Spring 2015

Isogeometric Analysis for Electromagnetism

Tahsin Khajah
Old Dominion University

ISOGEOMETRIC ANALYSIS FOR ELECTROMAGNETISM

by

Tahsin Khajah
M.Sc. Sharif University of Technology, Tehran, Iran

A Dissertation Submitted to the Faculty of
Old Dominion University in Partial Fulfillment of the
Requirements for the Degree of

DOCTOR OF PHILOSOPHY

DEPARTMENT OF MECHANICAL AND AEROSPACE ENIGNEERING

OLD DOMINION UNIVERSITY
May 2015

Approved by:

Gene Hou (Director)

Sebastian Bawab (Member)

Michel Audette (Member)

Miltiadis Kotinis (Member)

Stacie Ringleb (Member)

ABSTRACT

ISOGEOMETRIC ANALYSIS FOR ELECTROMAGNETISM

Tahsin Khajah
Old Dominion University, 2015
Director: Dr. Gene Hou

The combination of numerical analysis with the scanning technology has been seeing increased use in many research areas. There is an emerging need for high-fidelity geometric modeling and meshing for practical applications. The Isogeometric Analysis (IGA) is a comprehensive computational framework, which integrates geometric modeling and meshing with analysis. Different from other existing numerical methods, the IGA can generate analysis ready models without loss of geometrical accuracy. In IGA, the continuity and the quality of a solution can be conveniently controlled and refined. These features enable IGA to integrate modeling, analysis, and design in a unified framework, the root idea of IGA. The IGA for electromagmetics is studied here for steady and transient electromagnetics as well as electromagnetic scattering. The solution procedure and the associated Matlab codes are developed to simulate the electromagnetic radiation on a biological tissues. The scattered and the total electrical fields are computed over the complex geometry of a brain section with realistic material properties. A perfectly matched layer (PML) is developed to model the far field boundary condition. The IGA platform developed here offers a reliable simulation within an accurate representation of the geometry. The results of this research can be used both in evaluating the potential health and safety risks of electromagnetic radiations and in optimizing the design of radiating devices used in non-invasive diagnostics and therapies.

TO MY FAMILY

ACKNOWLEDGMENTS

I am deeply in debt to my advisor, Dr. Hou, for providing me with invaluable guidance and encouragement throughout my graduate studies. I greatly appreciate his confidence in my work and scholarly advice. His patience and encouragement have played a fundamental role in the completion of this work in the present form.

I would also like to thank Dr. Bawab for his support, interest, and time serving on the dissertation committee. I greatly acknowledge the funding sources that my doctoral dissertation possible.

Many thanks to Dr. Audette for providing me with insight and helping me stay encouraged about this research and its possible biomedical applications. I am grateful to Dr. Xiao who supported me during tough times and helped me set up the electromagnetic radiation problem. I would like to thank Dr. Kotinis and Dr. Ringleb for the time serving on the dissertation committee and for their scholarly advice and encouragement.

I consider myself lucky to have been mentored by these experts in different fields. With their help, I have gained a unique skill set that can connect advanced numerical methods, electromagnetism and geometrical modeling.

Lastly, I would like to thank my family for their unconditional love and support. To my mother and father, thank you both for opening your arms for me when the world closed its doors. Many thanks to my wife who understood me, encouraged me, and stood by me. My brothers and beloved sister, I am grateful for your heartwarming support.

TABLE OF CONTENTS

Page

Chapter Page

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Studying the effects of electromagnetic radiation (EM) on biological media has attracted a great interest in literature. There are two main objectives in such a study. The first objective is to evaluate the potential health and safety risks of electromagnetic radiation and to set standards for safe exposures. Examples of this case are human body exposed to radiation from cellular phones, imaging devices, wireless Local Area Networks (LANs), etc. Specifically the safety of imaging devices for patients with metallic implants is not fully regulated and under investigation. The second objective is the development and utilization of electromagnetic radiation devices for medical applications such as non-invasive diagnostics and therapies.

One of the therapeutic applications is the hyperthermia treatment which is used to shrink the tumor size by heating the cancer cells. Electromagnetic radiation can generate the required heat inside the body. However, focusing the electromagnetic field on a specific area without damaging the nearby cells is far from trivial. Biological tissues including the brain layers, muscle, fat, blood, and organ tissues have relatively high dielectric constants and conductivities. Therefore, the radiated electromagnetic field power is absorbed and the electromagnetic wave is attenuated exponentially inside the tissues. Hence, delivering electromagnetic energy to deep seated tissue/tumor remains a difficult problem. Another challenge is the possibility of generating standing waves during radiation which can results in *hot spots* [35].

Standing waves are generated from the scattered fields of the interface between the material layers [68] and can lead to a deep burn of the tissue. Since nerve cells are not distributed evenly throughout the body, the burning tissue might not even be felt by the patient. As a result, reliable electromagnetic analysis on biological tissues demands high fidelity model generation so that it can predict the hot spots. Since the circulating blood can distribute the generated heat very effectively, avoiding hot spots inside the brain is more critical. Consequently, an accurate patient specific model is required for brain radiation analysis.

Another therapeutic application is brain stimulation. A customizable model providing a detailed geometric representation of the brain leads to a precise evaluation of the electrical field generated inside the brain. In *Repetitive Transcranial Magnetic Stimulation (rTMS)* a magnet is used to activate the outer layers of the brain by sending short electromagnetic pulses. These pulses will create small electrical currents in the targeted area of the brain witch stimulate the nerve cells. rTMS which can roughly target a specific part of the brain is approved by FDA in 2008 for treatment of major depression [32,45,49] only after other treatments fails to improve the patient conditions. This treatment is rather new and therefore with unknown long-term side effects. Common side effects include contraction of the scalp, jaw and face muscles, mild headaches, and brief lightheadedness.

The possibility of precisely estimating the electromagnetic radiations on high fidelity models can be used in both optimal designing of cell phones and other mobile

devices as well as minimization of the undesirable effects of radiation during diagnostic and therapeutic applications. To achieve this, realistic models of the brain should be generated. Then the electromagnetic fields should be estimated within the resulting hi-fidelity models. Furthermore, the analysis method should provide an automatic mechanism to refine the the solution in oder to achieve numerical results with acceptable accuracy. The common methods to numerically solve the governing equations related to electromagnetic fields are discussed next.

1.1 COMPUTATIONAL ELECTROMAGNETICS

The popular methods for computational electromagnetics are the Finite Difference Time Domain method (FDTD), the Method of Moments (MOM), and the Finite Elements (FEM) in general and the the Edge Finite Elements in particular.

Finite Difference Time Domain (FDTD) is a very popular method to solve electromagnetic problems which is constructed based on the Yee's algorithm developed in 1966 [86]. The FDTD method estimates the electric and magnetic fields in space and time simultaneously. In other words, the Maxwell's equations are solved in its coupled form. This method relies on a underlying structured grid which limits its applications for complex geometries. FDTD also losses accuracy when used for inhomogeneous media. Hybrid methods [5] and local mesh refinement strategies can be used to partially resolve these shortcomings [8, 60, 87] at the cost of increased complexity and reduced efficiency.

The *Boundary Element Method (BEM)* and *Method of Moments (MOM)* are considered to be superior to FEM in the sense that it is only necessary to discretize the boundary of the solution domain. This means that BEM requires only surface elements for three dimensional problems, whereas the FEM employs volume elements. It is significantly more efficient for problems with small surface/volume ratio. However, BEM is primarily applicable to linear differential equations with constant coefficients. Its implementation to more general types of differential equations is both cumbersome and computationally expensive: the resulting matrices are fully populated and the computational time tends to grow according to the square of the problem size.

