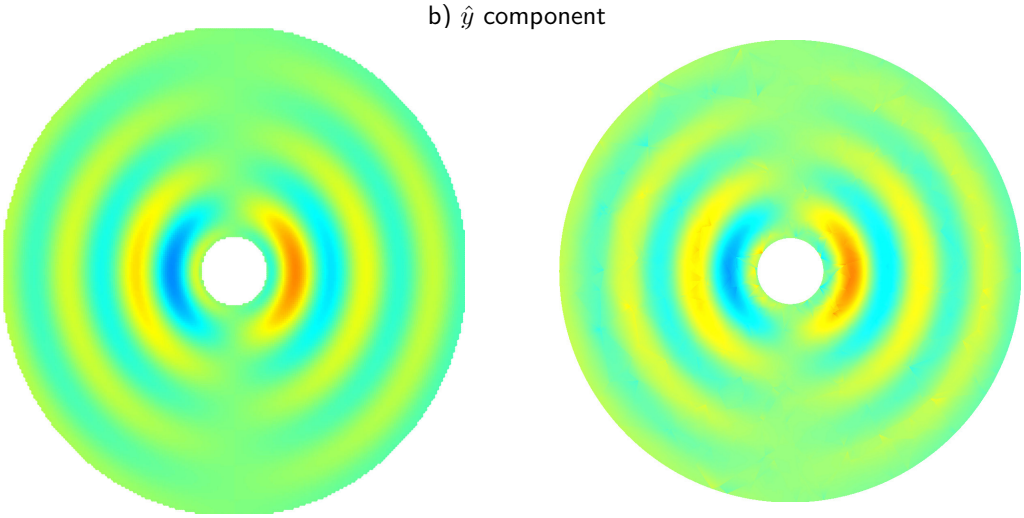
Fig. 8.2: Real component of the scattered electric field in the XY plane.



a) $\hat{x}$ component

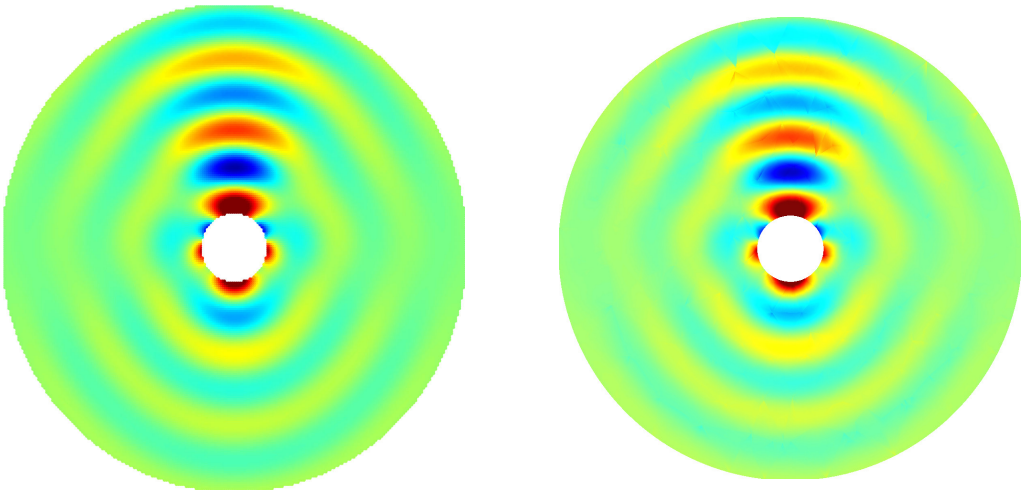


b) $\hat{y}$ component

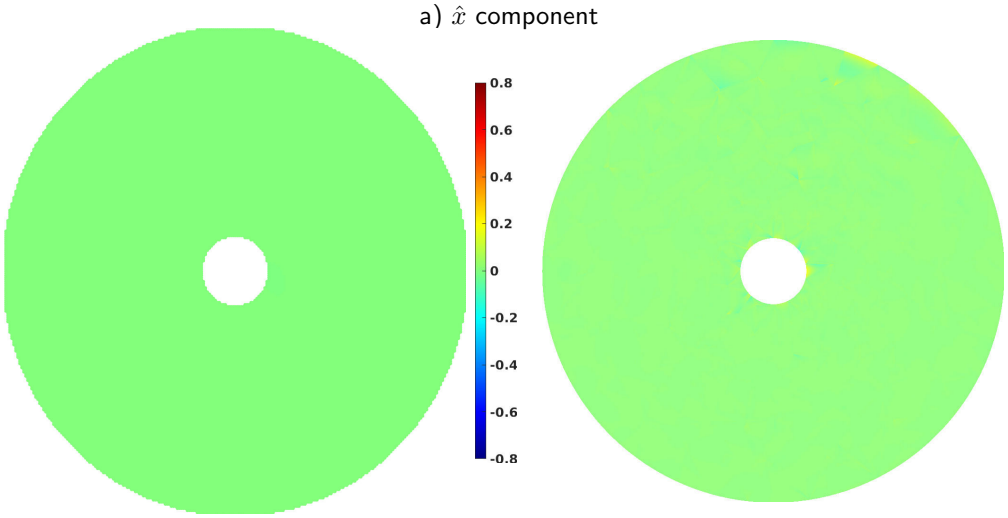


c) $\hat{z}$ component

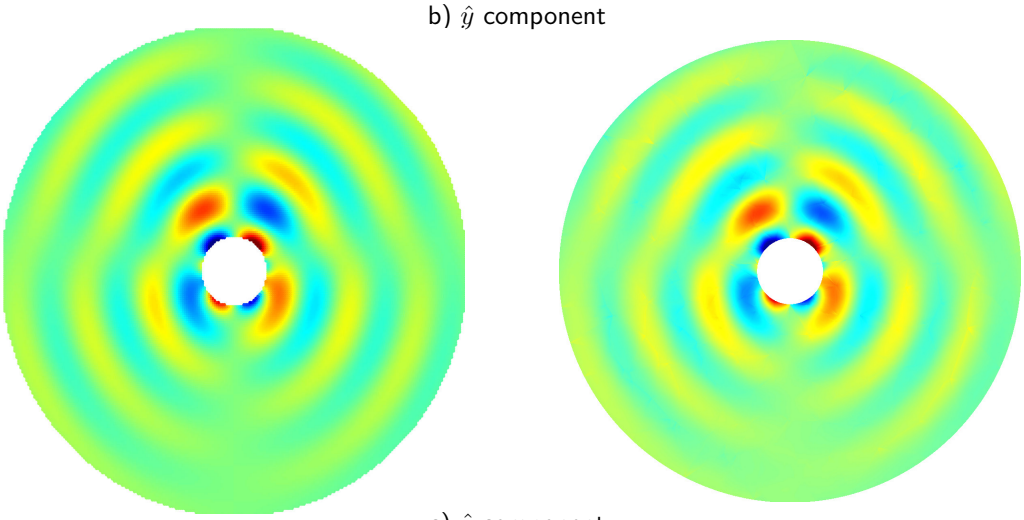
Fig. 8.3: Imaginary component of the scattered electric field in the XY plane.



a) $\hat{x}$ component

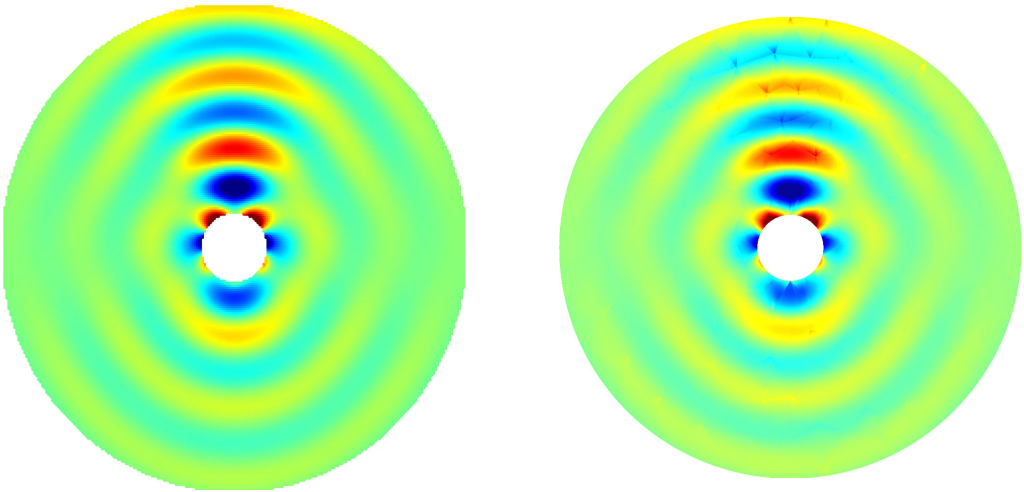


b) $\hat{y}$ component

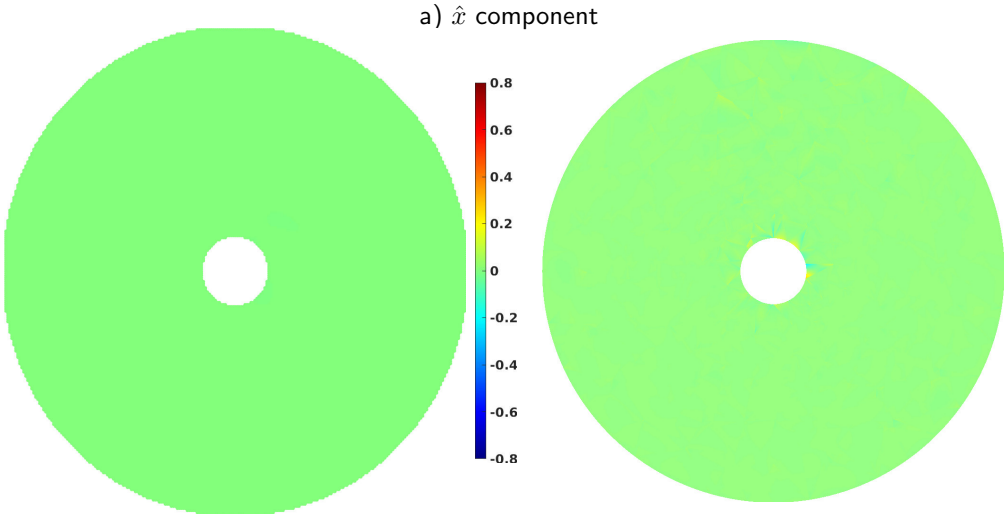


c) $\hat{z}$ component

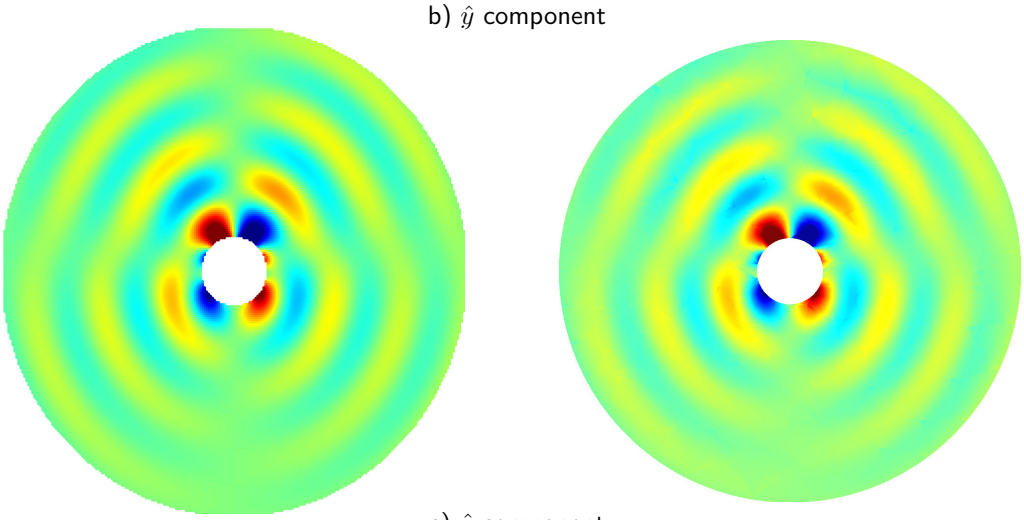
Fig. 8.4: Real component of the scattered electric field in the XZ plane.



a) $\hat{x}$ component

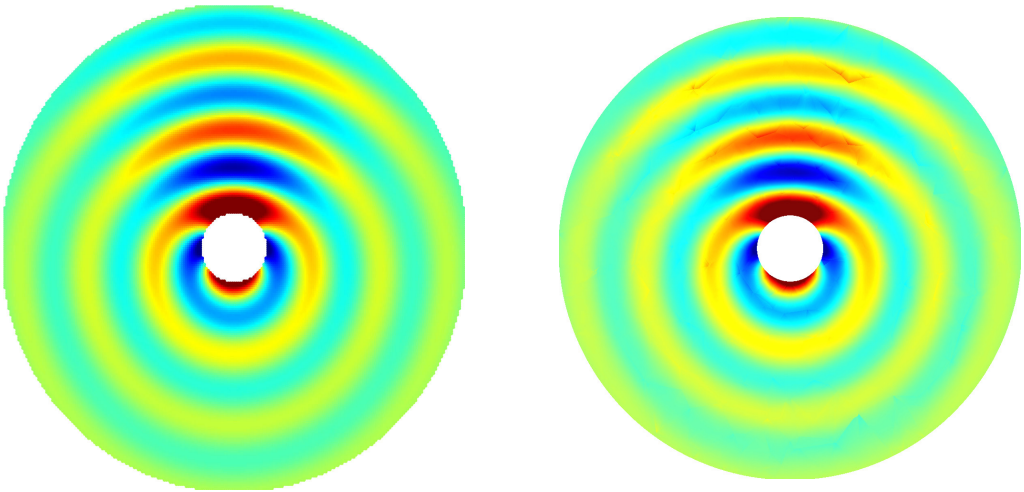


b) $\hat{y}$ component

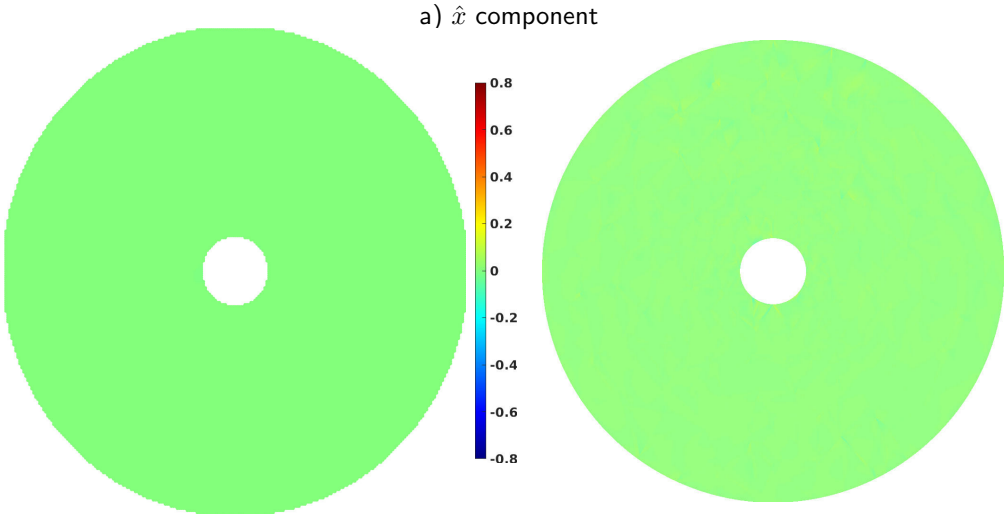


c) $\hat{z}$ component

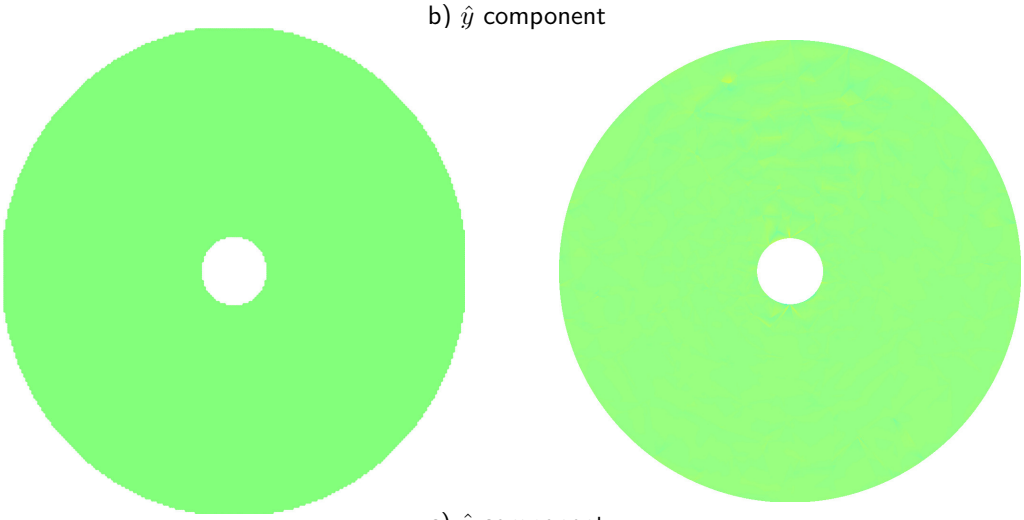
Fig. 8.5: Imaginary component of the scattered electric field in the XZ plane.