The Finite Element Method (FEM) is a numerical method for solving partial differential equations. This method was developed in the mid-1950s to solve problems in stress analysis. Since then, FEM has been generalized and used to solve a broad range of differential equations such as Laplaces equation, Poissons equation, as well as equations of electrostatics and electromagnetics.

The most significant practical advantage of FEM compared to other methods is its general applicability and its capacity to handle an analysis domain with an arbitrary geometry. For example a two-dimensional analysis domain is discretized into a finite number of elementary units, such as triangles or quadrilaterals, which aptly enough are denoted *finite elements*. The Finite element method was first used for electromagnetics in 1969 to study homogeneous wave guide problem [73]. FEM for electromagnetics was not widely accepted before the introduction of the edge based

vector elements by Nedelec [55,56] to overcome the problem of spurious modes. Edge Finite Elements can provide field tangential continuity along the interface of elements while the normal component is allowed to be discontinuous. These element are also called **H**(curl)-conforming elements.

The codes developed in this study were built upon the existing research tool GeoPDEs [23]. The GeoPDEs has an excellent software architecture which provides a flexible framework to support the use of IGA to solve many academic and engineering problems. The existing GeoPDEs platform has been used to solve Maxwells eigenvalue problem. The codes developed here can solve multi-patch steady state, the transient Maxwells equations, and finally electromagnetic scattering problem. The latter included the PML construction for the IGA. The imaged-based model generation code developed in this study has been used to generate hi-fidelity brain section model. The presented model generation method is compatible with the GeoPDEs structure, which can generate imaged-based models for other applications.

1.2 ELECTROMAGNETIC RADIATION IN BIOLOGICAL MEDIA

Biological tissues are geometrically complex and composed of several layers with inhomogeneous frequency dispersive properties. Like any other media, permittivity, permeability, and conductivity can be defined for each material involved in the human body. Therefore, wave propagation in biological media can be considered a problem of electromagnetic wave incident in a lossy medium. When propagating wave encounters the interface of two mediums a portion of the incident EM energy is

reflected from the interface and a portion is transmitted. In lossy materials, another portion of the electric field dies out eventually through conversion to heat energy.

The geometry of material interfaces affects the generated electromagnetic field inside the brain. Therefore representing an accurate geometry in analysis domain is essential to obtain a reliable solution. The most common numerical method used to find the solution on complex geometries is FEM. However generating a analysis ready geometry for FEM is far from trivial and takes about 80% of the analysis time. Any change in the geometrical model requires remeshing and manual modifications to regenerate the analysis domain. Also the initial model looses geometrical accuracy during discretization.

Isogeometric Analysis (IGA) is a new approach which implements CAD basis functions in the solution domain. Therefore this method provide a direct communication between the analysis domain and the CAD model. Since meshes are generated from the embedded parametric discretization, and therefore can be updated automatically, there is no re-meshing required even for large deformations which makes IGA more flexible for analysis of customizable models. This property is also essential when using IGA as an analysis tool for shape optimization. The models created for IGA do not loose geometrical detail in the analysis domain. In most biomedical applications there is no CAD model. However, numerous effective algorithms have been developed by CAD community that can be used to generate image-based hi-fidelity models.

In addition, IGA provides superior refinement possibilities; the effective knot

insertion algorithms can be used for h-refinement in IGA without affecting the geometry and the continuity of the solution. The existing convenient and effective algorithms to generate p^{th} order NURBS basis functions can be used directly in order elevation or p-refinement in IGA. While the computational cost of FEM and IGA are similar for the same number of degrees of freedom, IGA provides geometrical accuracy, superior refinement possibilities, and control over continuity. Also, the possibility of k-refinement, which is a unique to IGA, can be used to increase the accuracy of the solution with lower computational cost compared to conventional p-refinement in FEM. Taking advantage of these distinguished features for refinement is one of the emphasis of this study.

1.3 SCOPE OF THE DISSERTATION

The objective of the dissertation is development of the Isogeometric Analysis (IGA) for electromagnetism. The required codes were generated to solve the steady and time-dependent Maxwell equations as well as electromagnetic scattering problems in the IGA context. The hi-fidelity image-based model are generated for the IGA and the generated codes are applied to solve an electromagnetic scattering problem in a realistic human head section.

Chapter 2 reviews the h- and p- versions of FEM, the IGA, and the Meshless Methods (MMs) by identifying the common steps and comparing the pros and the cons of each method. The basics of the IGA method is also described in this Chapter. Chapter 3 presents the formulation of the Maxwell equations in the time domain as

well as the time harmonic form. The associated variational equations are derived which are the base for numerical analysis. Next, the formulation of the analytical solution of electromagnetic scattering is presented. Finally, to mimic the infinite domain in the scattering problem, the procedure to construct a Perfectly Matched Layer (PML), originally from the FEM, is extended to the IGA in this study. The PML in the IGA context enjoys exact geometry representation so that it can start exactly at the boundary of the scatterer.

Chapter 4 is dedicated to the generation of the image-based, hi-fidelity models ready for the IGA. In this study, the B-Spline interpolation and Coons patch algorithms are adopted to convert the geometrical data to the multi-patch conforming models ready for IGA. Once the analysis ready model is established, the developed IGA code will be used to find the deposition of electromagnetic energy in the brain section. Numerical simulations presented in Chapter 5 evaluates the IGA code against analytical solution and illustrate the technology. In addition, the numerical results for the scattering problem with a human brain section are presented in this chapter. The developed platform can directly link the imaging data to analysis domain so as to considerably reduce the analysis time. Consequently, converting patient specific data to analysis ready geometries can be done much faster using the IGA method than using the conventional FEM. The advantages of solution refinement mentioned above make IGA a very promising platform for biomedical applications. This dissertation is concluded with Chapter 6 which summarizes the research efforts, their results, limitations and the future direction of research.

CHAPTER 2

A REVIEW OF CONVENTIONAL FINITE ELEMENT METHOD (FEM), ISOGEOMETRIC ANALYSIS (IGA) AND MESHLESS METHODS (MMS)

The most commonly used method for analysis of complex geometries is the Finite Element Method (FEM). However, FEM has its own shortcomings that can affect the accuracy of the solution, computational time, and costs. Several techniques have been proposed to overcome these shortcomings, and these can be grouped into three categories [21]:

1. Improving the variational method, used to obtain finite dimensional form, through selective integration or stabilized methods.

2. Improving the finite element spaces by modifying the underlying basis functions.

3. Methods that combine both types of improvements.

The focus of this chapter will be on FEM, as well as *Isogeometric Analysis (IGA)* and *Meshless Methods (MMs)*. A brief description of the latter two methods follows. Isogeometric Analysis is a recent method that exploits in its analysis the interpolative *shape functions* primarily used in Computer-aided Design (CAD) to represent the underlying geometry. Initially proposed by Hughes [37], this approach

provides an exact representation of the geometry in the solution domain in analysis. IGA applications can be found in structural mechanics, solid mechanics, fluid mechanics, as well as contact mechanics.

In the Finite Element Method, translating CAD files into geometries suitable for analysis requires in excess of 80% of the overall analysis time: mesh generation accounts for 20% while creation of the requisite geometry for analysis requires about 60% [21]. The shortcomings of FEM that can be alleviated by IGA include the following [21]:

- Difficulties in representing the geometrical detail and model imperfections affecting the accuracy of the geometry-sensitive problems such as shell buckling analysis, boundary layer phenomena, hydrodynamics, and contact between bodies.
- Automatic adaptive mesh refinements are not very effective in FEM due to a lack of geometric exactness and automatic communication with CAD.
- No automatic CAD geometry to mesh mapping is possible in FEM, which is necessary for shape optimization.
- Models suitable for analysis are not automatically created or readily meshed.