a) $\hat{x}$ component

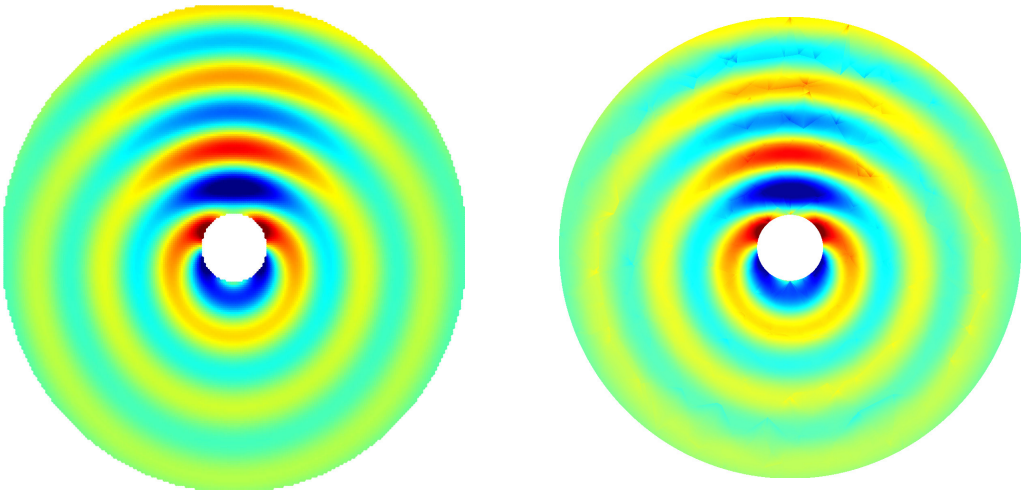


b) $\hat{y}$ component

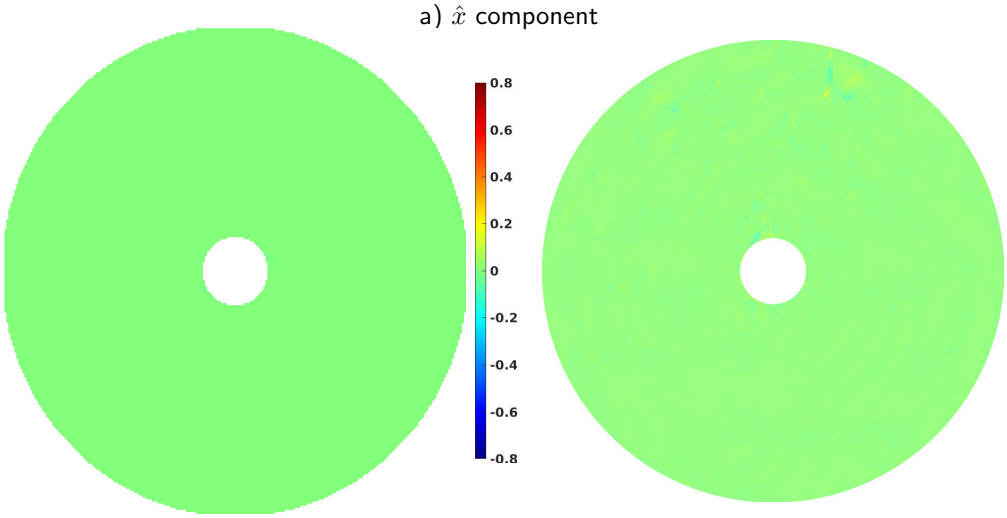


c) $\hat{z}$ component

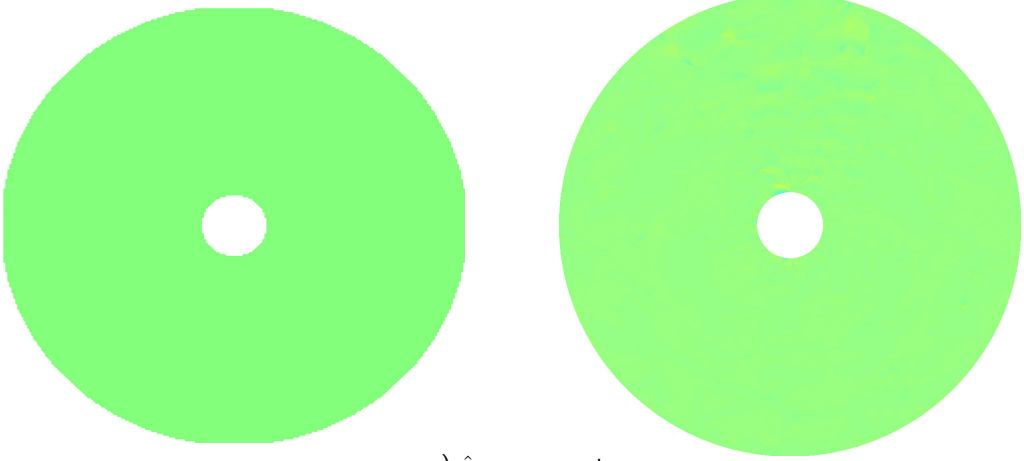
Fig. 8.6: Real component of the scattered electric field in the YZ plane.



a) $\hat{x}$ component



b) $\hat{y}$ component



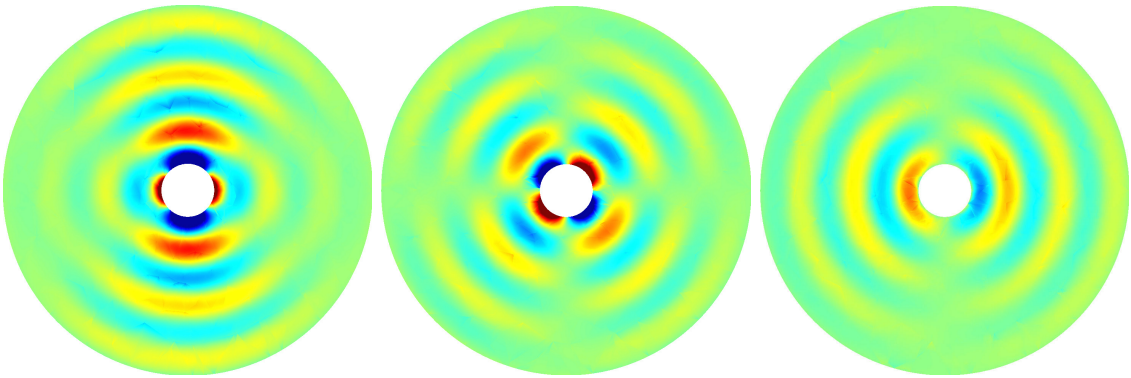
c) $\hat{z}$ component

Fig. 8.7: Imaginary component of the scattered electric field in the YZ plane.

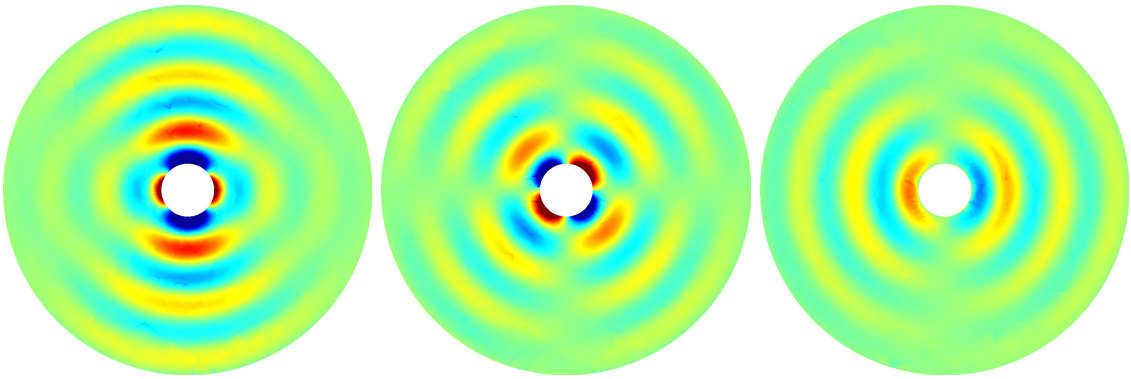
The figures show a good agreement between the analytic solution and HDGM for each component of the electric field. Close inspection reveals a number of artifacts produced by HDGM. Many of these artifacts appear as a jump in field value, which is to the discontinuous solution between elements. These can be seen, for example, in the $\hat{x}$ and $\hat{z}$ components of Figure 8.2. Another type of artifact can be seen when the polynomial field within an element does not accurately capture the solution. For example, in Figure 8.5 the $\hat{x}$ component in blue near the top of the image does not match the analytic solution. We see the nodal values are approximately correct, but the solution between nodes are inaccurate.

A number of figures show components of the electric field for which the analytic solution is entirely zero. Here, HDGM produces near-zero fields but has small discrepancies. Some of these artifacts show the nature of the 2^{nd} order fields as the solution varies along the faces of a given element. The number and severity of artifacts can be reduced by increasing the number of elements or the order used to represent their fields, however this comes with added computational cost.

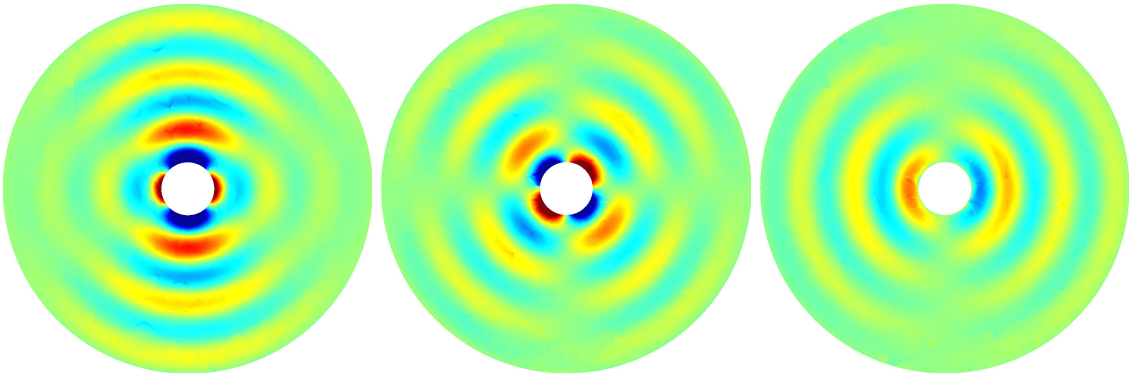
Next, the scattered electric fields computed using DGM, HDGM using a magnetic field basis, and HDGM using an electric field basis are compared. Only the XY plane is examined as the other planes follow a similar pattern. Figure 8.8 compares the real component in the XY plane and Figure 8.9 compares the imaginary component in the XY plane.



a) DGM solution



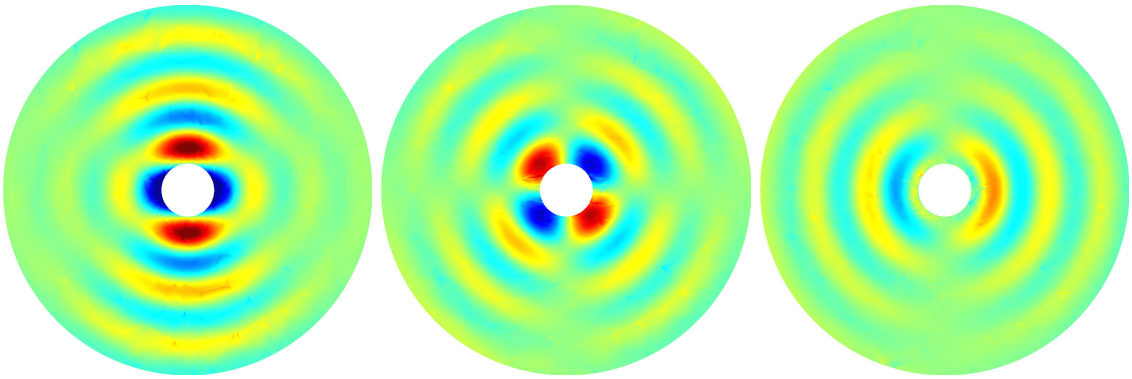
b) HDGM solution using a magnetic field basis



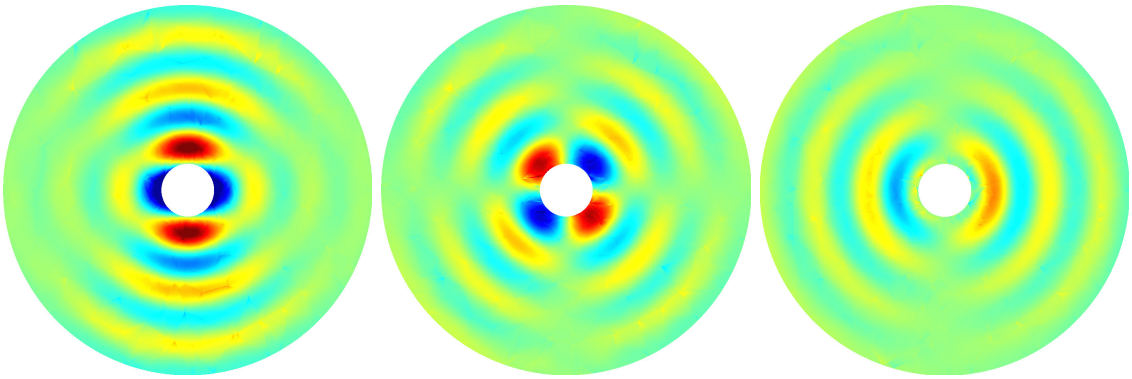
c) HDGM solution using an electric field basis

Fig. 8.8: A comparison of the scattered real components of the electric field in the XY plane.

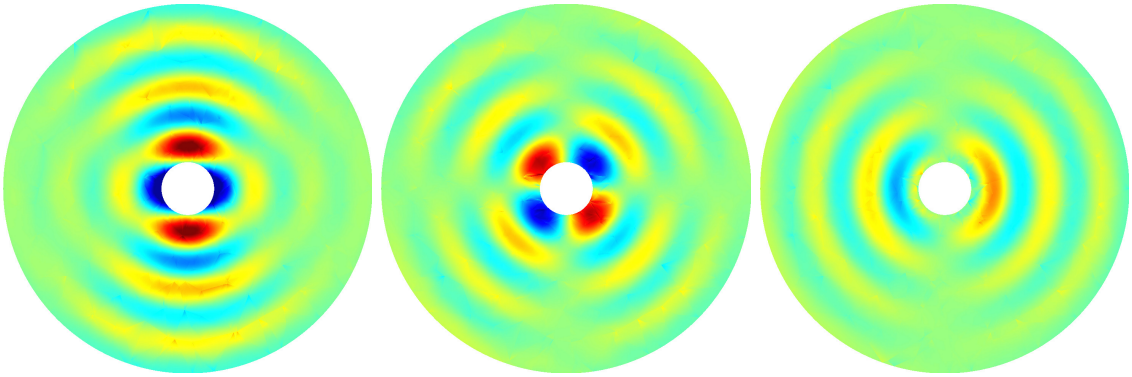
Consistent field patterns are seen regardless of formulation. Even artifacts are identical between solvers. In fact, further examination shows the results agree to floating point



a) DGM solution



b) HDGM solution using a magnetic field basis

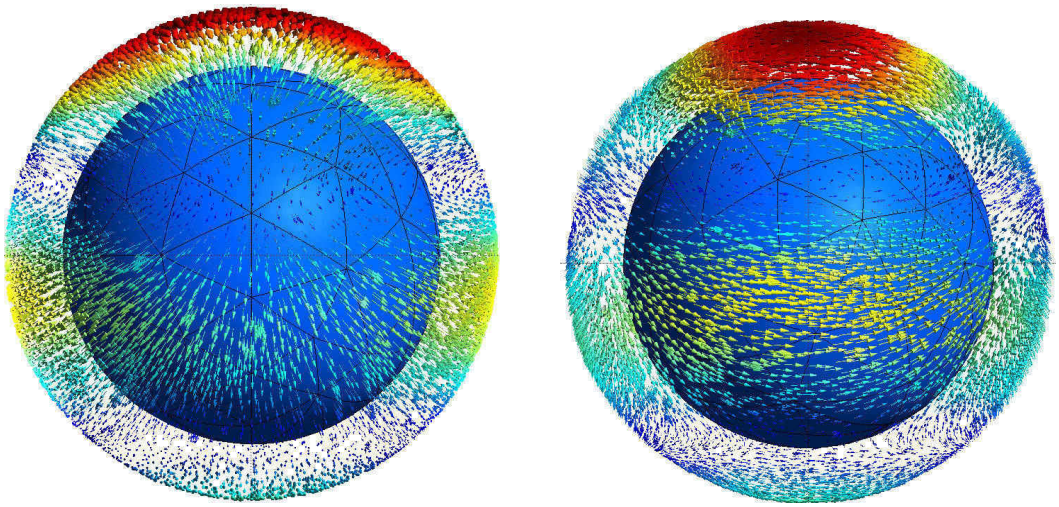


c) HDGM solution using an electric field basis

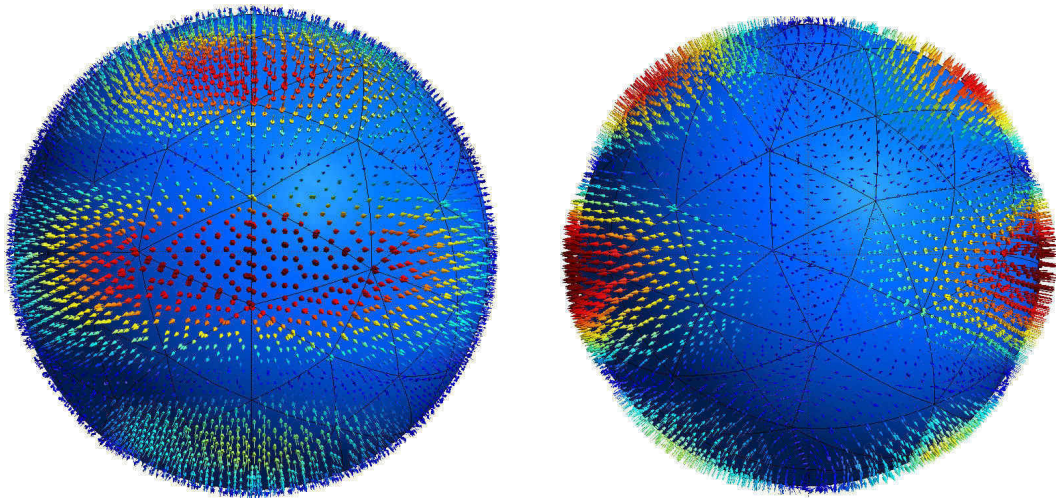
Fig. 8.9: A comparison of the scattered imaginary components of the electric field in the XY plane.

precision. This is expected as all 3 formulations use the same flux to couple elements, based on (2.14). They also used the same boundary conditions derived in Appendix C.

An interesting observation can be seen by examining the electric fields near the PEC sphere. Figure 8.10 plots the real and imaginary components of the scattered and total electric field vectors. The scattered fields are computed such that, when added to the incident field, the total fields are normal to the PEC sphere. This is expected, and helps verify proper implementation of PEC conditions, as the total electric field vector cannot have a tangential component on a PEC surface.



a) Real and imaginary scattered electric field



b) Real and imaginary total electric field

Fig. 8.10: Electric field near the PEC sphere.

8.1.2 Quantitative Analysis

In addition to the qualitative observations above, we quantify error by sampling the scattered electric field values at a number of points and compare to the analytic solution. We choose to only examine 3 field components in the XY plane; the real part of the x component along the $\hat{y}$ axis as shown in Figure 8.11, the imaginary part of the y component along the line $x = y$ as shown in Figure 8.12, and the real of the z component along the $\hat{x}$ axis as shown in Figure 8.13. The analytic solution for the selected components are shown in Figures 8.11 through 8.13. Since the computed fields are identical regardless of formulation, only HDGM using an electric field basis is shown.

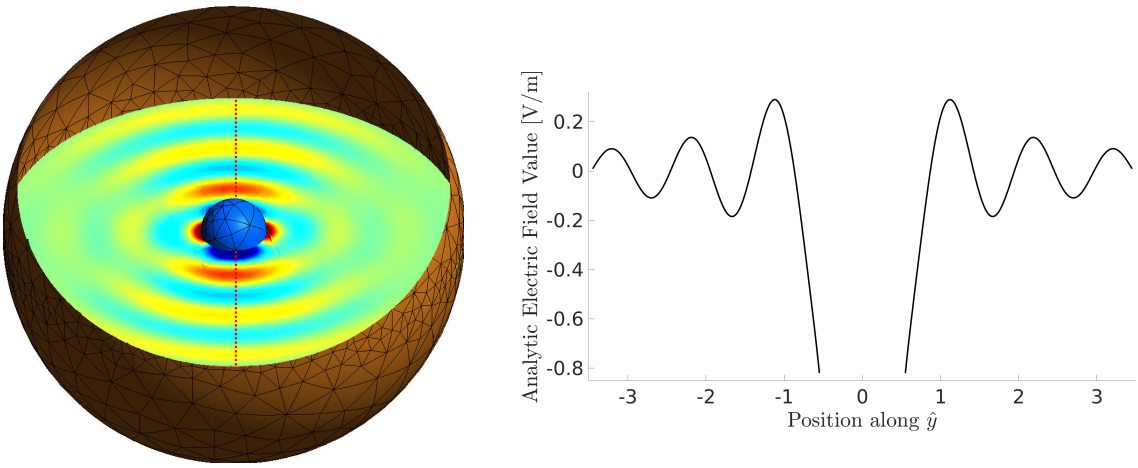


Fig. 8.11: Analytic solution for the x component of the real electric field along the $\hat{y}$ axis.