Meshing, or creating a discretization suitable for analysis, is considered the most labor/time consuming part of FEM. This time can be reduced considerably in IGA by modifying and implementing existing CAD shape functions. However, it is not always possible to readily create a direct link between a CAD model and its analysis.

This complexity arises because CAD models are created as a linear combination of shape functions describing the surface of the object and therefore may require some modifications to serve as the basis functions for analysis in three dimensional domains. As a result, creating a truly direct coupling with CAD is only possible when implementing isogeometric boundary element methods [70, 74]. Meshless Methods (MMs), by largely obviating meshing (although meshing can be exploited to provide proximity information about each node), and thereby alleviate problems related to meshing, especially in some applications such as interactive surgery simulation requiring dynamic re-meshing.

Shortcomings of FEM that can be resolved through MM implementation include the following [57]:

- Errors due to distorted or low quality meshes, which can be exacerbated by deformations.
- Re-meshing that is computationally expensive and time and labor consuming.
- Limited suitability of FEM for discontinuities not aligned with element edges.
- Projection of quantities of interest between successive meshes leads to reduced accuracy and increased computational cost.
- Difficulty of implementing h-adaptivity the possibility of refining, coarsening or relocating a mesh.

However, meshless methods suffer from different types of shortcomings mainly the increased computational cost, reduced accuracy, and difficulties in imposing boundary

conditions. FEM, IGA, and MMs follow similar steps for solving a PDE over a given geometry. In this Chapter each of these common steps is discussed in one section, while advantages and disadvantages of each method are also provided. This discussion expands on the finding that no single method can be selected as the best for all problems and that the selection of the best available method is problem-specific.

Section 2.1 describes the strong and weak forms of a problem. In Section 2.2, the discretization of the geometry and the solution space are defined; moreover, popular discretization methods are described. Section 2.3 focuses on common shape functions and their properties. Shape function selection is shown to be one of the most important steps in solving a discretized problem that will affect the accuracy of the solution, the method of boundary condition imposition, and the computational cost. Section 2.4 discusses the available methods for numerical integration in FEM, IGA, and MMs.

In Section 2.5, the assembly process is briefly discussed. Boundary condition imposition can be challenging specifically in MMs and IGA. The effect of shape function selection on boundary condition imposition method, available techniques and related challenges are discussed in Section 2.6. Refinement is used to achieve reliable solutions and check the convergence rate. There are two main types of refinement: either the number of elements is increased or the order of shape functions is increased. A brief discussion of the available refinements for FEM, IGA, and MMs are provided in Section 2.7. Finally, methods for solving time-dependent problems are discussed in Section 2.8.

2.1 STRONG AND WEAK FORM OF THE PROBLEM

The differential form of the governing *Boundary Value Problem (BVP)* is called the *strong form* of the problem. Let a strong form of the *Laplaces equation* be considered as an example of boundary value problems: *Find* $u : \Omega \rightarrow R$ *such that*:

$$\Delta + \boldsymbol{f} = 0 \quad in \ \ \Omega, \tag{2.1.1a}$$

$$u = g \quad in \ \ \Omega_D, \tag{2.1.1b}$$

$$\bigtriangledown u \cdot \boldsymbol{n} = h \quad in \ \ \Gamma_N, \tag{2.1.1c}$$

$$\beta \cdot u + \bigtriangledown u \cdot \boldsymbol{n} = r \quad in \ \ \Gamma_R, \tag{2.1.1d}$$

where the boundary condition is described as a combination of Dirichlet (Γ_D), Neumann(Γ_N), and Robin(Γ_R) boundary conditions. Therefore the domain boundary ($\partial\Omega$ *or* Γ) can be described as: $\Gamma_D \cup \Gamma_N \cup \Gamma_R = \Gamma \equiv \partial\Omega$ and $\Gamma_D \cap \Gamma_N \cap \Gamma_R = \emptyset$. **n** is defined as unit outward normal vector on Ω. The functions : $\Omega \rightarrow R$, g: $\Gamma_D \rightarrow R$, h: $\Gamma_N \rightarrow R$, and $\Gamma_R \rightarrow R$ are all given, as is the constant β.

Collocation, least-squares finite element analysis and *Galerkin methods* can be used to solve the problem numerically. The first step in solving the differential equations is obtaining the weak or variational form of the problem, after which the variational equations are approximated through the use of interpolation functions. In order to obtain a weak form, two classes of functions are characterized. The first is to be composed of *candidate or trial solutions* which should satisfy Dirichlet boundary conditions . The derivatives of the trial solutions are required to be square-integrable for the problem given in Eq.2.1.1d. The space of square-integrable functions is defined

as the collection of all functions $u : \Omega \rightarrow R$ such that:

$$\int_{\Omega} u^2 \, d\Omega < +\infty. \tag{2.1.2}$$

Considering a multi-index $\boldsymbol{\alpha} \in N^d$ where d is the number of spatial dimensions in space, for $\boldsymbol{\alpha} = \{\alpha_1, \alpha_2, \cdots, \alpha_d\}$ the $|\boldsymbol{\alpha}| = \sum_{i=1}^{d} \alpha_i$. Now the derivative operator of order $|\boldsymbol{\alpha}|, (D^{\alpha})$, can be defined as [21]: $D_u^{\boldsymbol{\alpha}} = \frac{\partial^{|\alpha|} u}{\partial x_1^{\alpha_1} \partial x_2^{\alpha_2} \cdots \partial x_d^{\alpha_d}}$ or $D_u^{\boldsymbol{\alpha}} = D_1^{\alpha_1} D_1^{\alpha_1} \cdots D_n^{\alpha_n}$, where $D_i^j = \frac{\partial^j}{\partial x_i^j}$. The first condition for $u : \Omega \rightarrow$ R to be a trial solution of Eq. 2.1.1 is:

$$\int_{\Omega} \nabla u \cdot \nabla u \, d\Omega < +\infty \tag{2.1.3}$$

Such a function is said to be in the Sobolev space,$H^1(\Omega)$,which is characterized by:

$$H^1(\Omega) = \{u | D^{\boldsymbol{\alpha}} u \in L^2(\Omega), |\boldsymbol{\alpha}| \leq 1\} \tag{2.1.4}$$

The second condition for $u : \Omega \rightarrow$ R, to be a trial solution is that it has to satisfy Dirichlet boundary conditions. In other words, the solution should take on the prescribed value at the specified boundary.

$$u|_{\Gamma_D} = g. \tag{2.1.5}$$

As a result, the set of trial solutions can be defined as:

$$S = \{u | u \in H^1(\Omega), u|_{\Gamma_D} = g\} \tag{2.1.6}$$

This second set of functions is called *weighting functions w.* These are very similar to trial functions, except that it must satisfy *homogeneous* Dirichlet boundary conditions, i.e. $w|_{\Gamma_D} = 0$. The set of weighting functions can be defined by:

$$V = \{w | w \in H^1(\Omega), w|_{\Gamma_D} = 0\} \tag{2.1.7}$$

The variational form of this BVP can be obtained by multiplying the strong form of the differential equation, Eq. 2.1.1, by a set of arbitrary weighting functions,$w \in V$, and integrating the results by parts. The Neumann and Robin boundary conditions (2.1.1.c and 2.1.1.d respectively) will appear during integration which are incorporated by the given boundary conditions of the problem. The resulting weak form of the problem then becomes:

Given $f, g, h,$ and $r,$ find $u \in S$ such that for all $w \in V$:

$$\int_{\Omega} \nabla w \cdot \nabla u \; d\Omega + \beta \int_{\Gamma_R} wu \; d\Gamma = \int_{\Gamma} wf \; d\Omega + \int_{\Gamma_N} w \; h \; d\Gamma + \int_{\Gamma_R} w \; r \; d\Gamma \qquad (2.1.8)$$

Note that all the unknown information, namely u, is contained in the left hand side of the equation, while all the given data f, h, and r are contained in the right hand side. The requirement that u must be contained in Sobolev space is obvious by observing the first term of this equation: without this requirement, the first term may be unbounded. While the strong form of differential equation requires u to have well defined second derivatives, the weak form only requires that the first derivatives be square-integrable. A formulation of this type is called a variational equation, which is very similar to the result of the *Virtual Work Principle*. It can be shown that the weak and strong solutions are identical under appropriate regularity assumptions. Additional notations are used for simplification:

$$a(w, u) = L(w) \qquad (2.1.9)$$

where

$$a(w, u) = \int_{\Omega} \nabla w \cdot \nabla u \; d\Omega + \beta \int_{\Gamma_R} w \; u \; d\Gamma \qquad (2.1.10)$$

and

$$\int_{\Gamma} w\ f\ d\Omega + \int_{\Omega_N} w\ h\ d\Gamma + \int_{\Gamma_R} w\ r\ d\Gamma \tag{2.1.11}$$

Note that $a(w,u)$ is symmetric and bilinear and $L(w)$ is linear, namely:

$$a(u,v) = a(v,u) \tag{2.1.12}$$

$$a(C_1u + C_2v, w) = C_1a(u,w) + C_2a(v,w) \tag{2.1.13}$$

$$L(C_1u + C_2v) = C_1L(u) + C_2L(v) \tag{2.1.14}$$

2.2 DISCRETIZATION

The solution of a PDE is usually a function of the coordinates. Finding the solution for the entire physical domain is only possible for very simple geometries. To overcome this difficulty, numerical methods are utilized to determine the function values for discrete coordinates on grid points. For this purpose both the function space and the physical space are discretized into a finite number of functions and elements respectively. In other word, the discretization in the physical space is equivalent to dividing the physical space into finite elements, which results in a mesh with grid nodes corresponding to a function values in a discretized functional space.

2.2.1 GALERKINS METHOD

The Galerkin method is used for obtaining approximate solutions to a boundary value problem expressed in the weak form, as in Eq. 2.1.8. The first step is to construct a finite-dimensional approximation of trial and weighting function spaces.

The collections of these functions are denoted by S^h and V^h respectively. The superscript refers to the association of S^h and V^h with the h^{th} mesh or discretization of the domain (Ω).

$$S^h \subset S \tag{2.2.1}$$

$$V^h \subset V \tag{2.2.2}$$

For each member function $v^h \in V^h$, a function $u^h \in S^h$ is constructed as follows:

$$u^h = v^h + g^h \tag{2.2.3}$$

where g^h is a given function satisfying essential boundary condition described by Eq.2.1.1b. In other words, $g^h \in S^h$ such that $g^h|_{\Gamma_D} = g$. Since the value of any function, $v^h \in V^h$, is defined to be zero on boundary, u^h of Eq.2.2.3 also satisfies the essential boundary condition. The Galerkin (or Bubnov-Galerkin) form of the problem is then stated as follows:

Given l, q, and h, find $u^h = v^h + g^h$, where $v^h \in V^h$ such that for all $w^h \in V^h$:

$$a(w^h, u^h) = L(w^h) \tag{2.2.4}$$

Since $a(.,.)$ is bilinear, it can be re-written as:

$$a(w^h, v^h) = L(w^h) - a(w^h, g^h). \tag{2.2.5}$$

The variables on the right hand side of Eq.2.2.5 are given while the unknown information, v^h, is on the left hand side. The Galerkin method leads to a coupled system of linear algebraic equations. Let n_{np} denote the total number of degrees of freedom and p denote the order of shape functions. Then, the solution space consists of all

linear combinations of a given set of functions $N_a : \Omega \to R$ where $a = 1 \cdots, n_{np}$. Without loss of generality, one can assume a numbering for these functions where there exists an integer $n_{eq} < n_{np}$ such that $N_a|_{\Gamma_D} = 0$ for all $a = 1, \cdots, n_{eq}$. For a single element, the number of non-zero basis function is $(p+1)^{d_p}$ where d_p is the spatial dimension of the domain. Therefore, considering the weight function $w^h \in V^h$, there exists a set of constants $C_a, a = 1, \cdots, n_{eq}$, such that:

$$w^h = \sum_{a=1}^{n_{eq}} N_a C_a. \tag{2.2.6}$$

As a result, n_{eq} is the number of equations to be solved. The function g^h is given similarly by coefficients $g_b, b = 1, \cdots, n_{np}$. In practice g_b is selected such that $g_1 = \cdots = g_{n_{eq}} = 0$, because they have no effect on its value on Γ_D, and whereby one has:

$$g^h = \sum_{b=n_{eq+1}}^{n_{np}} N_b g_b. \tag{2.2.7}$$

where g_b are control variables associated with Dirichlet boundary condition. Determining these control variables are discussed in Section 2.6.1. Recalling Eq.2.2.3, for any function $u^h \in S^h$, there exists a set of coefficients d_a, $a = 1, , n_{eq}$ such that:

$$u^h = \sum_{a=1}^{n_{eq}} N_a d_a + \sum_{b=n_{eq+1}}^{n_{np}} N_b g_b = \sum_{a=1}^{n_{eq}} N_a d_a + g^h. \tag{2.2.8}$$

The following development leads to a matrix-vector expression of the weak formulation. By rearranging Eq.2.2.3 and inserting into Eq.2.2.8 one can find:

$$v^h = u^h - g^h = \sum_{a=1}^{n_{eq}} N_a d_a. \tag{2.2.9}$$

By inserting u^h, and v^h from Eq.2.2.9, and w^h of Eq.2.2.6 into Eq.2.2.5 one can obtain:

$$a\Big(\sum_{a=1}^{n_{eq}} N_a C_a, \sum_{b=1}^{n_{eq}} N_b d_b\Big) = L\Big(\sum_{a=1}^{n_{eq}} N_a C_a\Big) - a\Big(\sum_{a=1}^{n_{eq}} N_a C_a, g^h\Big), \tag{2.2.10}$$

or

$$\sum_{a=1}^{n_{eq}} C_a \Big\{ \sum_{b=1}^{n_{eq}} a(N_a, N_b) d_b - L(N_a) + a(N_a, g^h) \Big\} = 0. \tag{2.2.11}$$