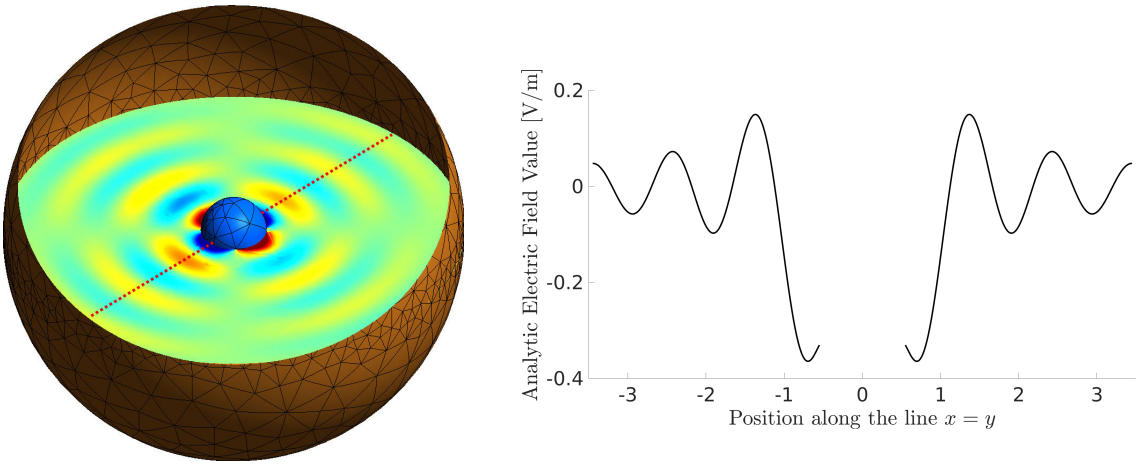


Fig. 8.12: Analytic solution for the y component of the imaginary electric field along the line $x = y$.

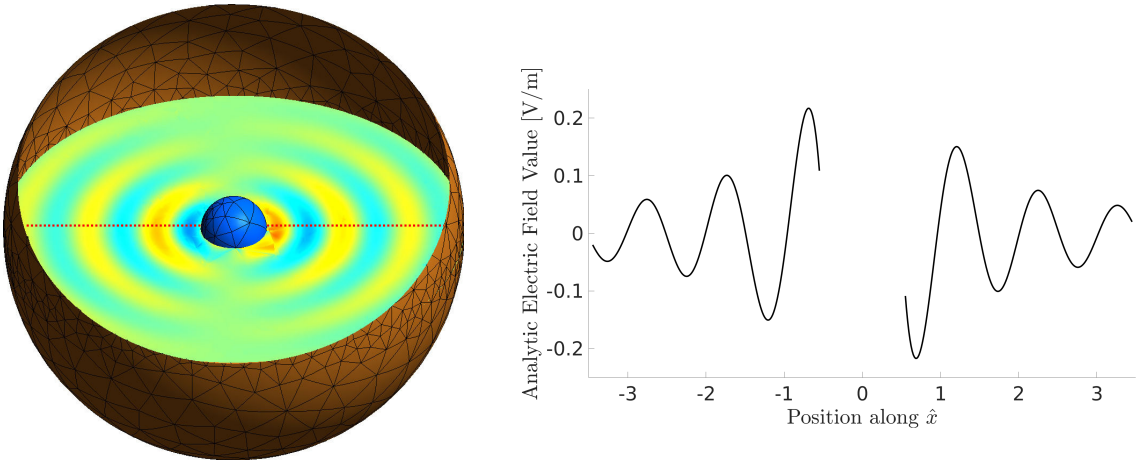


Fig. 8.13: Analytic solution for the z component of the real electric field along the $\hat{x}$ axis.

We examine the effect on accuracy due to changing the order of geometric representation, the order of field solution, and the number of elements. For each scenario an average relative error is computed by averaging the relative error from each component. Convergence for is shown in Figure 8.14.

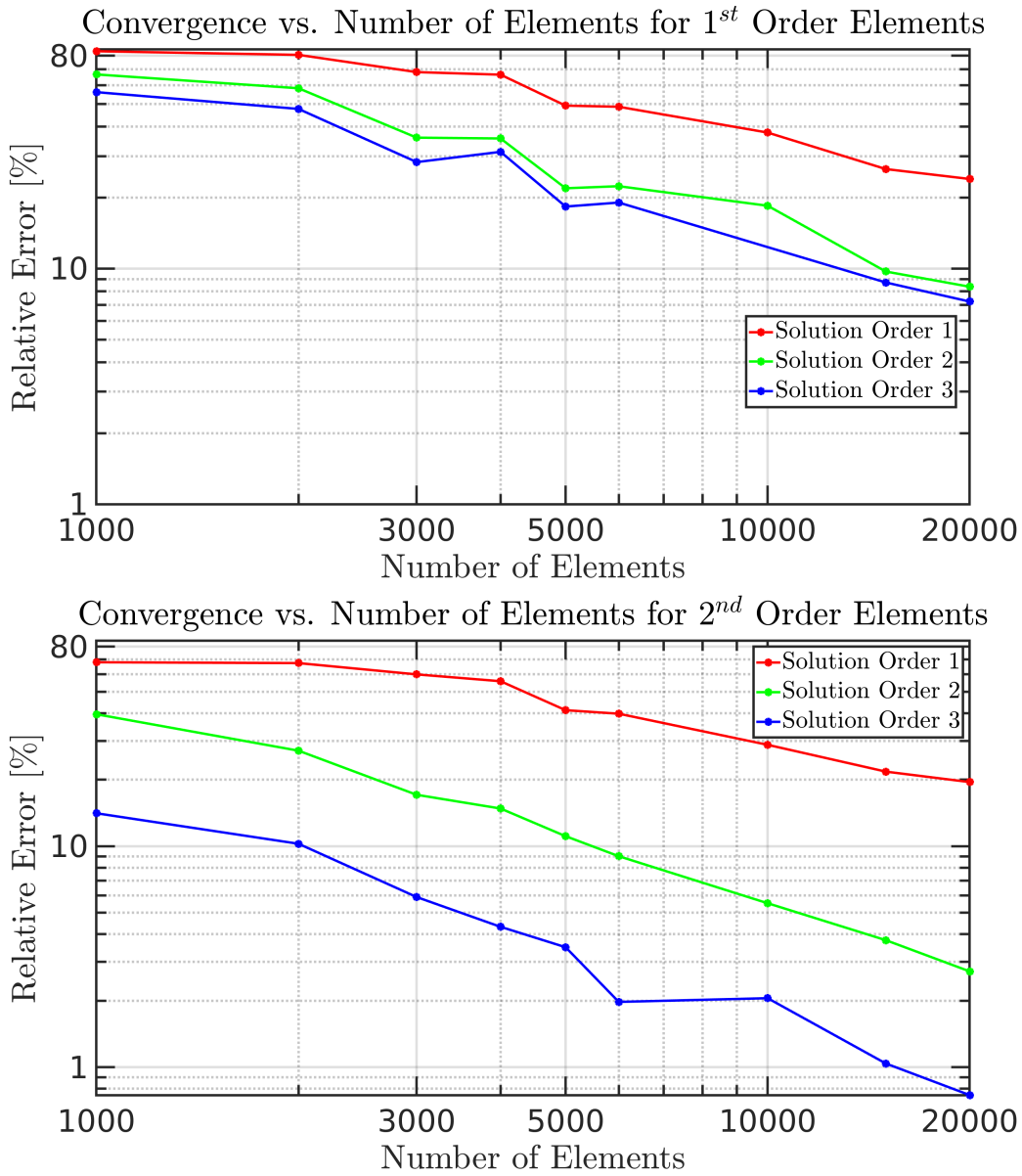


Fig. 8.14: Convergence analysis of HDGM.

From these results we see good convergence properties. First we see that, regardless of geometric and solution order, the solution becomes more accurate as the number of elements is increased. We also can see that the 3^{rd} order field solutions are always the most accurate and the 1^{st} order solutions are always the least accurate.

Comparing the geometric expansion order we see limited convergence when using linear elements. Even with a large number of elements and high field order, the poor geometric representation of linear elements limits the accuracy. This shows the need for high order geometric representations. Changing linear elements to high-order elements increases the accuracy for all solutions, and allowed the best accuracy achieved to be reduced from approximately 7% to less than 0.8%.

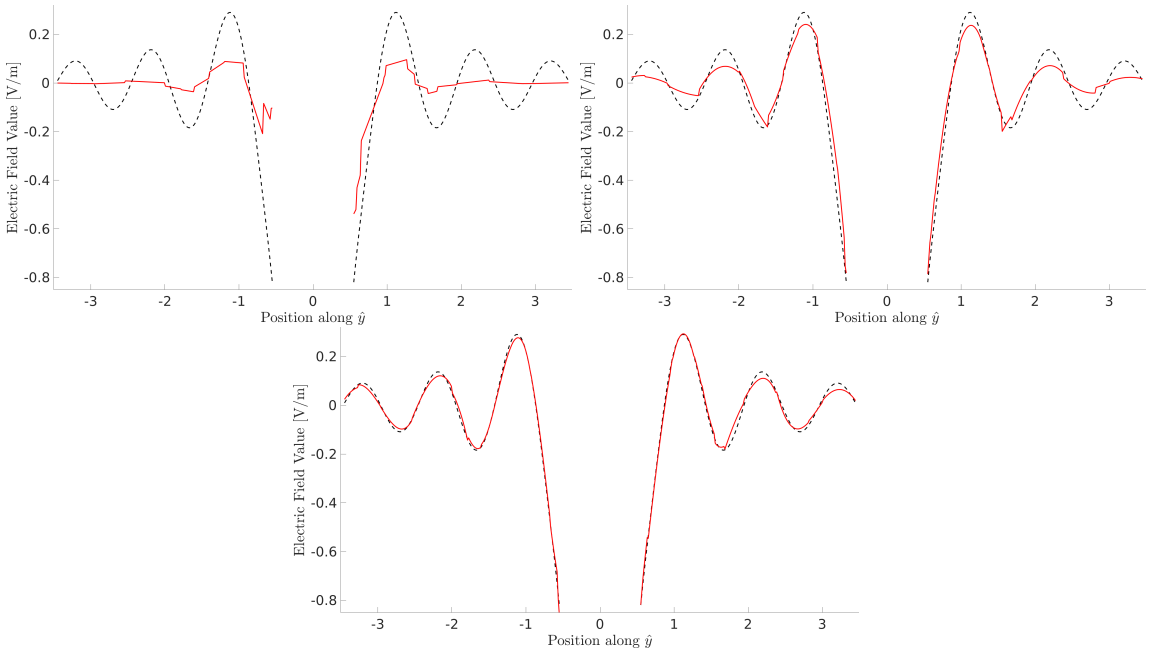


Fig. 8.15: HDGM solution on the same mesh for field orders 1 through 3, showing increased accuracy.

8.2 Scattering in Metallic Chambers

The next example compares the run time, memory consumption, and error of the DGM and HDGM solvers. It adds complexity as the problem does not have an analytic solution. Figure 8.16 shows the problem setup which simulates the scattering of an incident field inside a grain bin. The height and diameter of the grain bin are both approximately 10 [m]. A heap of grain shown in yellow which is modeled with a permittivity of $3 - j0.5$ and a region of air above the grain is modeled with a permittivity of $1 - j0.001$. The small loss in the air helps dampen resonances in the chamber. Lastly the bin surface is modeled as a PEC. A single $\hat{y}$ directed magnetic dipole at the location of the green dot is used to excite an incident field at 60 MHz. The analytic electric and magnetic incident fields in the XZ plane are shown in Figure 8.18.

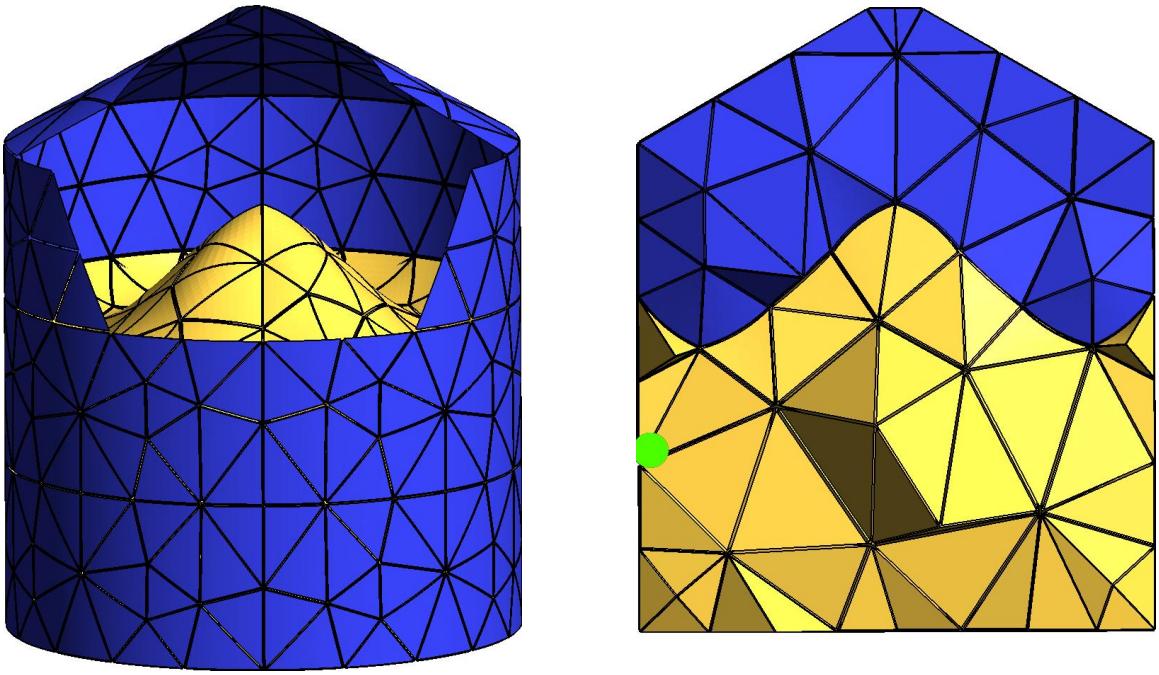


Fig. 8.16: Setup used for the forward problem.

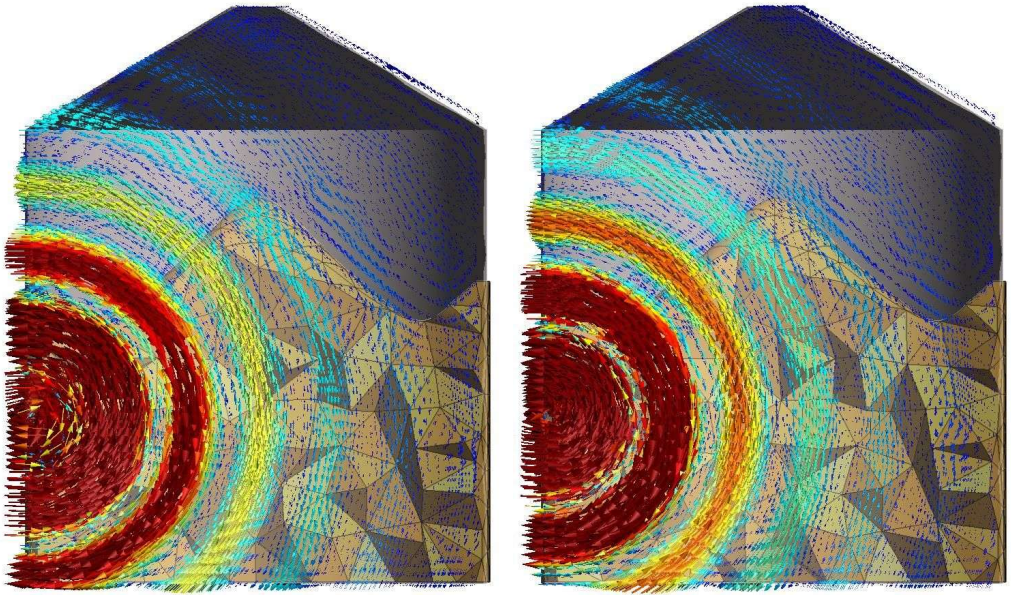


Fig. 8.17: Real and imaginary components of the analytic incident electric field.

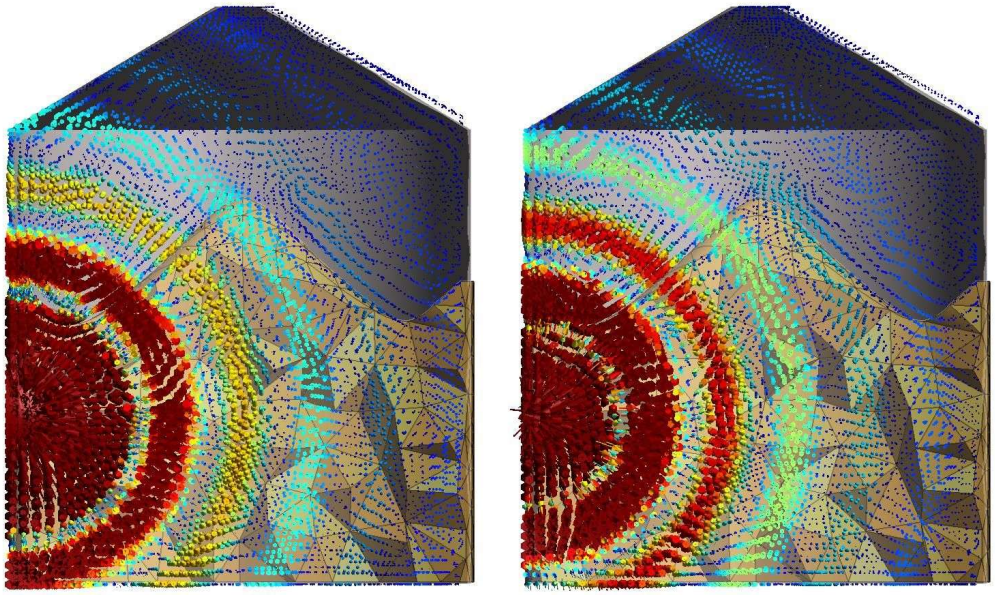


Fig. 8.18: Real and imaginary components of the analytic incident magnetic field.

Scattered vector fields computed using HDGM are shown below. Scattering is seen from both the grain and bin walls, partially due to the resonance nature of a PEC chamber.

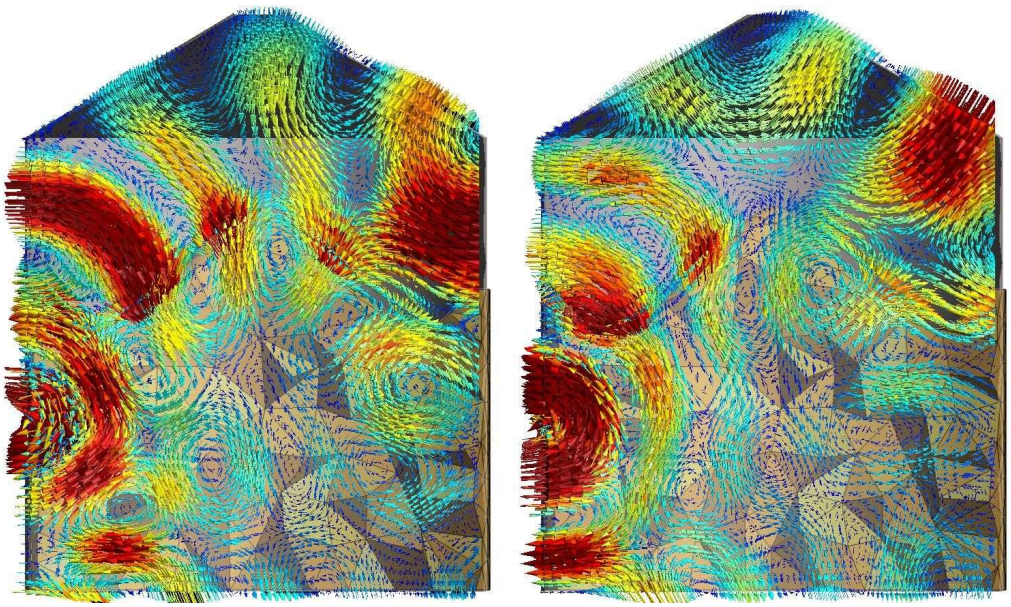


Fig. 8.19: Real and imaginary components of the scattered electric field.