Since C_a's are arbitrary, the term in parenthesis should vanish. Thus, for $a = 1, \cdots, n_{eq}$, the following holds true:

$$\sum_{b=1}^{n_{eq}} a(N_a, N_b) d_b - L(N_a) + a(N_a, g^h) = 0. \tag{2.2.12}$$

One may define the following terms for $a, b = 1, \cdots, n_{eq}$:

$$K_{ab} = a(N_a, N_b), \tag{2.2.13}$$

$$F_a = L(N_a) - a(N_a, g^h). \tag{2.2.14}$$

Collectively, the above expressions can be expressed in terms of a matrix and vectors, as follows:

$$\boldsymbol{K} = [K_{ab}], \tag{2.2.15}$$

$$\boldsymbol{F} = \{F_a\}, \tag{2.2.16}$$

$$\boldsymbol{d} = \{d_b\}, \tag{2.2.17}$$

as a result, Eq.2.2.10 can be written in the matrix form:

$$\boldsymbol{K}\boldsymbol{d} = \boldsymbol{F}. \tag{2.2.18}$$

Solving for $1, \cdots, n_{eq}$, $\boldsymbol{d}$ is found as:

$$\boldsymbol{d} = \boldsymbol{K}^{-1}\boldsymbol{F}. \tag{2.2.19}$$

Moreover, by inserting the calculated $\boldsymbol{d}$ from Eq.2.2.20 back into Eq.2.2.8, the solution for u^h is given by:

$$u^h = \sum_{a=1}^{n_{eq}} N_a d_a + \sum_{b=n_{eq+1}}^{n_{np}} N_b g_b. \tag{2.2.20}$$

2.2.2 COLLOCATION METHOD

In the collocation method, the trial function selected is the *Dirac delta function*:

$$\delta(x) = \begin{cases} 1 & x = x_i, \\ 0 & \text{otherwise.} \end{cases}$$

$$\int_\Omega u\delta(x - x_i)du = u(x_i) \quad \forall x_i \in \Omega^h \tag{2.2.21}$$

The sifting property of Dirac delta distribution reduces the weak form of Eq.2.1.9 to a discrete strong form, evaluated only at a set of discrete points called the *collocation points* (x_i). Consider the problem presented in Eq.2.2.22 and Eq.2.2.23 in which $\mathcal{L}$ is a linear differential operator. The only stated boundary conditions are Dirichlet boundary conditions given in Eq.2.2.23, also $f = 0$. The collocation problem is: find the discretized solution, u^h, such that:

$$\mathcal{L}u^h(\boldsymbol{x}_i) = 0 \ , I \in \Omega - \Gamma_D \equiv \Omega_{int}, \tag{2.2.22}$$

$$u(\boldsymbol{x}_i) = g(\boldsymbol{x}_i) \ , I \in \Gamma_D. \tag{2.2.23}$$

Since the entire boundary in this example is the Dirichlet boundary, the domain described by Eq.2.2.22 is representing interior collocation points. Boundary collocation points are used in Eq.2.2.23 to assign the prescribed values. By substituting Eq.2.2.8 into Eq.2.2.22 and given that the differential operator is assumed to be linear, one can obtain:

$$\sum_{a=1}^{n_{eq}} \mathcal{L}(N_a(x_i))d_a + g^h(x_i)) = 0, \tag{2.2.24}$$

$$K_{ab} = \mathcal{L}(N_a(x_i)), \tag{2.2.25}$$

$$F_a = -g^h(x_i), \tag{2.2.26}$$

$$\boldsymbol{K} = [K_{ab}], \tag{2.2.27}$$

$$\boldsymbol{F} = \{F_a\}, \tag{2.2.28}$$

$$\boldsymbol{d} = \{d_b\}. \tag{2.2.29}$$

For $a, b = 1, 2, 3, , n_{eq}$, Eq.2.2.22 can be rewritten as:

$$\boldsymbol{K}\boldsymbol{d} = \boldsymbol{F}, \tag{2.2.30}$$

and for non-singular $\mathbf{K}$ there is a unique solution for Eq.2.2.30:

$$\boldsymbol{d} = \boldsymbol{K}^{-1}\boldsymbol{F}. \tag{2.2.31}$$

The fact that there is no integration involved in obtaining the set of algebraic equations is considered one of the advantages of this method. Also the shape functions are evaluated at the collocation points only rather than integration points as in other methods therefore the assembly and evaluations is much faster. However high order

derivatives of shape functions should be evaluated in collocation method. Thus, the shape function must be at least continuous to the order of the PDE.

Another disadvantage is the difficulty in imposing natural Newmann boundary conditions. In the Galerkin method, the Neumann boundary conditions can be incorporated directly through integration by part. However, in the collocation method, there is no integration involved and therefore imposing Neumann boundary conditions is not as straightforward. One should obtain the unit outward normal and describe and impose the prescribed Neumann boundary conditions in terms of the collocation points. Finally, whether or not $\boldsymbol{K}$ can be inverted depends on the location of the collocation points. The necessity to consider an equal number of collocation points and required degrees of freedom in the solution space is a limiting factor. Appropriately locating collocation points is an open research topic.

2.2.3 LEAST SQUARES METHOD (LSM)

The Least Squares Method is very similar to the Galerkin method. The difference is that the differential operator is applied to the weighting function before multiplying to the strong form and integration. Let us consider a simple problem with Dirichlet boundary conditions only.

$$\mathcal{L}u^h(x) = f(x), \quad x \in \Omega - \Omega_D \equiv \Omega_{int}, \tag{2.2.32}$$

$$u(\boldsymbol{x}) = g(\boldsymbol{x}), \quad x \in \Omega_D, \tag{2.2.33}$$

the weak formulation using this approach is:

$$\int_\Omega \mathcal{L}(w)\mathcal{L}(u)d\Omega = \int \mathcal{L}(w)f d\Omega. \tag{2.2.34}$$

Therefore, the trial and weighting functions, u, and w respectively, have the same number of derivatives applied to them. Consequently, it is required that trial and weighting spaces have square-integrable derivatives of the highest order of Eq.2.2.32. Otherwise the left hand side of Eq.2.2.34 may be unbounded. Equation 2.2.34 can be re-written in bilinear form as:

$$a(w,u) = l(w), \tag{2.2.35}$$

where the left and right side of Eq.2.2.35 are defined as follows:

$$a(w,u) \equiv \int_{\Omega} \mathcal{L}(w)\mathcal{L}(u)d\Omega, \tag{2.2.36}$$

and

$$\mathcal{L}(w) \equiv \int_{\Omega} \mathcal{L}(w)f d\Omega. \tag{2.2.37}$$

The remaining steps are identical to the Galerkin method. The resulting stiffness matrix is positive-definite and symmetric regardless of the selected differential operator $\mathcal{L}$. This is in contrast with the collocation and the Least Square methods where stiffness matrix properties are operator-dependent. The requirement of the higher order derivative-integrable functional spaces is challenging, especially for common FEM shape functions. Such a space can be constructed easily inside a patch in IGA but not on the patch boundaries. Considering the fact that fairly complicated shapes can be presented as one patch in IGA, this characteristic makes LSM a promising approach in the IGA framework.

2.2.4 THE PARTITION OF UNITY FINITE ELEMENT METHOD (PUFEM)

The PUFEM is the generalization of the conventional FEM [52]. This method is based on the Partition of Unity property of the basis functions. The PUFEM exploited the following approximation:

$$u^h(\boldsymbol{x}) = \sum_{I=1}^{N} \phi_I^0(\boldsymbol{x}) \Big(\sum_{j=1}^{l} p_j(\boldsymbol{x}) v_{jI} \Big) = \sum_{I}^{N} \Big(\phi_I^0(\boldsymbol{x}) \mathbf{P}^{\mathbf{T}}(\boldsymbol{x}) \Big) \boldsymbol{v}_I, \tag{2.2.38}$$

where shape functions, $\phi_I^0(x)$, are usually obtained from Lagrangian polynomials and v_{jI} are nodal unknowns. The basic idea is to incorporate a non-smooth enrichment function into the solution space using the Partition of Unity property. This is done by incorporating enrichment functions which are typically non-polynomials into the $\boldsymbol{P}(\boldsymbol{x})$.

Therefore the known features of the desired solution, such as cracks, can be implemented by such an enhancement. In this method the number of terms dictates the order of completeness of the approximation. *Completeness* is defined as the ability of an approximation function to reproduce a polynomial of a certain order. For instance, an approximation is called first order (linear) if it produces linear functions exactly.

2.2.5 MESHLESS METHODS

Similarly to FEM, there are two types of MMs: the *intrinsic basis* is similar to FEM *h*-method while the *extrinsic basis* is created to increase the approximation order similar to FEM *p*-method. In addition to the described methods, other

techniques are available for MMs [57]. In the following section the *Reproducing Kernel Particle Method (RKPM), Moving Least Square Method (MLS)* and *hp-clouds* method are briefly discussed.

Reproducing Kernel Particle Method (RKPM)

Initially developed in 1995 ,the discretized approximation in this method has the following form:

$$u^h(\boldsymbol{x}) = \int_{\Omega_y} C(\boldsymbol{x}, \boldsymbol{y}) w(\boldsymbol{x} - \boldsymbol{y}) u(\boldsymbol{y}) d\Omega_y, \tag{2.2.39}$$

where Ω_y represents the discretized domain described by a set of nodes. The correction function, $C(\boldsymbol{x}, \boldsymbol{y})$, is multiplied by a weight function w to increase the completeness of the approximation. The correction function can be constructed by the following procedure [17]. The spline functions are introduced to approximate the unknown function:

$$u(\boldsymbol{y}) = \boldsymbol{P}^T(\boldsymbol{y})\boldsymbol{a}, \tag{2.2.40}$$

$$\boldsymbol{P}(\boldsymbol{y}) = \boldsymbol{P}(\boldsymbol{y})\boldsymbol{P}^T(\boldsymbol{y})\boldsymbol{a}. \tag{2.2.41}$$

Multiplying Eq.2.2.41 by its $\boldsymbol{P}(\boldsymbol{y})$ and integrating over their domain, one has:

$$\int_{\Omega_y} \boldsymbol{P}(\boldsymbol{y}) w(\boldsymbol{x} - \boldsymbol{y}) u(\boldsymbol{y}) d\Omega_y = \int_{\Omega_y} \boldsymbol{P}(\boldsymbol{y})\boldsymbol{P}^T(\boldsymbol{y}) w(\boldsymbol{x} - \boldsymbol{y}) d\Omega_y \boldsymbol{a}, \tag{2.2.42}$$

which is a system of linear equations for $\boldsymbol{a}$, by substituting into the solution in its discretized form, $u^h(\boldsymbol{x}) = P^T(\boldsymbol{x})\boldsymbol{a}$, it yields:

$$u^h(\boldsymbol{x}) = \boldsymbol{P}^T(\boldsymbol{x})\Big(\Big[\int_{\Omega_y} \boldsymbol{P}(\boldsymbol{y})\boldsymbol{P}^T(\boldsymbol{y}) w(\boldsymbol{x} - \boldsymbol{y}) d\Omega_y\Big]^{-1} \int_{\Omega_y} \boldsymbol{P}(\boldsymbol{y}) w(\boldsymbol{x} - \boldsymbol{y}) u(\boldsymbol{y}) d\Omega_y\Big). \tag{2.2.43}$$

Therefore, the correction function can be defined as:

$$\boldsymbol{C}(\boldsymbol{x},\boldsymbol{y}) = \boldsymbol{P}^T(\boldsymbol{x})[\int_{\Omega_y} \boldsymbol{P}(\boldsymbol{y}))\boldsymbol{P}^T(\boldsymbol{y})w(\boldsymbol{x}-\boldsymbol{y})d\Omega_y]^{-1}\boldsymbol{P}(\boldsymbol{y}) = \boldsymbol{p}^T(\boldsymbol{x})[\boldsymbol{M}(\boldsymbol{x})]^{-1}\boldsymbol{P}(\boldsymbol{y}), \tag{2.2.44}$$

which leads to the discretized RKPM:

$$\begin{aligned} u^h(\boldsymbol{x}) &= \int_{\Omega_y} C(\boldsymbol{x},\boldsymbol{y})w(\boldsymbol{x}-\boldsymbol{y})u(\boldsymbol{y})d\Omega_y \\ &= \sum_{I=1}^{N} C(\boldsymbol{x},\boldsymbol{x}_I)w(\boldsymbol{x}-\boldsymbol{x}_I)u_I\Delta V_I = \boldsymbol{p}^T(\boldsymbol{x})[\boldsymbol{M}(\boldsymbol{x})]^{-1}\sum_{I=1}^{N}\boldsymbol{P}\big((\boldsymbol{x}_I)w(\boldsymbol{x}-\boldsymbol{x}_I)\big)\Delta V_I, \end{aligned} \tag{2.2.45}$$

where u_I is the local function value at coordinate x_I and ΔV_I is the volume of Ω_y. The moment matrix, $\boldsymbol{M}(\boldsymbol{x})$, is computed by the following integration:

$$\boldsymbol{M}(\boldsymbol{x})\int_{\Omega_y}\boldsymbol{P}(\boldsymbol{y})\boldsymbol{P}^T(\boldsymbol{y})w(\boldsymbol{x}-\boldsymbol{y})d\Omega_y = \sum_{l=1}^{N}\boldsymbol{P}(\boldsymbol{x}_I)\boldsymbol{P}^T(\boldsymbol{x}_I)w(\boldsymbol{x}-\boldsymbol{x}_I)\Delta V_I. \tag{2.2.46}$$

Moving Least Squares (MLS)

In this method, polynomials of order m with non-constant coefficients are used for approximation of the unknown function in the discretized domain [72]. The local approximation around a point $\bar{x} \in \Omega$, evaluated at point $x \in \Omega$, is given by:

$$u_L^h(\bar{x},x) = \mathbf{P}^T(\bar{x})\mathbf{a}(x), \tag{2.2.47}$$

where the basis function,$P(x)$, is a complete polynomial of order m:

$$\boldsymbol{P}^T(x) = [1, x, x^2, \cdots, x^m], \tag{2.2.48}$$

and $\boldsymbol{a}(x) = [a_0(x), a_1(x), a_2(x), \cdots, a_m(x)]$ is a set of non-constant coefficients. The $a_j(x)$ are obtained by minimizing the following functional $\mathcal{F}(x)$, which is the

weighted summation of the squared difference between the local function approximation $u_L^h(x_I, x)$ and the specified local function value u_I at node I (having coordinate x_I):

$$\mathcal{F}(x) = \sum_{I=1}^{n} w(x - x_I)[u_L^h(x_I, x) - u_I]^2 = \sum_{I=1}^{n} w(x - x_I)[\mathbf{P}^T(x_I)\mathbf{a}(x) - u_I]^2, \quad (2.2.49)$$

where n is the number of nodes in the neighborhood of x such that $w(x - x_I) \neq 0$. To find the minimum of this functional, its derivatives with respect to $a(x)$ are set to zero. This requirement can be stated as:

$$\sum_{I=1}^{n} w(x - x_I)\mathbf{P}(x_I)\mathbf{P}^T(x_I)\mathbf{a}(x) = \sum_{I=1}^{n} w(x - x_I)\mathbf{P}(x_I)u_I. \quad (2.2.50)$$