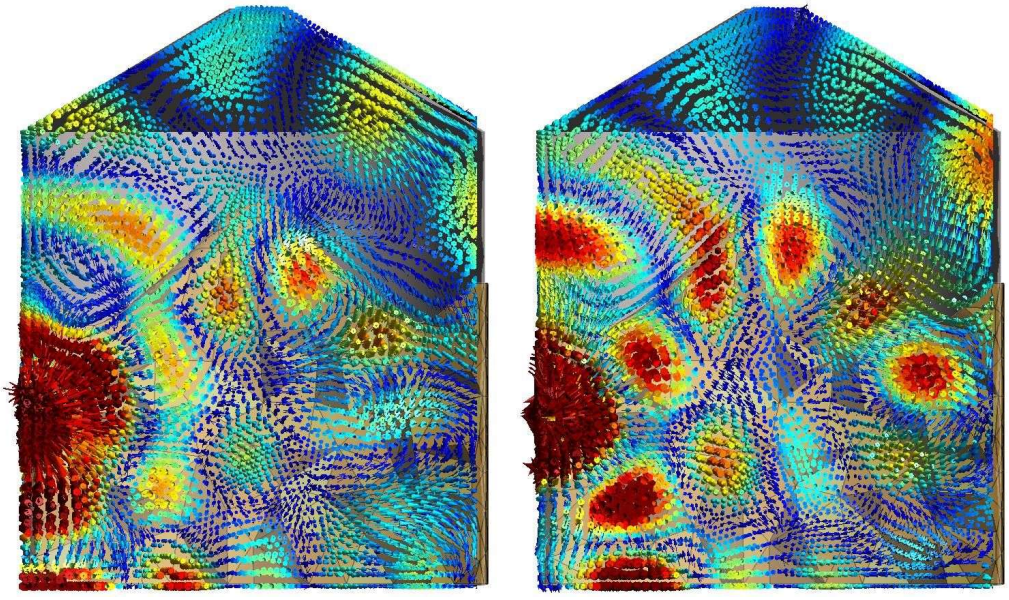


Fig. 8.20: Real and imaginary components of the scattered magnetic field.

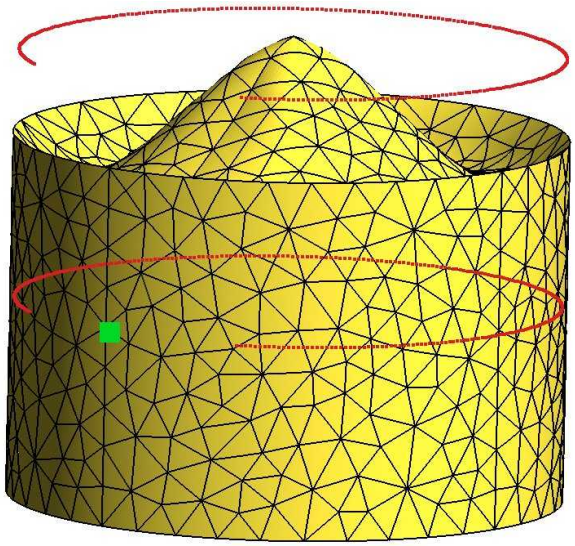


Fig. 8.21: Receivers are placed in two rings; one in the grain heap and the other in air.

A series of receivers are placed in two circular rings along the bin wall as shown in Figure 8.21. One ring measures field values within the grain and is placed at the height of the transmitter. The other ring captures field values in air and is placed half way between the height of grain on the wall and the bin eave. The real and imaginary components of the scattered field for each plane of receivers is shown in Figure 8.23. The receivers on these planes begin and end 30 degrees from the transmitter. Each receiver computes the tangential component of the magnetic field along the bin surface, which matches that used in real-world MWI [48].

To quantify error, a reference solution is generated by using HDGM to solve the problem on two very fine meshes. The two meshes consist of 7432 and 8800 elements, both using 2^{nd} order geometry and 5^{th} order fields. The fields computed using these meshes were compared, and showed a 0.2767% relative error signaling HDGM had converged. Next, the process was repeated using a FEM solver. The meshes contained 5.3 and 6.9 million linear elements. The solutions were compared showing a 3.1624% relative error. While this did not show as good of agreement as HDGM, additional attempts did not improve the

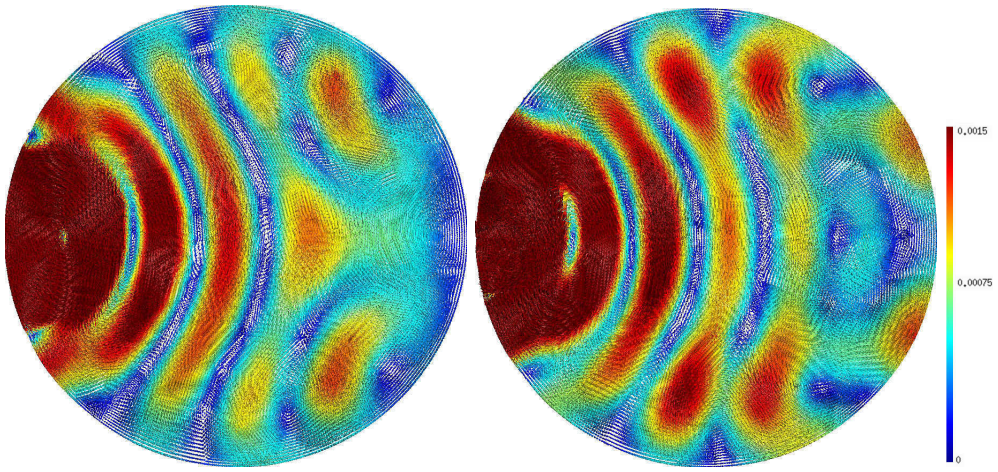


Fig. 8.22: Real and imaginary components of the scattered magnetic field along the plane of receivers in grain.

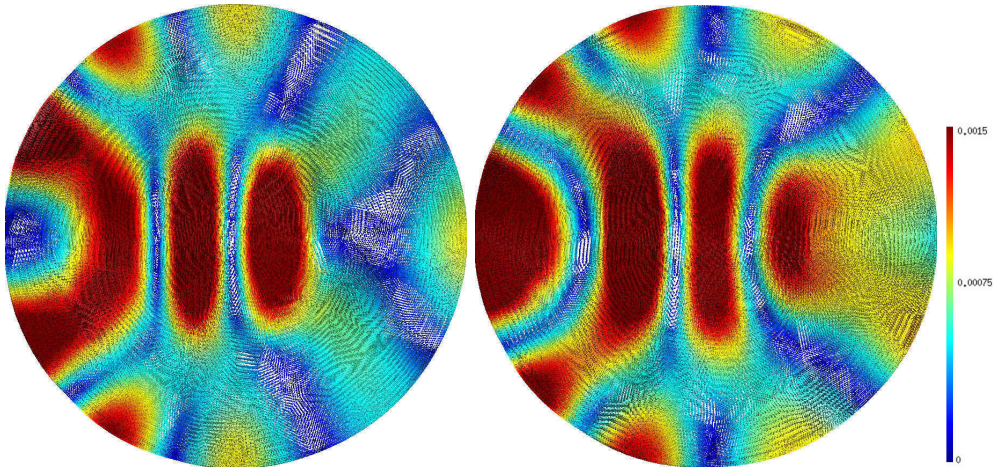


Fig. 8.23: Real and imaginary components of the scattered magnetic field along the plane of receivers in air.

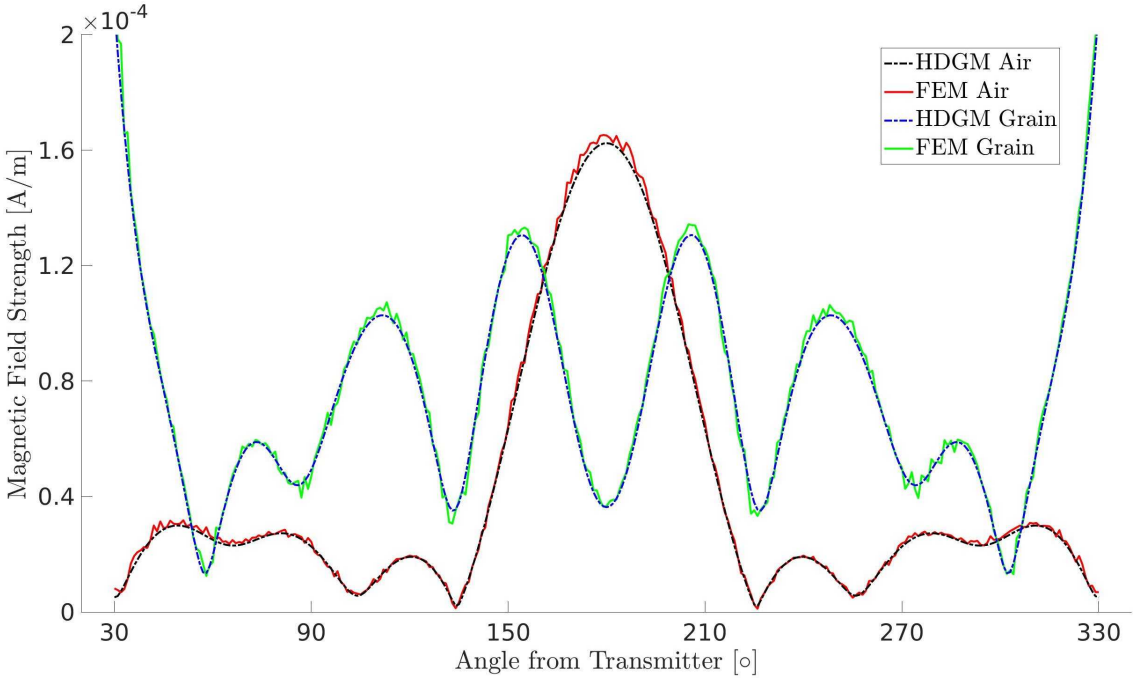


Fig. 8.24: Magnitude of the tangential magnetic field computed using the reference solution.

accuracy. Figure 8.24 shows the HDGM and FEM reference along both rings of receivers. It shows good agreement between the FEM and HDGM reference solutions. It also shows error in the FEM solution, likely due to the linear approximations as we would expect a smooth solution. For this reason we use the HDGM solution as the reference.

The solution to this problem is computed for a large variety of element solution orders and number of elements for both 1^{st} and 2^{nd} order geometric elements. A summary of results is shown in Table 8.1 which compare run time, memory usage, and accuracy of each solve. Note that the run time is measured using two 16-core intel Xeon gold 6130 processors.

Solver	Elements	Geometric Order	Solution Order	Run Time [s]	Memory Usage [GB]	Relative Error [%]
FEM	248 875	1	1	14	15.6	6.3640
FEM	496 317	1	1	35	33.2	4.4624
FEM	992 653	1	1	102	79.7	3.5489
FEM	1 990 780	1	1	331	194.5	3.0303
FEM	2 999 880	1	1	622	321.0	2.5700
FEM	3 969 210	1	1	1041	465.1	2.8006
FEM	5 316 370	1	1	2351	675.3	2.5635
FEM	6 949 120	1	1	3751	927.6	2.1612
DGM	1 486	1	1	1.9	3.0	49.484
DGM	1 486	1	2	6.7	8.9	22.2809
DGM	1 486	1	3	23.3	24.3	17.8010
DGM	1 486	1	4	81.0	63.6	15.1823
DGM	1 486	1	5	263.3	155.2	13.8103
DGM	4 038	1	1	4.6	6.4	43.0828
DGM	4 038	1	2	26.3	24.6	9.2875
DGM	4 038	1	3	117.6	81.9	8.7947
DGM	4 038	1	4	439.8	218.7	8.0232
DGM	4 038	1	5	1421.0	521.9	7.8196
HDGM	1 486	2	1	0.6	2.3	42.3248
HDGM	1 486	2	2	1.1	3.5	11.9644
HDGM	1 486	2	3	2.2	6.4	4.3795
HDGM	1 486	2	4	3.6	12.2	3.0362
HDGM	1 486	2	5	6.9	23.4	1.9269
HDGM	4 038	2	1	1.2	3.3	19.2139
HDGM	4 038	2	2	2.4	6.5	4.3994
HDGM	4 038	2	3	5.3	14.8	1.5908
HDGM	4 038	2	4	12.2	31.3	0.5295
HDGM	4 038	2	5	27.4	60.9	0.4210
HDGM	7 432	2	5	152.1	131.1	0.2767
HDGM	8 038	2	5	210.4	160.4	REF

Table 8.1: Comparison of run time, memory, and accuracy for FEM, DGM, and HDGM.

We begin examining this table by inspecting the performance of the FEM solver. We see that it requires many more elements, but is able to solve the matrix system fairly quickly for the number of elements. While FEM was never able to achieve as low of error as HDGM, it was able to achieve much better error when under-resolved. Interestingly, the

FEM solution with 6.9 million elements was able to achieve 2.1612% relative error when comparing to HDGM, but only achieved 3.1624% relative error when comparing it to the 5.3 million element FEM solution.

Examining the performance of DGM, we see that it does not reach a reasonable error level. While run time and memory usage was generally lower than FEM, it only obtained a 7.8% relative error when using 5^{th} order field solutions and 4000 elements. Performance as field order increases also shows a lack of convergence. The poor performance is partially due to the lack of high order geometry capabilities. If the DGM solver implementation allowed high order geometries, the exact same error as HDGM using high order geometries is expected.

Next, HDGM is examined. Note that the times presented are for HDGM using an electric field basis. HDGM using a magnetic field basis is expected to have similar run times and error. Timing and error for HDGM using first order geometric elements can be approximated by using the timing of HDGM with the accuracy of DGM. HDGM shows greatly improved performance over both FEM and DGM. We see using 1500 elements it was able to achieve 1.9% relative error, and using 4000 elements achieved 0.27% error; better than the best from FEM and DGM. Comparing to DGM, we see the increased performance across the board from high order representation of geometry. Poor performance is seen when HDGM is under resolved, which is seen for first and second order field representations.

Comparing run time and memory consumption HDGM shows drastic performance improvement over both DGM and FEM. For a mesh consisting of 4000 elements using 5^{th} order fields HDGM was approximately 50 times faster and used 8 times less memory than DGM, due to the reduced DOF in the global matrix. HDGM was also able to reduce error further than FEM using 4000 elements with 3^{rd} order fields or 1500 elements using 5^{th} order fields. These results reduced the memory requirements by a factor of 60 and the computational demand by a factor of 500.

8.3 HDGM for CSI

In this example we examine the use of the HDGM solver for CSI. A real-world application is used, in which a region of spoiled grain is detected inside a grain bin [48]. Section 8.3.1 begins by using synthetic data and is followed by 8.3.2 which uses experimentally collected data. Since the use of DGM for MWI has previously been studied [5, 21, 34, 41, 43–46], this section focuses on the benefits of HDGM over DGM.

8.3.1 CSI Results Using Synthetic Data

Tests involving synthetic data are useful to examine the ability of CSI using HDGM in a well controlled environment. Data is generated using the HDGM forward solver. Each datapoint measures the magnitude of the tangential magnetic value along the bin wall. These field values are collected for 24 antennas inside the bin. Each antenna acts as both a transmitter and receiver, allowing a total of 552 datapoints to be collected.

The model used to collect this data is shown in Figure 8.25. Here we have grain which is modeled using a permittivity of $4.5-j0.5$, a sphere of radius 1 [m] representing spoiled grain having a permittivity of $5.5 - j0.8$, air with a slight loss and a permittivity of $1 - j0.001$, a concrete floor as an impedance boundary using a permittivity of $4 - j0.1$, and a metallic bin which is modeled as a PEC.

To avoid an inverse crime, 10% white uniform noise was added to the data. The mesh used for the inverse problem was changed and did not include the spoiled region. CSI using both a DGM and HDGM forward solver were run for 150 iterations. Inversion results using linear elements are shown Figures 8.26 and 8.27. Results when using high order elements are shown in Figures 8.28 and 8.29. Each figure shows the location of permittivities above

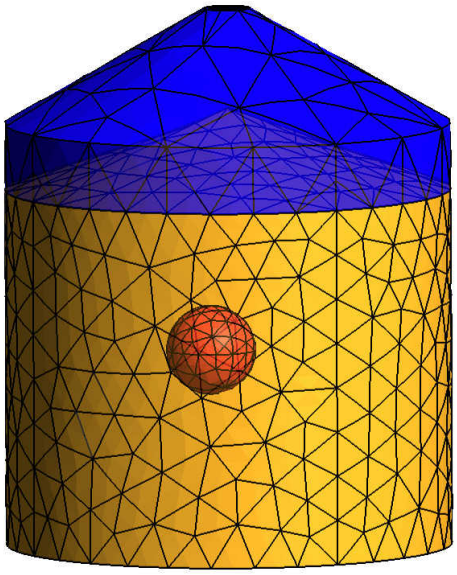
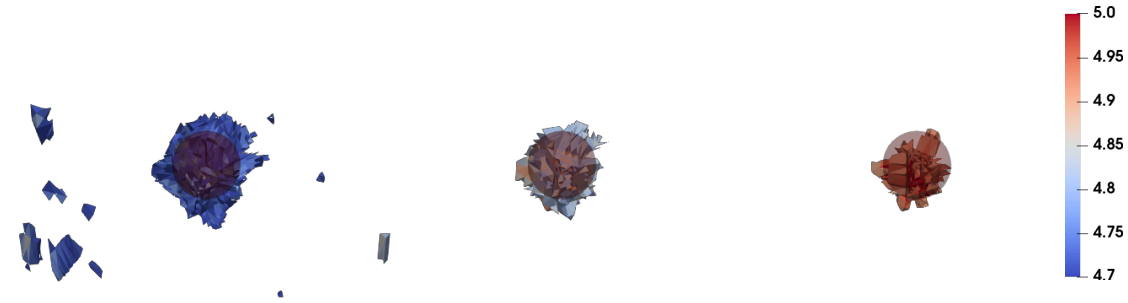


Fig. 8.25: Problem setup used to generate synthetic data.

a given threshold value, in addition to the expected location of spoilage using an opaque red sphere.

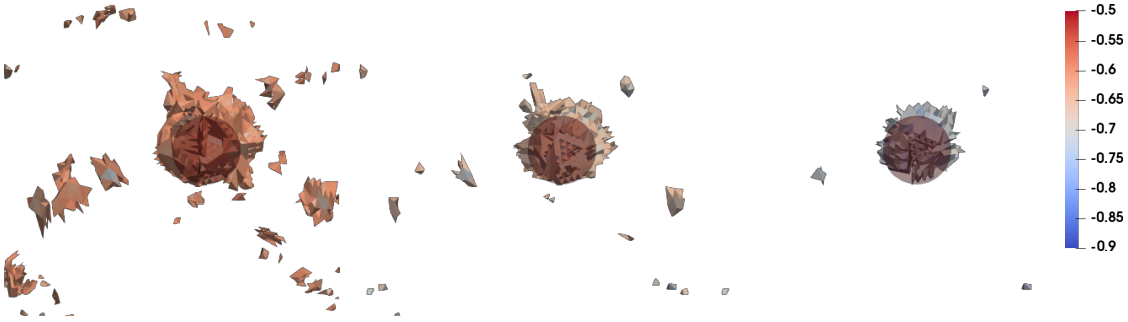
The location of the spoilage is easily detected using synthetic data. However, reconstructed values did not reach the same values used in the forward problem. The reconstruction had a maximum real permittivity of 5.26 while the forward model used a value of 5.5, and the reconstructed imaginary component had a minimum value of -0.899 which is lower than the true value of -0.8.