The matrix form of this requirement is:

$$\boldsymbol{A}(x)\boldsymbol{a}(x) = \boldsymbol{B}(x)\boldsymbol{u}, \quad (2.2.51)$$

where

$$\boldsymbol{A} = \sum_{I=1}^{n} w(x - x_I)\boldsymbol{P}(x_I)\boldsymbol{P}^T(x_I), \quad (2.2.52)$$

and

$$\boldsymbol{B}(x) = [w(x - x_1)\boldsymbol{P}(x_1), w(x - x_2)\boldsymbol{P}(x_2), \cdots, w(x - x_n)\boldsymbol{P}(x_n)], \quad (2.2.53)$$

Eq.2.2.47 is solved by:

$$u^h(x) = \boldsymbol{P}^T(x)[\mathbf{A}(x)]^{-1}\boldsymbol{B}(x)\boldsymbol{u}, \quad (2.2.54)$$

Eq.2.2.51 can be re-casted as:

$$u^h(x) = \sum_{I=1}^{n} \phi_I(x)u_I = \boldsymbol{\phi}^T(x)\boldsymbol{u} \quad (2.2.55)$$

$$\phi^T = \boldsymbol{P}^T(x)[\boldsymbol{A}(x)]^{-1}\boldsymbol{B}(x). \tag{2.2.56}$$

For a two dimensional problem, the linear basis is:

$$\boldsymbol{P}(x) = [1 \;\; x \;\; y], \tag{2.2.57}$$

and the quadratic basis is:

$$\boldsymbol{P}(x) = [1 \;\; x \;\; y \;\; x^2 \;\; y^2 \;\; xy]. \tag{2.2.58}$$

In this framework, the *moment matrix* $A_{m\times m}$ must be inverted to evaluate the shape functions at the x of concern. As a result, MLS-based meshless methods are computationally expensive. Another drawback is the possibility of singularity of the moment matrix. It can be shown that for $n < m$, the moment matrix will become singular.

MLS shape functions are a Partition of Unity. Since the value of the shape function is less than one at the node and non-zero at the other nodes of the domain, MLS shape functions do not satisfy the Kronecker delta property . The first derivative of a MLS shape function is zero-valued at the node which makes MLS-collocation- based MMs unstable [57]. Also, the basis $\boldsymbol{P}(\mathbf{x})$ and therefore the order of consistency cannot be varied from node to node without introducing a discontinuity in the approximation. Thus, it is not possible to increase the order of the approximation (*p-adaptivity*) by increasing the order of polynomial only (*intrinsic enrichment*). Therefore, only regions with different order of consistency can be obtained, that need to be blended together to obtain certain continuity between regions.

hp-clouds Method

In this method, a scattered set of nodes are used to construct the solution approximation. The approximation function is:

$$u^h(\boldsymbol{x}) = \sum_I^N \phi_I^k(\boldsymbol{x}) u_I + \sum_I^M \phi_I^m(\boldsymbol{x}) \sum_j^I P_j(\boldsymbol{x}) v_{jI}, \tag{2.2.59}$$

where $\phi_I^k(\boldsymbol{x})$, and $\phi_I^m(\boldsymbol{x})$ are meshless shape functions of order k and m respectively. The functions,P_j, contain high-order monomials as well as enhancement functions. By introducing enhancement functions, special properties such as discontinuities, singularities, and boundary layers can be captured in the solution.

2.3 SHAPE FUNCTION SELECTION

There are different ways to discretize the geometry of the problem, the choice of which depends on the selected method to represent the geometry. In an isoparametric setting the shape functions used to describe the geometry are also used to obtain the solution in analysis domain. As a result, the selection of the geometrical shape functions will directly affect the properties of the solution.

The parent space in a two-dimensional domain of an *Isoparametric elements* is defined as a closed bi-unit square in R^2 denoted by $\odot$. Let the domain mapping $x : \odot \rightarrow \bar{\Omega}^e$ be of the form:

$$x(\xi) = \sum_{a=1}^{n_{eq}} N_a(\xi) x_a^e, \tag{2.3.1}$$

where $\bar{\Omega}^e$ is the physical domain. If the element interpolation function can be written

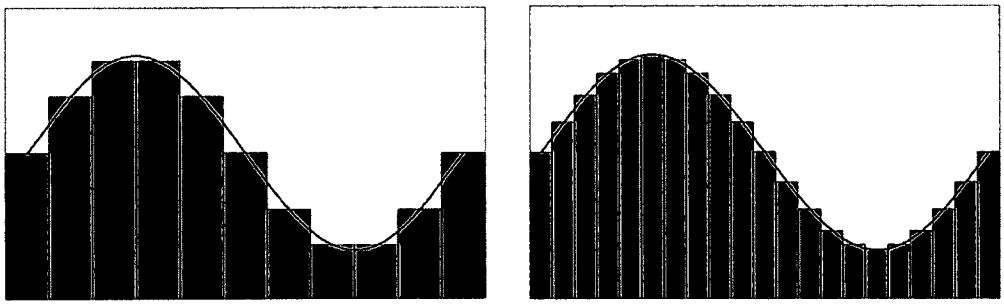
as:

$$u^h(\xi) = \sum_{a=1}^{n_{eq}} N_a(\xi) d_a^e, \tag{2.3.2}$$

such an element is said to be *isoparametric*. In other words, the shape function used to describe the geometry is also used to describe the solution space. It can be shown that an isoparametric element will generally achieve the required condition for convergence. If the computation is converging to a unique correct solution, the FEM based method is called *convergent*. To validate the convergence of the solution, the current result is compared with a solution with increased accuracy. If the more accurate solution is dramatically different with the current result, then the solution has not converged. Conversely, if the difference is less than the convergence criterion, usually less than a few percent, then the solution has converged.The sufficient conditions for convergence are [36]:

- Condition-1. Smoothness of the shape function on each element interior (at least C^m).
- Condition-2. Continuity of the shape function across the element boundary (at least C^{m-1}).
- Condition-3. Completeness.

In the above discussion, m, is the highest order of the derivative in the stiffness integrand, $a(u, v)$, as found in Eq.2.1.10, Eq.2.2.13, and Eq.2.2.36. Note that the required conditions mentioned above are only sufficient conditions for convergence. One can construct convergent elements without satisfying all the stated required



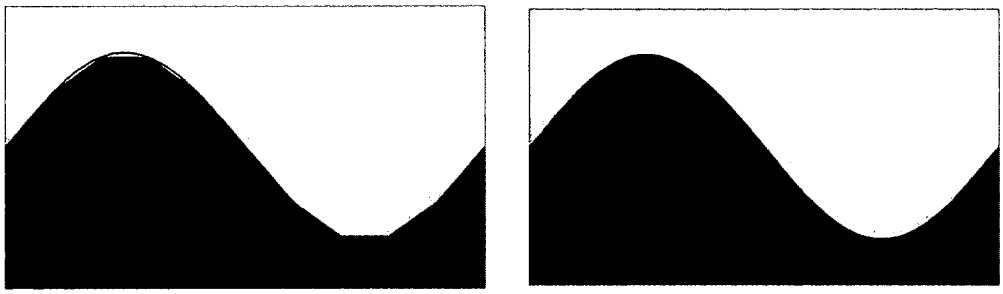
(a) Initial coarse mesh (b) Increased number of meshes

FIGURE. 2.1: h-method

conditions. Satisfying condition-1 and condition-2 (C^0 element) will guarantee that the shape function derivative will have finite jumps across element interfaces in the worst case. Otherwise with finite discontinuities in the shape functions on the element boundaries, the derivatives will be unbounded and the square of the derivatives cannot be calculated.