CSI results were also obtained using the DGM forward solver. Visually, they were identical to those produced using HDGM as the forward solvers give identical results. The data error was slightly lower when using DGM; which minimized the relative data error from 0.7974 to 0.0005815 versus HDGM which reached 0.0005979. This slight difference is attributed to compounding rounding error over 150 iterations. While the results produced using DGM and HDGM were similar, the computation performance was not. Using DGM, the initial



a) b) c)

Fig. 8.26: Location of real permittivites above a) 4.7, b) 4.8, and c) 4.9 when using linear elements.



a) b) c)

Fig. 8.27: Location of imaginary permittivites below a) -0.6, b) -0.65, and c) -0.7 when using linear elements.

Fig. 8.28: Location of real permittivites above a) 4.7, b) 4.8, and c) 4.9 when using high order elements.

matrix solve took 102 [s] and each iteration took 47.8 [s] on average. HDGM was able to improve this, taking 6.7 [s] for the initial solve and an average of 10.6 [s] per iteration.

Using high order geometries we see further improvement of inversion results. Both the real and imaginary parts of the reconstructed permittivity show improvement, as there are less artifacts surrounding the true location. The maximum real permittivity was 5.396, much closer to the true value of 5.5. The imaginary component had a minimum value of -0.874, closer to the true value than when using linear elements, but still lower than the true value of -0.8.

The relative data error was reduced to 0.000189 which shows great improvement over results using linear elements. However, this may be biased as the synthetic data was generated used high order elements.

a) b) c)

Fig. 8.29: Location of imaginary permittivites below a) -0.6, b) -0.65, and c) -0.7 when using high order elements.

8.3.2 CSI Results Using Experimental Data

We now investigate the performance of CSI with the HDGM forward solver using experimental data. This data was collected using the 151Research GrainViz system after a region of spoiled grain was buried in a hopper bin. This example has been previously studied using CSI and a FEM forward solver [48].

Inversion using the HDGM solver used 2435 high order elements and 3^{rd} order polynomials to represent fields and CSI update variables. The height and approximate permittivity of grain is assumed to be known, and a frequency of 89.6 MHz is selected to match the setup used by FEM.

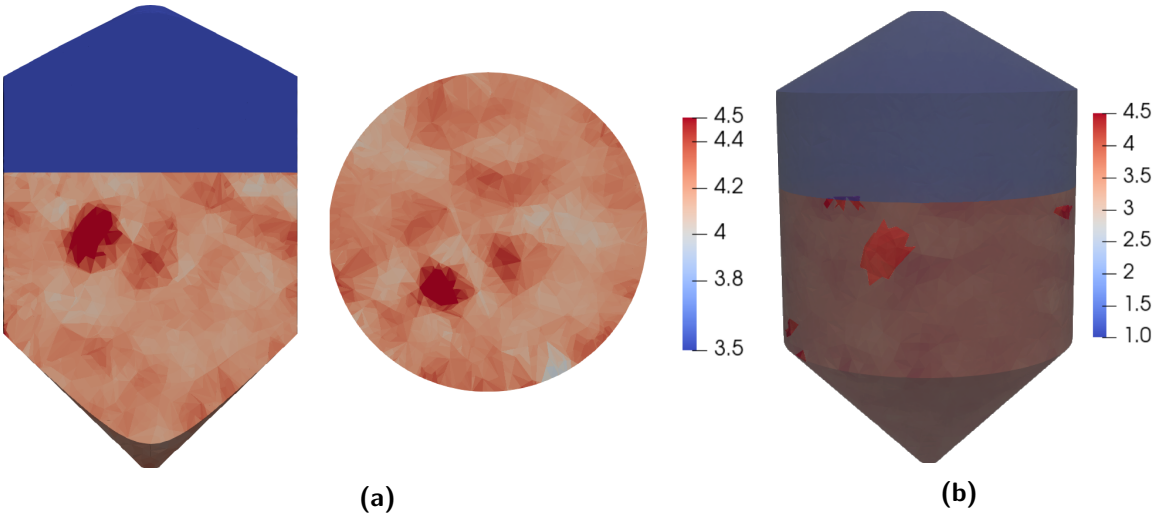


Fig. 8.30: CSI results using HDGM as the forward solver.

Good agreement is seen between the results produced using the HDGM and FEM forward solvers, both in terms of size and location of spoiled grain. These results agree with the expected location of the spoiled grain. Since FEM uses many more elements the images produced are more granular.

In terms of performance, FEM took approximately 33 [s] for the initial matrix solve and 30

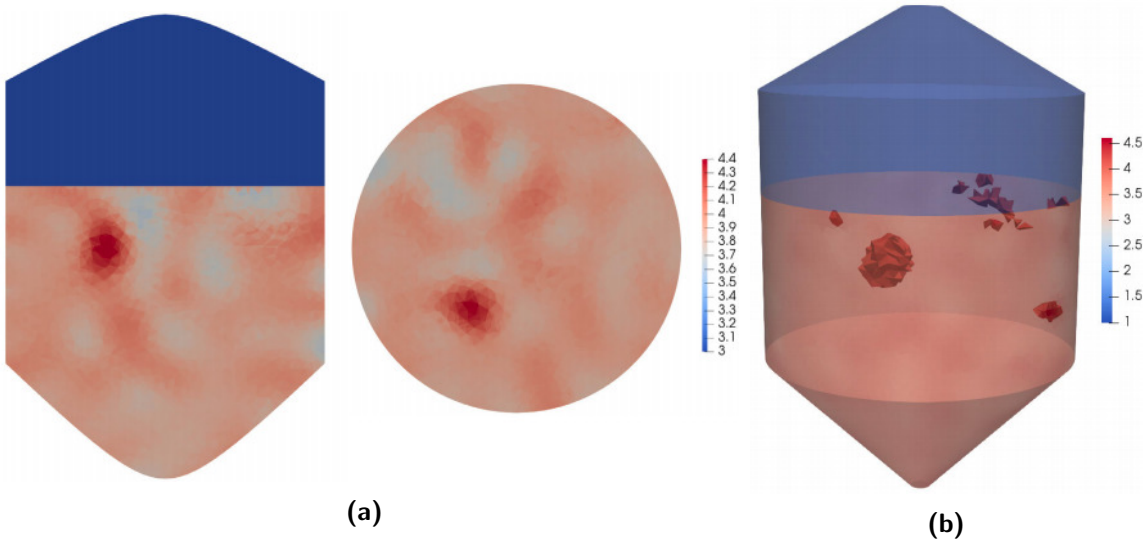


Fig. 8.31: CSI results using FEM as the forward solver. These results are presented in [48] and are repeated here with the authors permission.

[s] per iteration, DGM took 130 [s] for the initial solve and 66 [s] per iteration, and HDGM took 7 [s] for the initial matrix solve and 20 [s] per iteration. Once again this shows the improved performance of HDGM over DGM with no loss of accuracy.

While HDGM had exceptional performance for forward problems, the speedup over FEM for CSI was limited. This is attributed to the fact that, before factorization, the average density of the FEM matrix was 14 terms per row, while HDGM averaged 79 terms per row. This resulted in FEM requiring 0.29 [s] per right hand side to compute both forward substitution and back substitution, and HDGM requiring 0.14 [s]. HDGM additionally requires evaluating the local system, which further limited speedup.

9

Conclusions and Future Work

9.1 Conclusions

The work presented in this thesis constitutes the theory, implementation, and results from a time-harmonic 3D Hybridizable Discontinuous Galerkin Method (HDGM) forward solver for use in Microwave Imaging (MWI) algorithms. Starting from Maxwell's equations, Chapter 2 derived the strong form flux-split Partial Differential Equation (PDE) by discretizing Maxwell's equations. This PDE is formulated to compute the scattered electric and magnetic fields in the presence of constitutive parameters and boundary conditions on an unstructured mesh. Chapter 3 examined how to represent fields using a polynomial basis within each element. Using this basis the mass matrix and stiffness matrices were derived. These allow the evaluation of inner products between basis polynomials, and inner products of a basis polynomial with the derivative of another basis polynomial, respectively, which

is required to numerically model the strong form flux-split PDE. These derivation were repeated for high-order elements, which accurately model geometric features using a polynomial basis. This required introducing quadrature-based integration, and a non-constant Jacobian of the transformation.

The Discontinuous Galerkin Method (DGM) was introduced in Chapter 4, and used the elements' basis to compute a solvable numerical system of equations based on the strong form flux-split PDE. HDGM was finally introduced in Chapter 5, and improved upon DGM by introducing the local system. The local system allowed high-order representation of fields within each element, using only a single field tangential to the elements' surface. The numerical system of equations produced by HDGM was derived by coupling elements using only the surface fields. This significantly reduced the degrees of freedom in the global system of equations, at the expense of added computation during matrix construction.

Implementation details were covered in Chapter 6, including the parallelization scheme to ensure fast computations. The Contrast Source Inversion (CSI) imaging algorithm was briefly covered in Chapter 7, which focused on the derivations required to make use of the HDGM forward solver within CSI.

Lastly, Chapter 8 introduced several results to validate the HDGM forward solver and a number of its features. It showed accurate high-order reconstruction of scattered fields using only tangential surface fields. It was compared to existing forward solver algorithms which showed improved performance in both computational time and memory required. Comparing HDGM to DGM showed the reduced computational cost came at no expense as it maintained the same high-order accuracy of DGM to machine precision. The improved performance allows larger problems or higher frequencies to be simulated. While HDGM generally achieved better performance, comparisons to a finite element method showed HDGM has poor performance for under-discretized problems. Use of the HDGM forward

solver was then examined for both synthetic and experimental CSI problems. This showed significant performance benefits which helps MWI overcome one of its largest drawbacks; the computational demand.

9.2 Future Work

There are a number improvements to the HDGM algorithm, and its application for MWI, that deserve further investigation. Some of these include:

- Use of the HDGM solver for Gauss-Newton inversion algorithms. Implementation using DGM already exists, which shows a number of imaging benefits [5].
- Implementation of the superconvergent local Discontinuous Galerkin method, which allows field recovery of order $(p+2)$ while still using p^{th} order representation of surfaces fields [26].
- Use of graphics processing units to improve performance. Preliminary tests show this can significantly improve computational time of elements' local systems due to the high level of data parallel operations [46].
- Additional examination into the use of multiple frequencies in inversion algorithms. Some testing has been done to this end [45].
- Further acceleration of the forward solver using domain decomposition which has shown promising performance improvements [49].
- Better antenna modeling as each antenna is modeled as an ideal point source. Baumann [50] introduces a formulation using port planes which may model antenna flux more accurately.

A

Derivation of the Relation of Volume Between Coordinate Spaces

This appendix contains the proof that the determinant of the Jacobian and the transformation gives the volumetric relation between Cartesian and barycentric coordinate spaces.

In (A.1) we compute the change in terms of dx, dy, and dz with respect to an infinitesimally small change dr, ds, and dt in the barycentric coordinates. In (A.2) we define the triple product rule which computes the volume of a general cube given 3 edge lengths. Finally in (A.3) we use these values to find the corresponding change in volume between V_n and B_n coordinates.

$$\begin{aligned} dx &= \frac{\partial x}{\partial r}dr + \frac{\partial x}{\partial s}ds + \frac{\partial x}{\partial t}dt \\ dy &= \frac{\partial y}{\partial r}dr + \frac{\partial y}{\partial s}ds + \frac{\partial y}{\partial t}dt \\ dz &= \frac{\partial z}{\partial r}dr + \frac{\partial z}{\partial s}ds + \frac{\partial z}{\partial t}dt \end{aligned} \tag{A.1}$$

$$V_{a,b,c} = \vec{a} \cdot (\vec{b} \times \vec{c}) = \begin{vmatrix} a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \\ c_1 & c_2 & c_3 \end{vmatrix} \tag{A.2}$$

$$\begin{aligned} V_{barycentric} &= dr\ ds\ dt \\ V_{Cartesian} &= d\vec{x} \cdot (d\vec{y} \times d\vec{z}) \\ &\Rightarrow \left(\frac{\partial x}{\partial r}dr + \frac{\partial x}{\partial s}ds + \frac{\partial x}{\partial t}dt\right) \cdot \left[\frac{\partial y}{\partial r}dr + \frac{\partial y}{\partial s}ds + \frac{\partial y}{\partial t}dt\right] \times \left[\frac{\partial z}{\partial r}dr + \frac{\partial z}{\partial s}ds + \frac{\partial z}{\partial t}dt\right] \\ &\Rightarrow \begin{vmatrix} \frac{\partial x}{\partial r} & \frac{\partial x}{\partial s} & \frac{\partial x}{\partial t} \\ \frac{\partial y}{\partial r} & \frac{\partial y}{\partial s} & \frac{\partial y}{\partial t} \\ \frac{\partial z}{\partial r} & \frac{\partial z}{\partial s} & \frac{\partial z}{\partial t} \end{vmatrix} \begin{bmatrix} dr \\ ds \\ dt \end{bmatrix} \\ &\Rightarrow \left|\boldsymbol{\mathcal{J}}\right| drdsdt \end{aligned}$$

$$\frac{V_{Cartesian}}{V_{barycentric}} = |\boldsymbol{\mathcal{J}}| \tag{A.3}$$

B

Additional Lagrange Polynomials

This appendix shows additional Lagrange polynomials and derivatives of Lagrange polynomials for elements of order 4, 5, and 6.

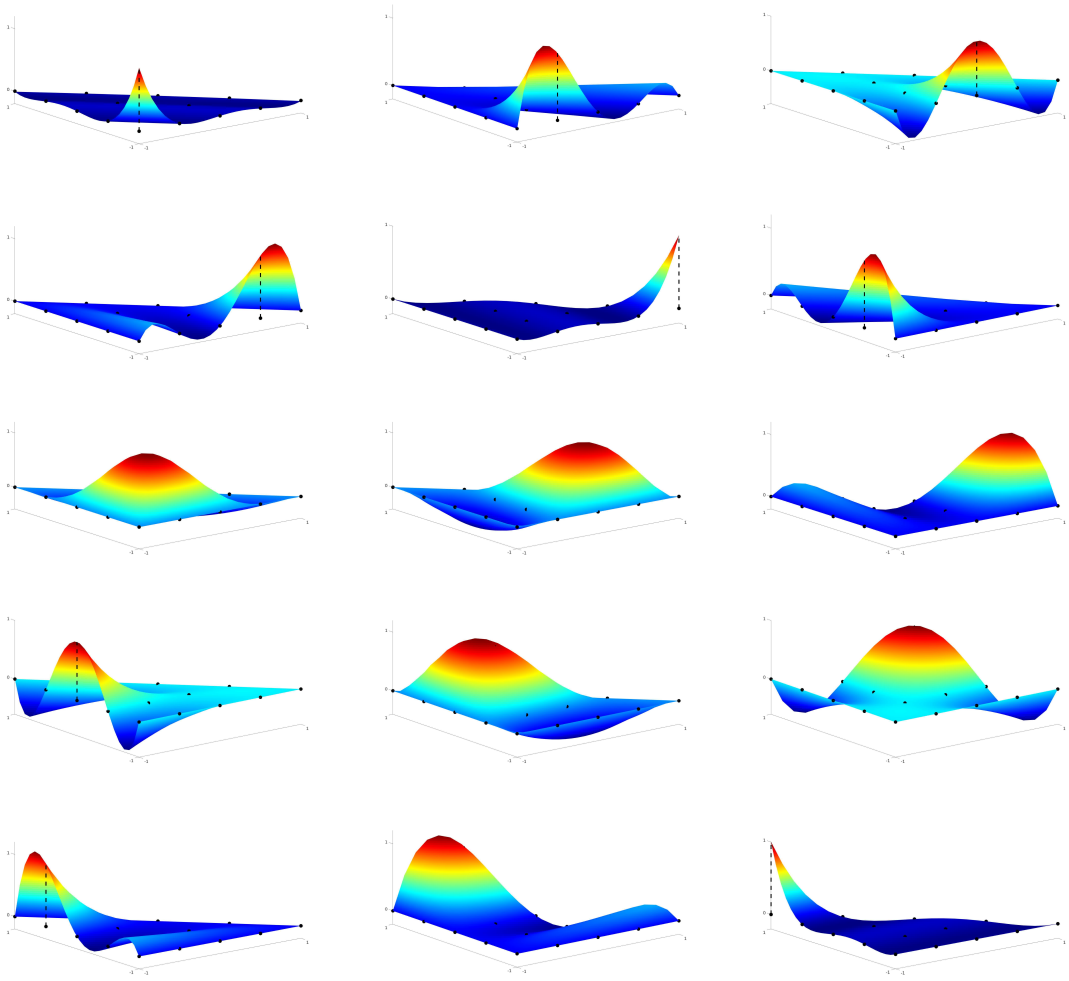


Fig. B.1: Lagrange polynomials for a 4^{th} order element.

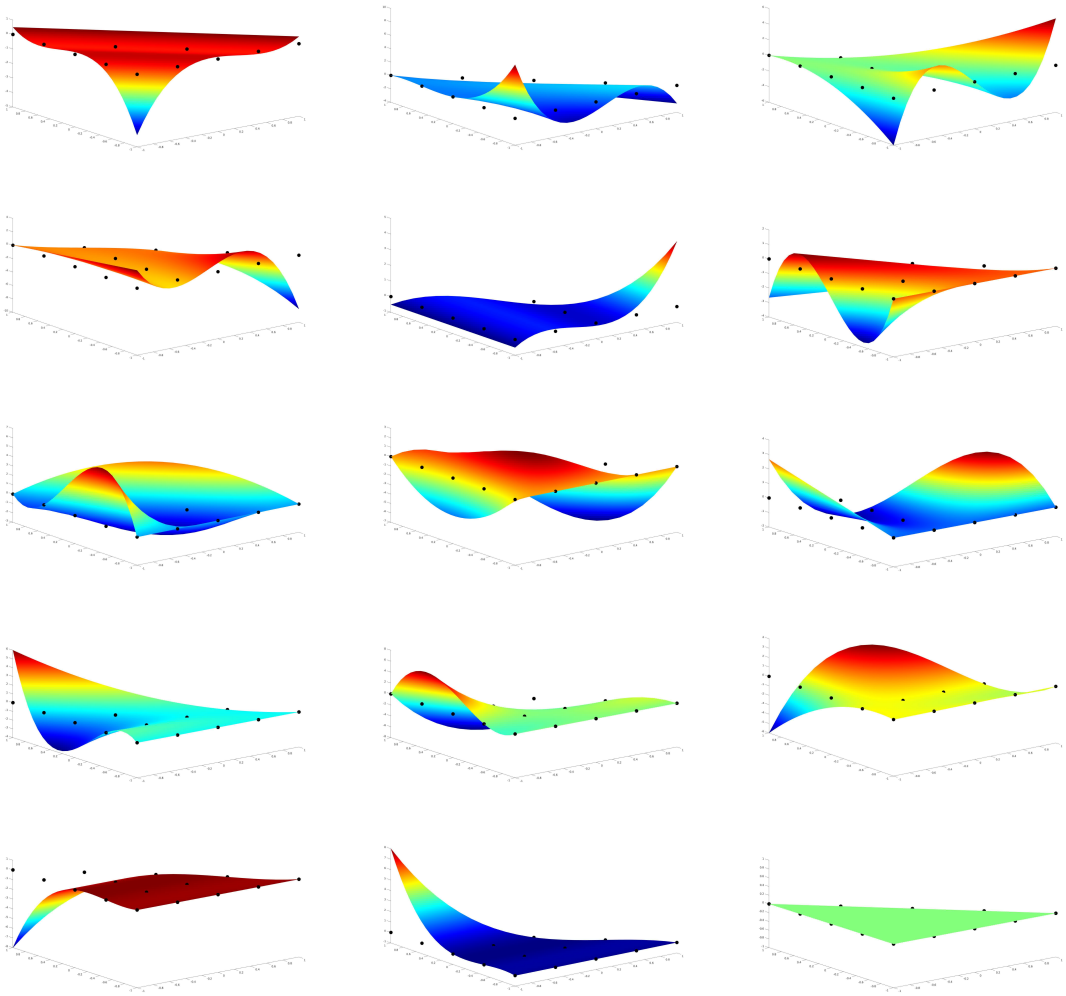


Fig. B.2: The derivative with respect to r of the Lagrange polynomials for a 4^{th} order element.

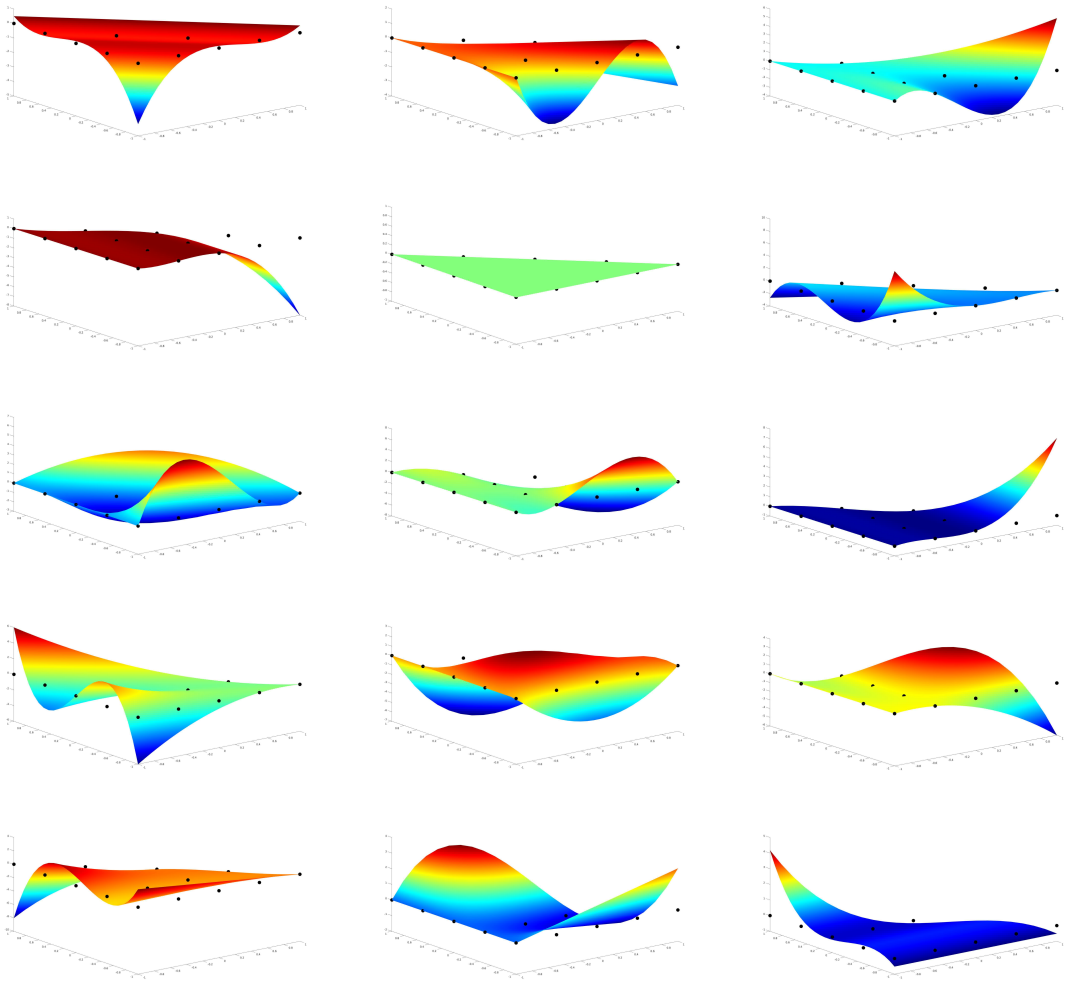


Fig. B.3: The derivative with respect to s of the Lagrange polynomials for a 4^{th} order element.

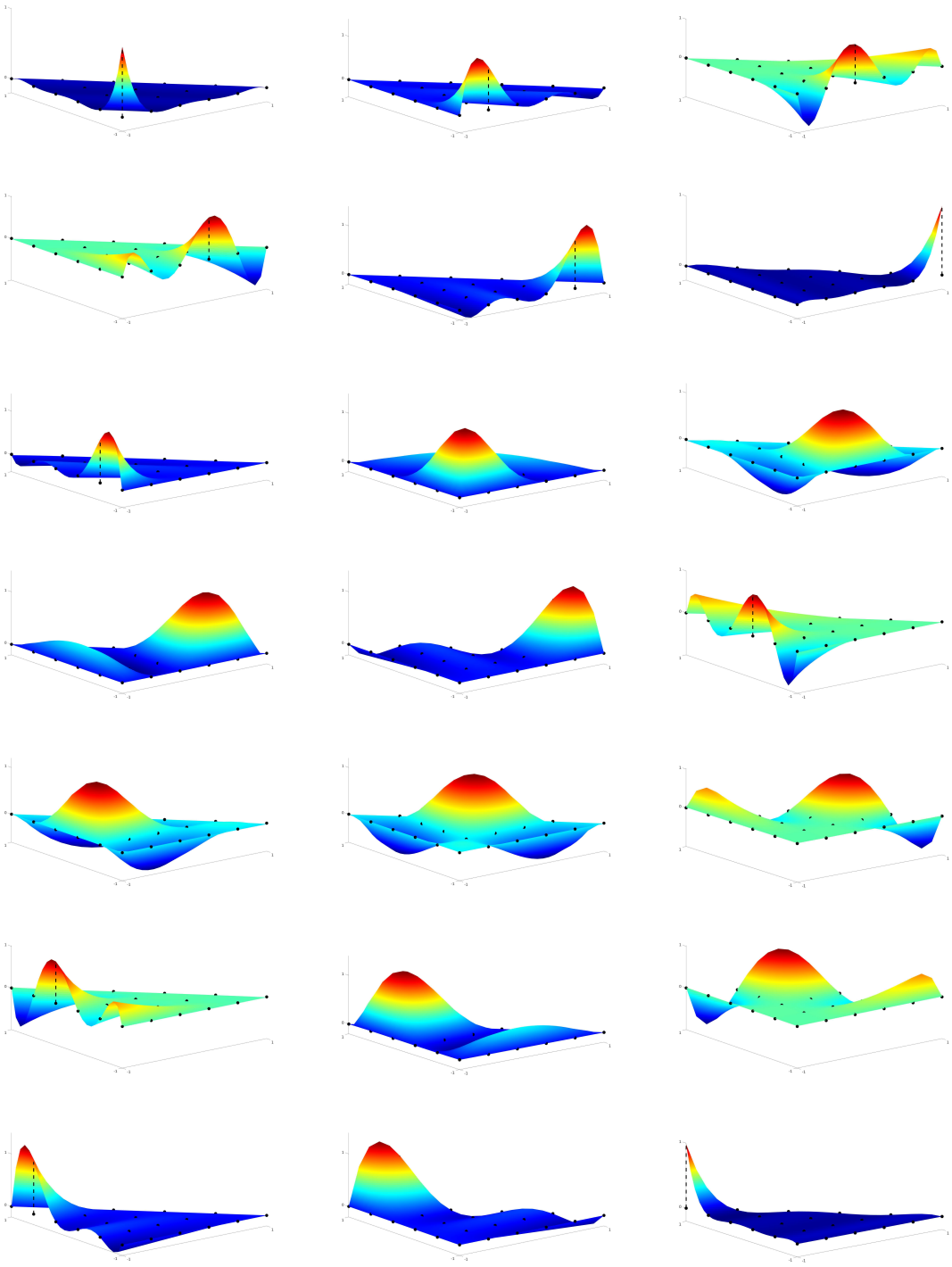


Fig. B.4: Lagrange polynomials for a 5^{th} order element.

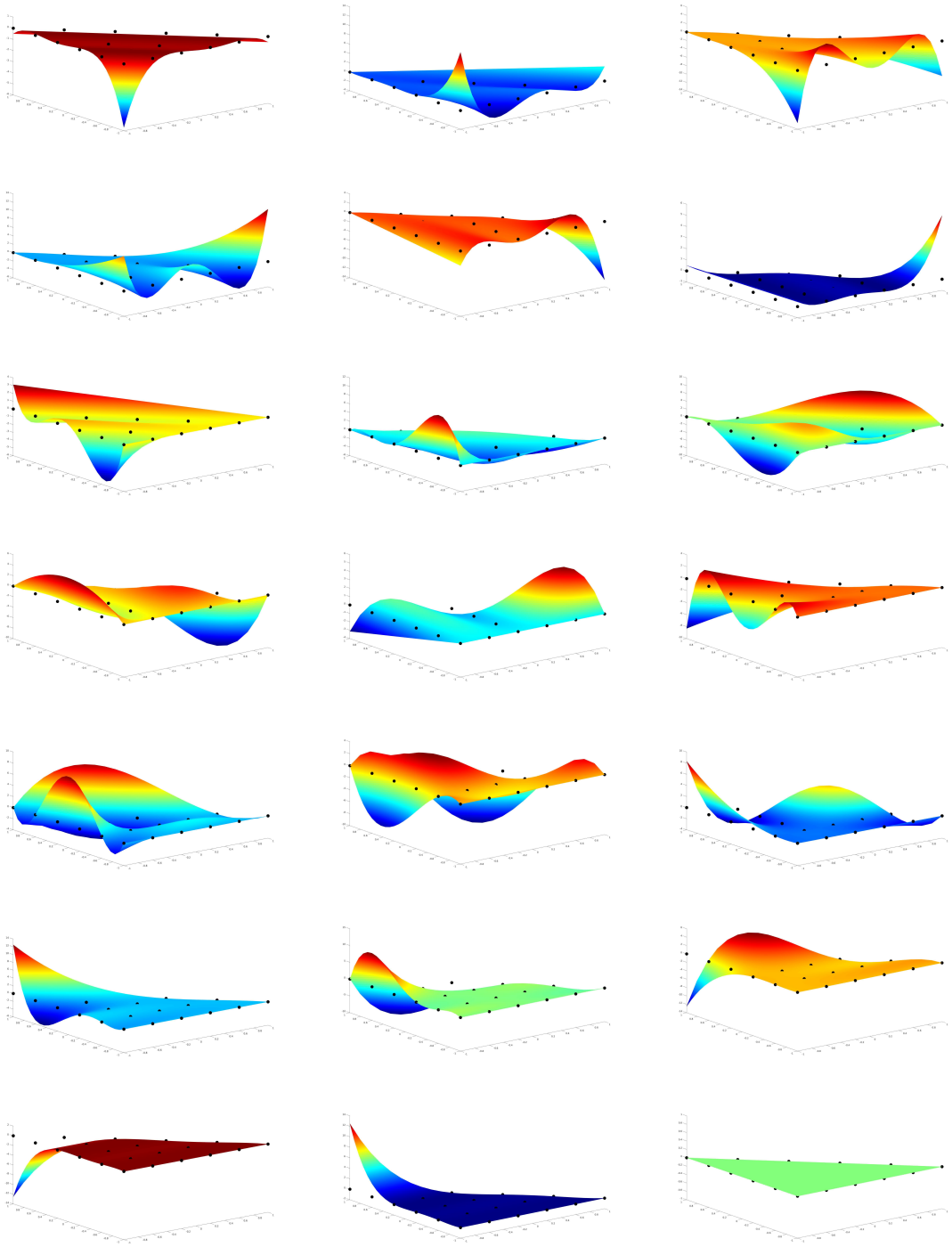


Fig. B.5: The derivative with respect to r of the Lagrange polynomials for a 5^{th} order element.

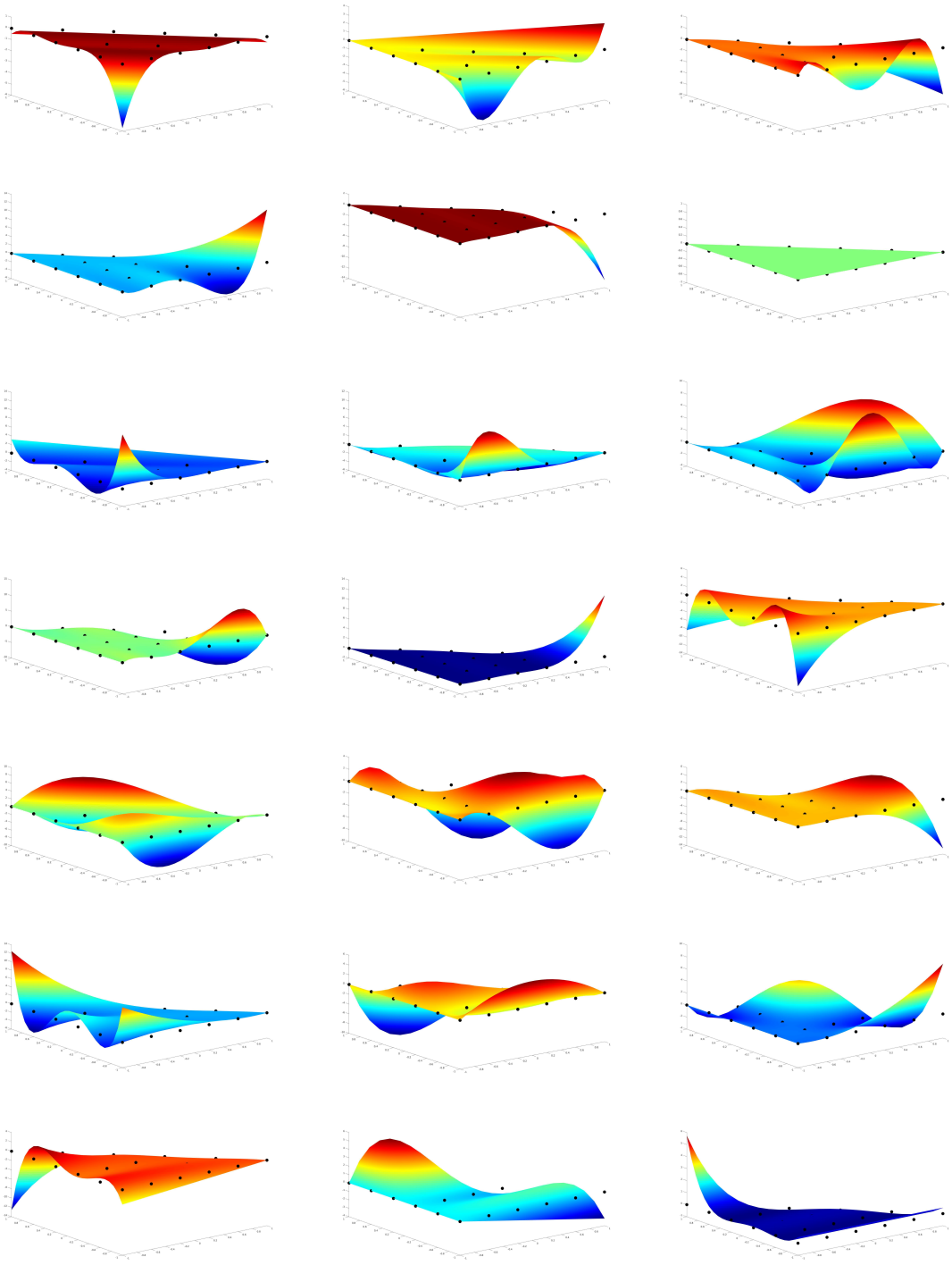


Fig. B.6: The derivative with respect to s of the Lagrange polynomials for a 5^{th} order element.

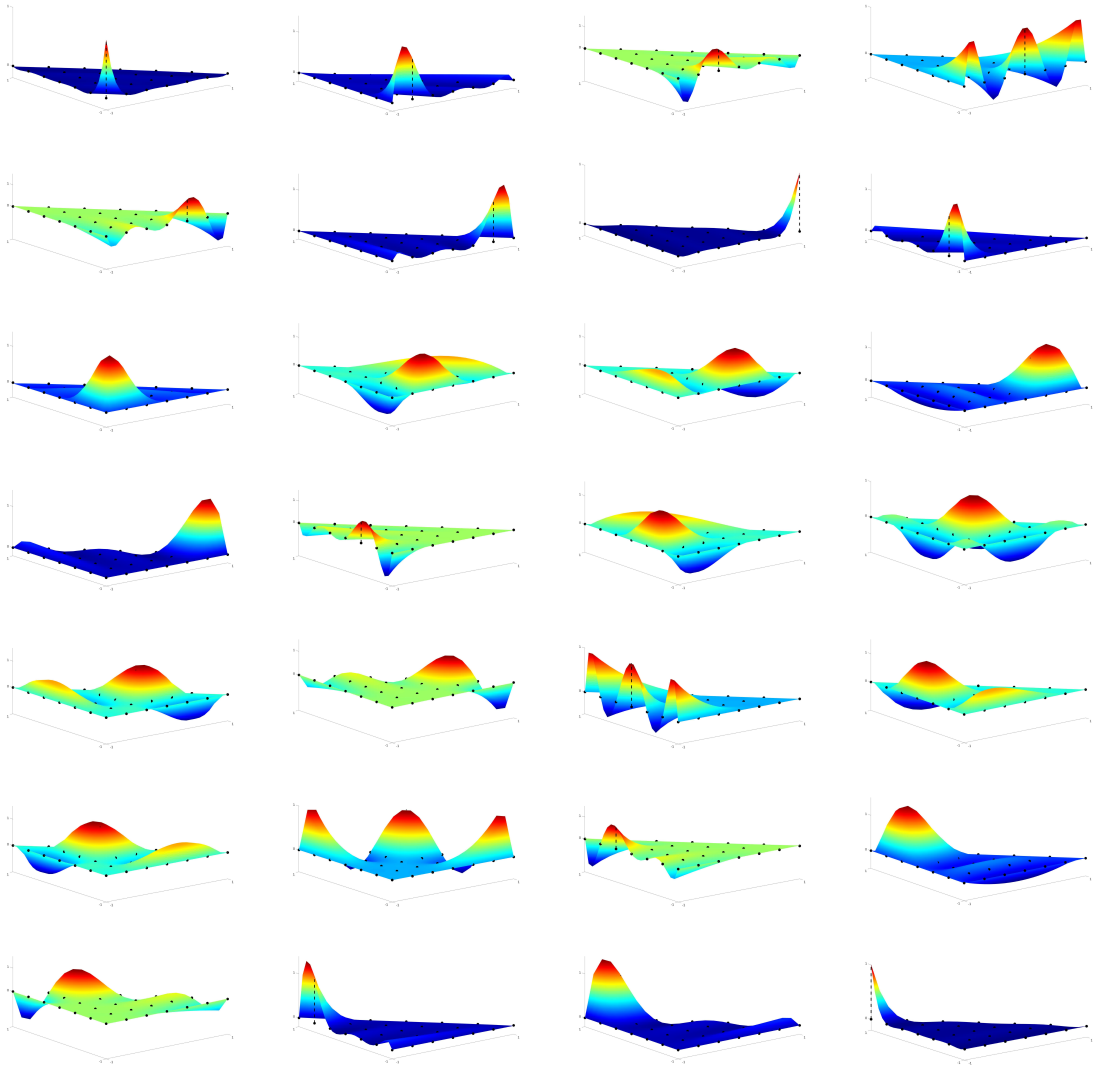


Fig. B.7: Lagrange polynomials for a 6^{th} order element.

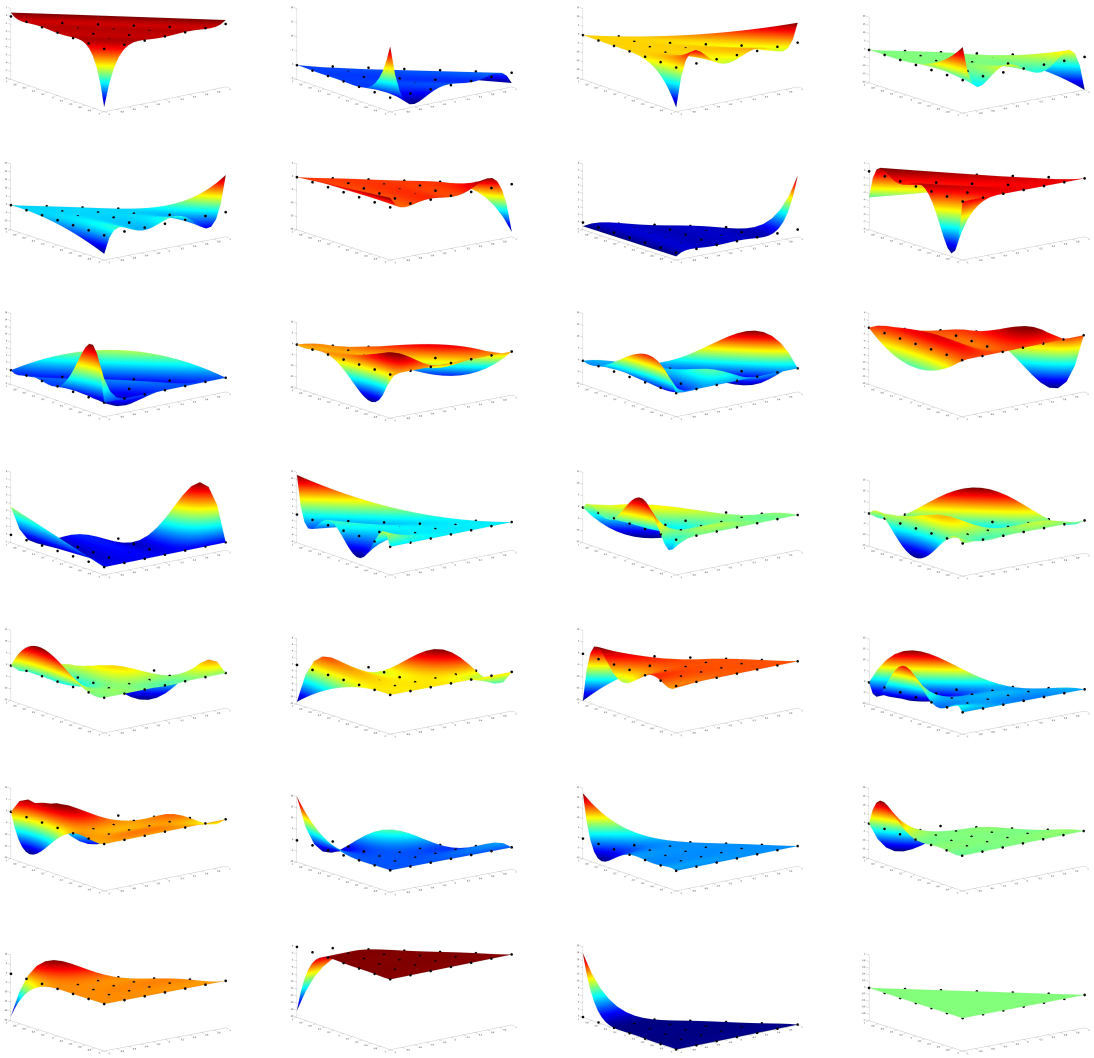


Fig. B.8: The derivative with respect to r of the Lagrange polynomials for a 6^{th} order element.

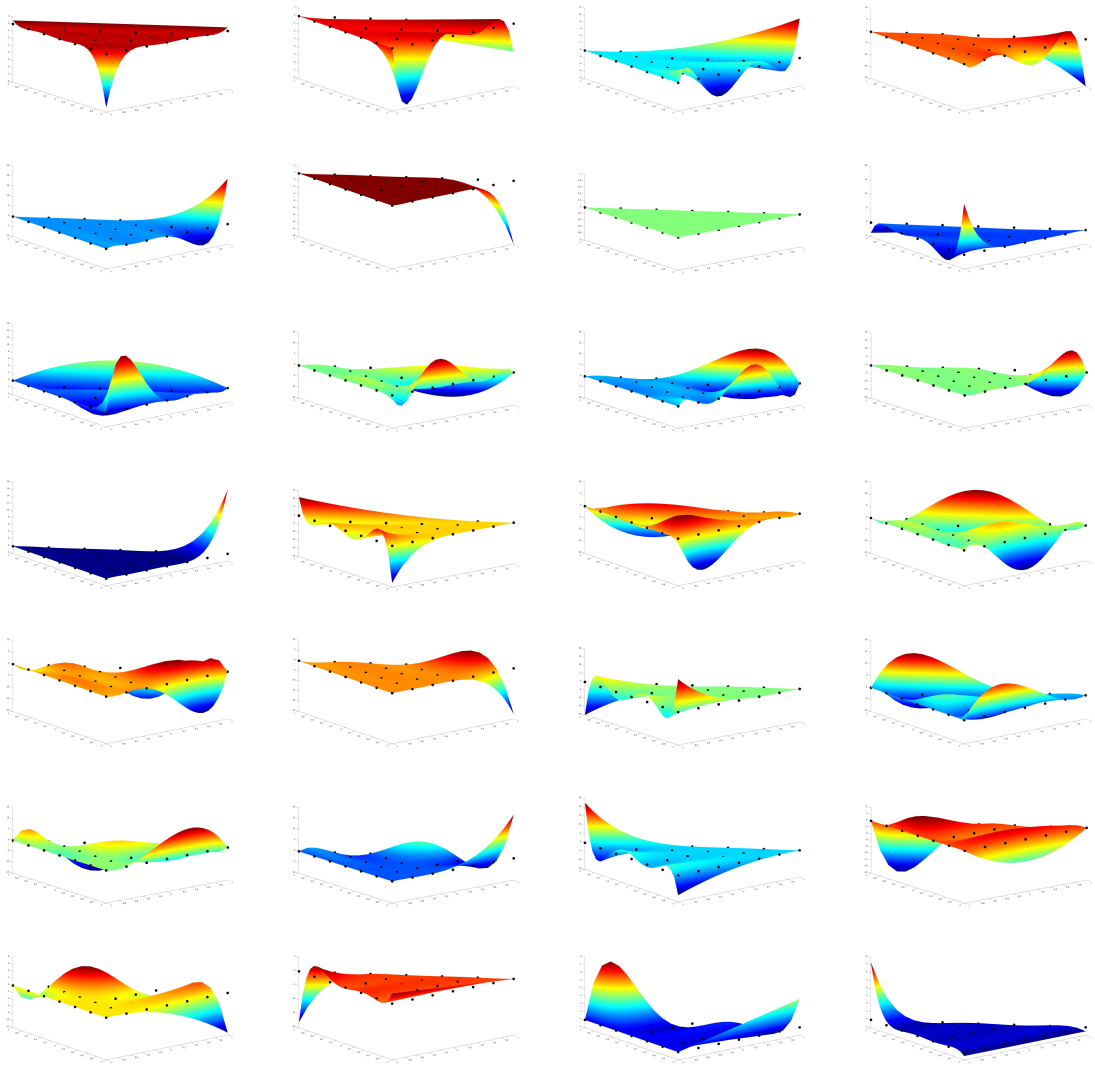


Fig. B.9: The derivative with respect to s of the Lagrange polynomials for a 6^{th} order element.

C

Derivation of Boundary Conditions

This appendix covers the derivation of the boundary conditions used in HDGM. The derivations beginning with (2.14) which we repeat below. We derive conditions for 3 types of boundary terminations; ABC, PEC, and IBC.

$$\hat{n} \times \vec{E}^{\wedge,sct} \triangleq$$
$$\frac{Y^-}{Y^- + Y^+}\hat{n} \times \vec{E}^{-,sct} + \frac{Y^+}{Y^- + Y^+}\hat{n} \times \vec{E}^{+,sct} - \frac{1}{Y^- + Y^+}\alpha\hat{n} \times \hat{n} \times (\vec{H}^{-,sct} - \vec{H}^{+,sct})$$

$$\hat{n} \times \vec{H}^{\wedge,sct} \triangleq$$
$$\frac{Z^-}{Z^- + Z^+}\hat{n} \times \vec{H}^{-,sct} + \frac{Z^+}{Z^- + Z^+}\hat{n} \times \vec{H}^{+,sct} + \frac{1}{Z^- + Z^+}\alpha\hat{n} \times \hat{n} \times (\vec{E}^{-,sct} - \vec{E}^{+,sct})$$

(2.14 revisited)

C.1 Silver-Müller Absorbing Boundary Condition

Assumption #1: All flux is outgoing $E^+ = 0, H^+ = 0$

$$\begin{aligned}\hat{n} \times \vec{H}^{\wedge,tot} &= \hat{n} \times \vec{H}^{-,tot} - \frac{Z^+}{Z^- + Z^+}\hat{n} \times \vec{H}^{-,tot} + \frac{1}{Z^- + Z^+}\hat{n} \times \hat{n} \times \vec{E}^{-,tot} \\ \hat{n} \times \vec{E}^{\wedge,tot} &= \hat{n} \times \vec{E}^{-,tot} - \frac{Y^+}{Y^- + Y^+}\hat{n} \times \vec{E}^{-,tot} - \frac{1}{Y^- + Y^+}\hat{n} \times \hat{n} \times \vec{H}^{-,tot}\end{aligned} \tag{C.1}$$

Assumption #2: Impedance is matched at the boundary $Z^+ = Z^-, Y^+ = Y^-$

$$\begin{aligned}\hat{n} \times \vec{H}^{\wedge,tot} &= \hat{n} \times \vec{H}^{-,tot} - \frac{1}{2}\hat{n} \times \vec{H}^{-,tot} + \frac{1}{2Z^-}\hat{n} \times \hat{n} \times \vec{E}^{-,tot} \\ &= \frac{1}{2}\hat{n} \times \vec{H}^{-,tot} + \frac{1}{2Z^-}\hat{n} \times \hat{n} \times \vec{E}^{-,tot}\end{aligned} \tag{C.2}$$

$$\begin{aligned}\hat{n} \times \vec{E}^{\wedge,tot} &= \hat{n} \times \vec{E}^{-,tot} - \frac{1}{2}\hat{n} \times \vec{E}^{-,tot} - \frac{1}{2Y^-}\hat{n} \times \hat{n} \times \vec{H}^{-,tot} \\ &= \frac{1}{2}\hat{n} \times \vec{E}^{-,tot} - \frac{1}{2}Z^-\hat{n} \times \hat{n} \times \vec{H}^{-,tot}\end{aligned}$$

From C.2 we can re-write the first equation as

$$\begin{aligned}Z^-\hat{n} \times \hat{n} \times \vec{H}^{\wedge,tot} &= \frac{1}{2}Z^-\hat{n} \times \hat{n} \times \vec{H}^{-,tot} + \frac{1}{2}\hat{n} \times \hat{n} \times \hat{n} \times \vec{E}^{-,tot} \\ \frac{1}{2}Z^-\hat{n} \times \hat{n} \times \vec{H}^{-,tot} &= Z^-\hat{n} \times \hat{n} \times \vec{H}^{\wedge,tot} - \frac{1}{2}\hat{n} \times \hat{n} \times \hat{n} \times \vec{E}^{-,tot}\end{aligned} \tag{C.3}$$

We now re-write and simplify the last equation from C.2 using C.3. We use the fact that $-\hat{n} \times \hat{n} \times \hat{n} \times A = \hat{n} \times A$

$$\begin{aligned}
\hat{n} \times \vec{E}^{\wedge,tot} &= \frac{1}{2}\hat{n} \times \vec{E}^{-,tot} - \frac{1}{2}Z^{-}\hat{n} \times \hat{n} \times \vec{H}^{-,tot} \\
\hat{n} \times \vec{E}^{\wedge,tot} &= \frac{1}{2}\hat{n} \times \vec{E}^{-,tot} - Z^{-}\hat{n} \times \hat{n} \times \vec{H}^{\wedge,tot} + \frac{1}{2}\hat{n} \times \hat{n} \times \hat{n} \times \vec{E}^{-,tot} \\
\hat{n} \times \vec{E}^{\wedge,tot} &= \frac{1}{2}\hat{n} \times \vec{E}^{-,tot} - Z^{-}\hat{n} \times \hat{n} \times \vec{H}^{\wedge,tot} - \frac{1}{2}\hat{n} \times \vec{E}^{-,tot} \\
\hat{n} \times \vec{E}^{\wedge,tot} &= -Z^{-}\hat{n} \times \hat{n} \times \vec{H}^{\wedge,tot}
\end{aligned} \tag{C.4}$$

From the true boundary definitions we can compute $\hat{n} \times \vec{E}^{\wedge,sct} = \hat{n} \times \vec{E}^{-,sct} - Z^{-}\hat{n} \times \hat{n} \times (\vec{H}^{-,sct} - \vec{H}^{\wedge,sct})$. We additionally know $\hat{n} \times \vec{E}^{\wedge,inc} = -Z^{-}\hat{n} \times \hat{n} \times \vec{H}^{\wedge,inc}$ since $\hat{n} \times \vec{E}^{\wedge,tot} = -Z^{-}\hat{n} \times \hat{n} \times \vec{H}^{\wedge,tot}$ would have to be imposed on any numerical incident field.

$$\begin{aligned}
&\hat{n} \times \vec{E}^{\wedge,sct} = \hat{n} \times \vec{E}^{\wedge,tot} - \hat{n} \times \vec{E}^{\wedge,inc} \\
&\hat{n} \times \vec{E}^{-,sct} - Z^{-}\hat{n} \times \hat{n} \times (\vec{H}^{-,sct} - \vec{H}^{\wedge,sct}) = -Z^{-}\hat{n} \times \hat{n} \times \vec{H}^{\wedge,tot} - (-Z^{-}\hat{n} \times \hat{n} \times H^{\wedge,inc}) \\
&\hat{n} \times \vec{E}^{-,sct} - Z^{-}\hat{n} \times \hat{n} \times (\vec{H}^{-,sct} - \vec{H}^{\wedge,sct}) = -Z^{-}\hat{n} \times \hat{n} \times \vec{H}^{\wedge,sct} \\
&\hat{n} \times \vec{E}^{-,sct} - Z^{-}\hat{n} \times \hat{n} \times (\vec{H}^{-,sct} - \vec{H}^{\wedge,sct}) + Z^{-}\hat{n} \times \hat{n} \times \vec{H}^{\wedge,sct} = 0 \\
&\hat{n} \times \vec{E}^{-,sct} - Z^{-}\hat{n} \times \hat{n} \times \vec{H}^{-,sct} + 2Z^{-}\hat{n} \times \hat{n} \times \vec{H}^{\wedge,sct} = 0
\end{aligned} \tag{C.5}$$

C.2 PEC Boundary Condition

We enforce the tangential electric field is 0 as PEC materials do not support a change in voltage. Assumption #1: All flux is outgoing $E^+ = 0, H^+ = 0$

$$\begin{aligned}\hat{n} \times \vec{H}^{\wedge,tot} &= \hat{n} \times \vec{H}^{-,tot} - \frac{Z^+}{Z^- + Z^+}\hat{n} \times \vec{H}^{-,tot} + \frac{1}{Z^- + Z^+}\hat{n} \times \hat{n} \times \vec{E}^{-,tot} \\ \hat{n} \times \vec{E}^{\wedge,tot} &= \hat{n} \times \vec{E}^{-,tot} - \frac{Y^+}{Y^- + Y^+}\hat{n} \times \vec{E}^{-,tot} - \frac{1}{Y^- + Y^+}\hat{n} \times \hat{n} \times \vec{H}^{-,tot}\end{aligned} \tag{C.6}$$

Assumption #2: The impedance of a PEC is 0 $Z^+ = 0, Y^+ = inf$

$$\begin{aligned}\hat{n} \times \vec{H}^{\wedge,tot} &= \hat{n} \times \vec{H}^{-,tot} + \frac{1}{Z^-}\hat{n} \times \hat{n} \times \vec{E}^{-,tot} \\ \hat{n} \times \vec{E}^{\wedge,tot} &= \hat{n} \times \vec{E}^{-,tot} - \hat{n} \times \vec{E}^{-,tot} = 0\end{aligned} \tag{C.7}$$

We now use the general boundary condition $\hat{n} \times \vec{E}^{\wedge,tot} = \hat{n} \times \vec{E}^{-,tot} - Z^-\hat{n} \times \hat{n} \times (\vec{H}^{-,tot} - \vec{H}^{\wedge,tot})$

$$\begin{aligned}&\hat{n} \times \vec{E}^{\wedge,tot} = 0 \\ &\hat{n} \times \vec{E}^{\wedge,tot} = \hat{n} \times \vec{E}^{-,tot} - Z^-\hat{n} \times \hat{n} \times (\vec{H}^{-,tot} - \vec{H}^{\wedge,tot}) \\ &0 = \hat{n} \times \vec{E}^{-,tot} - Z^-\hat{n} \times \hat{n} \times (\vec{H}^{-,tot} - \vec{H}^{\wedge,tot}) \\ &0 = \hat{n} \times (\vec{E}^{-,inc} + \vec{E}^{-,sct}) - Z^-\hat{n} \times \hat{n} \times (\vec{H}^{-,inc} + \vec{H}^{-,sct} - \vec{H}^{\wedge,int} - \vec{H}^{\wedge,sct})\end{aligned} \tag{C.8}$$

Since $-\hat{n} \times \vec{H}^{\wedge,inc} = -\hat{n} \times \vec{H}^{-,inc}$ and $-\hat{n} \times \vec{E}^{\wedge,inc} = -\hat{n} \times \vec{E}^{-,inc}$ we can write this as

$$\begin{aligned}0 &= \hat{n} \times (\vec{E}^{-,inc} + \vec{E}^{-,sct}) - Z^-\hat{n} \times \hat{n} \times (\vec{H}^{-,sct} - \vec{H}^{\wedge,sct}) \\ 0 &= \hat{n} \times (\vec{E}^{-,inc} + \vec{E}^{-,sct}) - Z^-\hat{n} \times \hat{n} \times (\vec{H}^{-,sct} - \vec{H}^{\wedge,sct}) \\ \hat{n} \times \vec{E}^{-,sct} &- Z^-\hat{n} \times \hat{n} \times (\vec{H}^{-,sct} - \vec{H}^{\wedge,sct}) = -\hat{n} \times \vec{E}^{-,inc}\end{aligned} \tag{C.9}$$

C.3 Impedance Boundary Condition

On an impedance boundary we enfoce $\hat{n} \times \vec{E}^{\wedge,tot} = Z_{boundary} \hat{n} \times \hat{n} \times \vec{H}^{\wedge,tot}$ for a given impedance $Z_{boundary}$. We assume the impedance Z^+ and admitance Y^+ of the neighbouring medium is known and that all flux is outgoing ($E^+ = H^+ = 0$).

$$\hat{n} \times \vec{H}^{\wedge,tot} = \hat{n} \times \vec{H}^{-,tot} - \frac{Z_{boundary}}{Z^- + Z_{boundary}} \hat{n} \times \vec{H}^{-,tot} + \frac{1}{Z^- + Z_{boundary}} \hat{n} \times \hat{n} \times \vec{E}^{-,tot} \tag{C.10}$$

We use this with $\hat{n} \times \vec{H}^{\wedge,inc} = \hat{n} \times \vec{H}^{-,inc}$ to obtain the following:

$$
\begin{aligned}
& -(Z^- + Z_{boundary})\hat{n} \times \vec{H}^{\wedge,tot} + (Z^- + Z_{boundary})\hat{n} \times \vec{H}^{-,tot} - Z_{boundary}\hat{n} \times \vec{H}^{-,tot} \\
& \quad + \hat{n} \times \hat{n} \times \vec{E}^{-,tot} = 0 \\
\\
\Rightarrow \quad & -(Z^- + Z_{boundary})\hat{n} \times \vec{H}^{\wedge,tot} + Z^-\hat{n} \times \vec{H}^{-,tot} + \hat{n} \times \hat{n} \times \vec{E}^{-,tot} = 0 \\
\\
\Rightarrow \quad & -(Z^- + Z_{boundary})\hat{n} \times \vec{H}^{\wedge,sct} - (Z^- + Z_{boundary})\hat{n} \times \vec{H}^{-,inc} + Z^-\hat{n} \times \vec{H}^{-,inc} \\
& \quad + Z^-\hat{n} \times \vec{H}^{-,sct} + \hat{n} \times \hat{n} \times \vec{E}^{-,inc} + \hat{n} \times \hat{n} \times \vec{E}^{-,sct} = 0 \\
\\
\Rightarrow \quad & -(Z^- + Z_{boundary})\hat{n} \times \vec{H}^{\wedge,sct} - Z_{boundary}\hat{n} \times \vec{H}^{-,inc} + Z^-\hat{n} \times \vec{H}^{-,sct} \\
& \quad + \hat{n} \times \hat{n} \times \vec{E}^{-,inc} + \hat{n} \times \hat{n} \times \vec{E}^{-,sct} = 0 \\
\\
\Rightarrow \quad & -(Z^- + Z_{boundary})\hat{n} \times \vec{H}^{\wedge,sct} + Z^-\hat{n} \times \vec{H}^{-,sct} + \hat{n} \times \hat{n} \times \vec{E}^{-,sct} \\
& \quad = Z_{boundary}\hat{n} \times \vec{H}^{-,inc} - \hat{n} \times \hat{n} \times \vec{E}^{-,inc} \\
\Rightarrow \quad & -(Z^- + Z_{boundary})\hat{n} \times \hat{n} \times \vec{H}^{\wedge,sct} + Z^-\hat{n} \times \hat{n} \times \vec{H}^{-,sct} - \hat{n} \times \vec{E}^{-,sct} \\
& \quad = Z_{boundary}\hat{n} \times \hat{n} \times \vec{H}^{-,inc} + \hat{n} \times \vec{E}^{-,inc}
\end{aligned}
\tag{C.11}
$$

C.4 Silver-Müller Absorbing Boundary Condition for Electric field formulation

We use the same assumptions as the previous ABC condition and begin with the total field below.

$$\hat{n} \times \vec{H}^{\wedge,sct} = \hat{n} \times \vec{H}^{\wedge,tot} - \hat{n} \times \vec{H}^{\wedge,inc} \tag{C.12}$$

We use (C.4) to obtain the following relationship.

$$\hat{n} \times \vec{H}^{\wedge,tot} = Y^{-} \hat{n} \times \hat{n} \times \vec{E}^{\wedge,tot} \tag{C.13}$$

As well as (5.21) to obtain:

$$\hat{n} \times \vec{H}^{\wedge,sct} = \hat{n} \times \vec{H}^{-,sct} - Y^{-} \hat{n} \times \hat{n} \times (\vec{E}^{\wedge,sct} - \vec{E}^{-,sct}) \tag{C.14}$$

We use these equations in the total field formulation to write

$$\hat{n} \times \vec{H}^{-,sct} - Y^{-} \hat{n} \times \hat{n} \times (\vec{E}^{\wedge,sct} - \vec{E}^{-,sct}) = Y^{-} \hat{n} \times \hat{n} \times \vec{E}^{\wedge,tot} - Y^{-} \hat{n} \times \hat{n} \times \vec{E}^{\wedge,inc} \tag{C.15}$$

We simplify this and replace $\hat{n} \times \hat{n} \times \vec{E}^{\wedge,sct}$ with $\vec{\Lambda}_E^t$.

$$\begin{aligned} \hat{n} \times \vec{H}^{-,sct} + Y^{-} \hat{n} \times \hat{n} \times \vec{E}^{-,sct} + Y^{-} \vec{\Lambda}_E^t &= -Y^{-} \vec{\Lambda}_E^t \\ \hat{n} \times \vec{H}^{-,sct} + Y^{-} \hat{n} \times \hat{n} \times \vec{E}^{-,sct} + 2Y^{-} \vec{\Lambda}_E^t &= 0 \end{aligned} \tag{C.16}$$

C.5 Impedance Boundary Condition for Electric field formulation

Once again we enfoce $\hat{n} \times \vec{E}^{\wedge,tot} = Z_{boundary} \hat{n} \times \hat{n} \times \vec{H}^{\wedge,tot}$ for a given impedance $Z_{boundary}$. We assume the impedance Z^+ and admitance Y^+ of the neighbouring medium is known and that all flux is outgoing ($E^+ = H^+ = 0$).

$$\hat{n} \times \vec{E}^{\wedge,tot} = \hat{n} \times \vec{E}^{-,tot} - \frac{Y_{boundary}}{Y^- + Y_{boundary}} \hat{n} \times \vec{E}^{-,tot} - \frac{1}{Y^- + Y_{boundary}} \hat{n} \times \hat{n} \times \vec{H}^{-,tot} \quad \text{(C.17)}$$

We use this with $\hat{n} \times \vec{E}^{\wedge,inc} = \hat{n} \times \vec{E}^{-,inc}$ to obtain the following:

$$
\begin{aligned}
&- \hat{n} \times \vec{E}^{\wedge,sct} - \hat{n} \times \vec{E}^{-,inc} + \hat{n} \times \vec{E}^{-,tot} - \frac{Y_{boundary}}{Y^- + Y_{boundary}} \hat{n} \times \vec{E}^{-,tot} \\
&\quad - \frac{1}{Y^- + Y_{boundary}} \hat{n} \times \hat{n} \times \vec{H}^{-,tot} = 0 \\
&\Rightarrow -(Y^- + Y_{boundary}) \hat{n} \times \vec{E}^{\wedge,sct} + (Y^- + Y_{boundary}) \hat{n} \times \vec{E}^{-,sct} - Y_{boundary} \hat{n} \times \vec{E}^{-,tot} \\
&\quad - \hat{n} \times \hat{n} \times \vec{H}^{-,tot} = 0 \\
&\Rightarrow -(Y^- + Y_{boundary}) \hat{n} \times \hat{n} \times \vec{E}^{\wedge,sct} + (Y^- + Y_{boundary}) \hat{n} \times \hat{n} \times \vec{E}^{-,sct} \\
&\quad - Y_{boundary} \hat{n} \times \hat{n} \times \vec{E}^{-,tot} - \hat{n} \times \hat{n} \times \hat{n} \times \vec{H}^{-,tot} = 0 \\
&\Rightarrow (Y^- + Y_{boundary}) \vec{\Lambda}_E^t + (Y^- + Y_{boundary}) \hat{n} \times \hat{n} \times \vec{E}^{-,sct} \\
&\quad + \hat{n} \times \vec{H}^{-,tot} = Y_{boundary} \hat{n} \times \hat{n} \times \vec{E}^{-,tot} \\
&\Rightarrow (Y^- + Y_{boundary}) \vec{\Lambda}_E^t + Y^- \hat{n} \times \hat{n} \times \vec{E}^{-,sct} + \hat{n} \times \vec{H}^{-,sct} \\
&\quad = Y_{boundary} \hat{n} \times \hat{n} \times \vec{E}^{-,inc} - \hat{n} \times \vec{H}^{-,inc}
\end{aligned}
\tag{C.18}
$$

References

[1] A. Zakaria, A. Baran, and J. Lovetri, "Estimation and use of prior information in fem-csi for biomedical microwave tomography," *IEEE Antennas and Wireless Propagation Letters*, vol. 11, pp. 1606–1609, 2012.

[2] A. Zakaria, I. Jeffrey, J. LoVetri, and A. Zakaria, "Full-vectorial parallel finite-element contrast source inversion method," *Progress In Electromagnetics Research*, vol. 142, pp. 463–483, 2013.

[3] A. Baran, D. Kurrant, A. Zakaria, E. Fear, and J. LoVetri, "Breast imaging using microwave tomography with radar-based tissue-regions estimation," *Progress in Electromagnetics Research*, vol. 149, pp. 161–171, 2014.

[4] K. Nemez, "A faceted chamber for magnetic field microwave breast imaging," *Masters thesis, University of Manitoba*, 2017.

[5] K. Brown, "Implementation of parallel 3-d dgm-csi and dgm-gni for electromagnetic imaging applications," *Masters thesis, University of Manitoba*, 2019.

[6] M. Asefi, I. Jeffrey, J. LoVetri, C. Gilmore, P. Card, and J. Paliwal, "Grain bin monitoring via electromagnetic imaging," *Computers and Electronics in Agriculture*, vol. 119, pp. 133–141, 2015.

[7] J. C. Maxwell, *A treatise on electricity and magnetism*. Oxford: Clarendon Press, 1873, vol. 1.

[8] ——, "Viii. a dynamical theory of the electromagnetic field," *Philosophical transactions of the Royal Society of London*, no. 155, pp. 459–512, 1865.

[9] J. A. Stratton, *Electromagnetic theory*. John Wiley & Sons, 2007, vol. 33.

[10] J.-M. Jin, *Theory and computation of electromagnetic fields*. John Wiley & Sons, 2011.

[11] C. Geuzaine and J.-F. Remacle, "Gmsh: A 3-d finite element mesh generator with built-in pre-and post-processing facilities," *International journal for numerical methods in engineering*, vol. 79, no. 11, pp. 1309–1331, 2009.

[12] G. A. Deschamps, "Electromagnetics and differential forms," *Proceedings of the IEEE*, vol. 69, no. 6, pp. 676–696, 1981.

[13] J. S. Hesthaven and T. Warburton, *Nodal discontinuous Galerkin methods: algorithms, analysis, and applications*. Springer Science & Business Media, 2007.

[14] E. Simsek, J. Liu, and Q. H. Liu, "A spectral integral method and hybrid sim/fem for layered media," *IEEE transactions on microwave theory and techniques*, vol. 54, no. 11, pp. 3878–3884, 2006.

[15] M. Ainsworth and B. Senior, "Aspects of an adaptive hp-finite element method: Adaptive strategy, conforming approximation and efficient solvers," *Computer Methods in Applied Mechanics and Engineering*, vol. 150, no. 1-4, pp. 65–87, 1997.

[16] B. Guo and I. Babuška, "The hp version of the finite element method," *Computational Mechanics*, vol. 1, no. 1, pp. 21–41, 1986.

[17] T. C. Warburton, S. J. Sherwin, and G. E. Karniadakis, "Basis functions for triangular and quadrilateral high-order elements," *SIAM Journal on Scientific Computing*, vol. 20, no. 5, pp. 1671–1695, 1999.

[18] G. Nikishkov, "Introduction to the finite element method," *University of Aizu*, pp. 1–70, 2004.

[19] A. Galántai, "The theory of newton's method," *Journal of Computational and Applied Mathematics*, vol. 124, no. 1-2, pp. 25–44, 2000.

[20] L. Zhang, T. Cui, and H. Liu, "A set of symmetric quadrature rules on triangles and tetrahedra," *Journal of Computational Mathematics*, pp. 89–96, 2009.

[21] I. Jeffrey, N. Geddert, K. Brown, and J. LoVetri, "The time-harmonic discontinuous galerkin method as a robust forward solver for microwave imaging applications," *Progress In Electromagnetics Research*, vol. 154, pp. 1–21, 2015.

[22] J. S. Hesthaven and T. Warburton, "High-order nodal discontinuous galerkin methods for the maxwell eigenvalue problem," *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, vol. 362, no. 1816, pp. 493–524, 2004.

[23] R. F. Harrington, "Time-harmonic," *Electromagnetic Fields*, pp. 168–171, 2001.

[24] N. C. Nguyen, J. Peraire, and B. Cockburn, "An implicit high-order hybridizable discontinuous galerkin method for nonlinear convection–diffusion equations," *Journal of Computational Physics*, vol. 228, no. 23, pp. 8841–8855, 2009.

[25] B. Cockburn, J. Gopalakrishnan, and R. Lazarov, "Unified hybridization of discontinuous galerkin, mixed, and continuous galerkin methods for second order elliptic problems," *SIAM Journal on Numerical Analysis*, vol. 47, no. 2, pp. 1319–1365, 2009.

[26] B. Cockburn, B. Dong, and J. Guzmán, "A superconvergent ldg-hybridizable galerkin method for second-order elliptic problems," *Mathematics of Computation*, vol. 77, no. 264, pp. 1887–1916, 2008.

[27] J. Peraire, N. Nguyen, and B. Cockburn, "A hybridizable discontinuous galerkin method for the compressible euler and navier-stokes equations," in *48th AIAA Aerospace Sciences meeting including the New Horizons forum and aerospace exposition*, 2010, p. 363.

[28] N. C. Nguyen, J. Peraire, and B. Cockburn, "High-order implicit hybridizable discontinuous galerkin methods for acoustics and elastodynamics," *Journal of Computational Physics*, vol. 230, no. 10, pp. 3695–3718, 2011.

[29] W. Gropp, E. Lusk, N. Doss, and A. Skjellum, "A high-performance, portable implementation of the mpi message passing interface standard," *Parallel computing*, vol. 22, no. 6, pp. 789–828, 1996.

[30] S. Balay, S. Abhyankar, M. F. Adams, J. Brown, P. Brune, K. Buschelman, L. Dalcin, A. Dener, V. Eijkhout, W. D. Gropp, D. Karpeyev, D. Kaushik, M. G. Knepley, D. A. May, L. C. McInnes, R. T. Mills, T. Munson, K. Rupp, P. Sanan, B. F. Smith, S. Zampini, H. Zhang, and H. Zhang, "PETSc users manual," Argonne National Laboratory, Tech. Rep. ANL-95/11 - Revision 3.14, 2020. [Online]. Available: https://www.mcs.anl.gov/petsc

[31] P. Amestoy, I. S. Duff, J. Koster, and J.-Y. L'Excellent, "A fully asynchronous multifrontal solver using distributed dynamic scheduling," *SIAM Journal on Matrix Analysis and Applications*, vol. 23, no. 1, pp. 15–41, 2001.

[32] E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen, *LAPACK Users' Guide*, 3rd ed. Philadelphia, PA: Society for Industrial and Applied Mathematics, 1999.

[33] E. G. Boman, U. V. Catalyurek, C. Chevalier, and K. D. Devine, "The Zoltan and Isorropia parallel toolkits for combinatorial scientific computing: Partitioning, ordering, and coloring," *Scientific Programming*, vol. 20, no. 2, pp. 129–150, 2012.

[34] E. Keeley, N. Geddert, K. Krakalovich, R. Kruk, J. LoVetri, C. Gilmore, and I. Jeffrey, "Stored grain inventory management using neural-network-based parametric electromagnetic inversion," *IEEE Access*, p. "to appear", 2020.

[35] P. M. Van Den Berg and R. E. Kleinman, "A contrast source inversion method," *Inverse problems*, vol. 13, no. 6, p. 1607, 1997.

[36] M. Hopfer, R. Planas, A. Hamidipour, T. Henriksson, and S. Semenov, "Electromagnetic tomography for detection, differentiation, and monitoring of brain stroke: A virtual data and human head phantom study." *IEEE Antennas and Propagation Magazine*, vol. 59, no. 5, pp. 86–97, 2017.

[37] C. Gilmore, A. Zakaria, S. Pistorius, and J. LoVetri, "Microwave imaging of human forearms: Pilot study and image enhancement," *International journal of biomedical imaging*, vol. 2013, 2013.

[38] G. Bellizzi, O. M. Bucci, and I. Catapano, "Microwave cancer imaging exploiting magnetic nanoparticles as contrast agent," *IEEE Transactions on Biomedical Engineering*, vol. 58, no. 9, pp. 2528–2536, 2011.

[39] A. Abubakar and P. M. Van Den Berg, "Non-linear three-dimensional inversion of cross-well electrical measurements," *Geophysical prospecting*, vol. 48, no. 1, pp. 109–134, 2001.

[40] M. Asefi, I. Jeffrey, J. LoVetri, C. Gilmore, P. Card, and J. Paliwal, "Grain bin monitoring via electromagnetic imaging," *Computers and Electronics in Agriculture*, vol. 119, pp. 133–141, 2015.

[41] I. Jeffrey, J. LoVetri, K. Brown, N. Geddert, M. Asefi, K. Nemez, A. Baran, and G. Faucher, "3d pde-based contrast source inversion for biomedical and agricultural applications," in *XXXIIth URSI General Assembly and Scientific Symposium (URSI GASS)*, 2017.

[42] A. Zakaria, "The finite-element contrast source inversion method for microwave imaging applications," *Ph.D. dissertation, University of Manitoba*, 2012.

[43] K. G. Brown, N. Geddert, M. Asefi, J. LoVetri, and I. Jeffrey, "Hybridizable discontinuous galerkin method contrast source inversion of 2-d and 3-d dielectric and magnetic targets," *IEEE Transactions on Microwave Theory and Techniques*, vol. 67, no. 5, pp. 1766–1777, 2019.

[44] K. Brown, N. Geddert, G. Faucher, A. Baran, M. Asefi, K. Nemez, J. LoVetri, and I. Jeffrey, "3d microwave imaging using the time-harmonic discontinuous galerkin method-contrast source inversion," in *USNC-URSI Radio Science Meeting, San Diego, California*, 2017.

[45] N. Geddert, L. Sparling, J. LoVetri, and I. Jeffrey, "Frequency-weighting for multi-frequency electromagnetic source contrast inversion," in *2018 International Workshop on Computing, Electromagnetics, and Machine Intelligence (CEMi)*. IEEE, 2018, pp. 15–16.

[46] N. Geddert and I. Jeffrey, "A dynamically balanced openmp-cuda implementation of pde-based contrast source inversion for microwave imaging," in *2018 18th International*

Symposium on Antenna Technology and Applied Electromagnetics (ANTEM). IEEE, 2018, pp. 1–2.

[47] W. Gibson, *Scattered Field of a Conducting and Stratified Spheres*, 2020. [Online]. Available: https://www.mathworks.com/matlabcentral/fileexchange/20430-scattered-field-of-a-conducting-and-stratified-spheres

[48] C. Gilmore, I. Jeffrey, M. Asefi, N. T. Geddert, K. G. Brown, and J. Lovetri, "Phaseless parametric inversion for system calibration and obtaining prior information," *IEEE Access*, vol. 7, pp. 128 735–128 745, 2019.

[49] L. Li, S. Lanteri, and R. Perrussel, "A hybridizable discontinuous galerkin method combined to a schwarz algorithm for the solution of 3d time-harmonic maxwells equation," *Journal of Computational Physics*, vol. 256, pp. 563–581, 2014.

[50] D. Baumann, "A 3-d numerical field solver based on the finite-volume time-domain method," Ph.D. dissertation, ETH Zurich, 2006.