2.3.1 CONVENTIONAL FEM SHAPE FUNCTIONS

In the conventional FEM, the geometry to be analyzed is broken into finite elements. Both linear and higher order elements can be used and shape functions are associated with the degrees of freedom defined at the vertices of each element. Two common approaches to improve solution quality using conventional Finite Element Method are the h-method and the p-method. If an object is modeled with a very coarse mesh as shown in Fig.2.1a, then the associated solution will be inaccurate. In order to increase the accuracy, the number of elements is increased as depicted in Fig 2.1b. Therefore, in this method, the mesh is refined to achieve convergence without changing the polynomial degree of the shape function. The polynomial order



(a) Initial polynomial order (b) Increased polynomial order

FIGURE. 2.2: p-method

usually is $p = 1$ or $p = 2$ in practical cases. The p-method implements higher-degree polynomials to obtain better solutions without changing the mesh geometry [28]. This method mostly implements a hierarchical set of shape functions that results in a simple and consistent approximation process.

The higher order shape functions are usually set to have zero values along the edges of the elements. As such, increasing the order of the polynomials in shape functions will change the element stiffness matrix but not the connectivity information. p-method technology has been demonstrated to be robust and superior to the conventional h-method for important classes of problems including thin domains and nonlinear applications. The effect of increasing the order of the shape function on the solution accuracy in a one-dimensional problem is shown in Fig.2.2a and Fig.2.2b.

h-method FEM

A one-dimensional example is used as a starting point to illuminate the discussion. A domain of [0 1] is partitioned into n non-overlapping intervals denoted

by$[x_a, x_{a+1}]$ where $x_a < x_{a+1}$ and $a = 1, 2, \cdots, n$. The subintervals are called elements. The length of elements is $h_a = x_{a+1} - x_a$ and the mesh parameter,h, is defined as the length of the maximum subinterval. In other words ,$h = max(h_a), a = 1, 2, \cdots, n$.

Linear shape functions are defined as follows [36]:

$$\begin{cases} \dfrac{(x - x_a)}{h_{a-1}} & x_{a-1} \leq x \leq x_a, \\ \dfrac{(x_{a+1} - x)}{h_a} & x_a \leq x \leq a_{a+1}, \\ 0 & \text{elsewhere.} \end{cases} \tag{2.3.3}$$

For boundary nodes one has:

$$N_1(x) = \frac{x_2 - x}{h_1}, \quad x_1 \leq x \leq x_2, \tag{2.3.4}$$

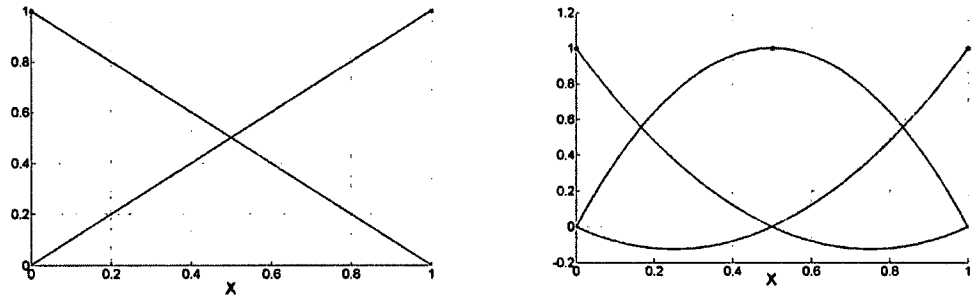
$$N_{n+1}(x) = \frac{x - x_n}{h_n}, \quad x_n \leq x \leq x_{n+1}. \tag{2.3.5}$$

Note that:

$$N_a(x_b) = \delta_{ab}, \tag{2.3.6}$$

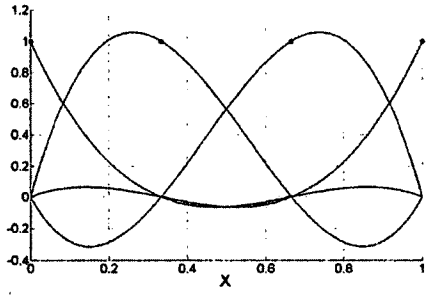
where δ_{ab} is *Kronecker delta*, which means $N_a = 1$ at a and zero for all other nodes. As a result of the definition of shape functions,N_a is non-zero only within a subinterval that contains x_a.

The resulting stiffness matrix is banded, symmetric and positive-definite, which is conducive to efficient calculations. Higher order elements allow more accurate presentation of the boundary edges and surfaces of the curved domains and will lead to more accurate results. However, implementing higher order shape functions will increase the computational cost. The optimal choice is problem-dependent

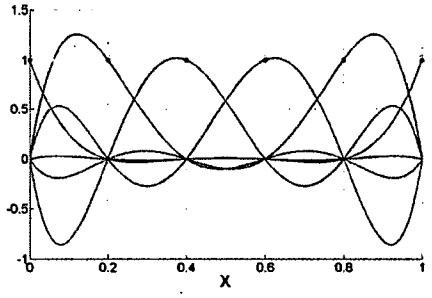


(a) 1^{st} order shape function

(b) 2^{nd} order shape function



(c) 3^{rd} order shape function



(d) 5^{th} order shape function

FIGURE. 2.3: Lagrangian shape functions - FEM

and no particular shape function is preferred exclusively. Higher order basis functions can be obtained from Lagrange polynomials as:

$$l_a^{n_{eq}-1}(\xi) = \frac{\prod_{b=1,b\neq a}^{n_{en}}(\xi - \xi_b)}{\prod_{b=1,b\neq a}^{n_{en}}(\xi_a - \xi_b)} = \frac{(\xi - \xi_1)\cdots(\xi - \xi_{a-1})(\xi - \xi_{a+1})\cdots(\xi - \xi_{en})}{(\xi_a - \xi_1)\cdots(\xi_a - \xi_{a-1})(\xi_a - \xi_{a+1})\cdots(\xi_a - \xi_{en})} \tag{2.3.7}$$

Note that since $l_a(\xi_a) = 1$ and $l_a(\xi_b)_{a\neq b} = 0$, one has $l_a(\xi_b) = \delta_{ab}$, which guarantees the interpolation property as shown for the one-dimensional case in Fig.2.3a. One can define shape functions of an element with n_{en} nodes in one dimension, using Eq.2.3.7:

$$N_a = l_a^{n_{en}-1}. \tag{2.3.8}$$

The Lagrangian polynomials shown in Fig. 2.3 are used as conventional FEM shape

functions. Linear shape functions are shown in Fig. 2.3a where the Kronecker delta property is clearly present. By increasing the order of the polynomial, the number of required shape functions is increased. Regardless of the selected order, the Lagrange polynomials provide Kronecker delta and Partition of Unity properties. Note that the value of the shape function can be negative. The shape functions for a bilinear quadrilateral can be obtained as a product of first-order Lagrange polynomials

$$N_a(\xi,\eta) = l_b^{n_{en}-1} l_c^{n_{en}-1}. \tag{2.3.9}$$

These elements in which the shape functions are obtained as a product of one-dimensional elements are called *Lagrange elements.* As an example, the Laplace equation is solved over an L-shaped domain with Dirichlet and Neumann boundary conditions. Constant triangular elements are used to discretize the given domain. The results are shown in Fig.2.4. The method used in this example is a typical h-method in which re-meshing is used to improve the solution accuracy.

$$\nabla^2 u = 0 \qquad on \ \ \Omega,$$

$$u = 2y - y^2 \qquad on \quad x = -2 \ (\Gamma_1),$$

$$\frac{\partial u}{\partial n} = 0 \qquad \text{on other boundaries } (\Gamma_2).$$

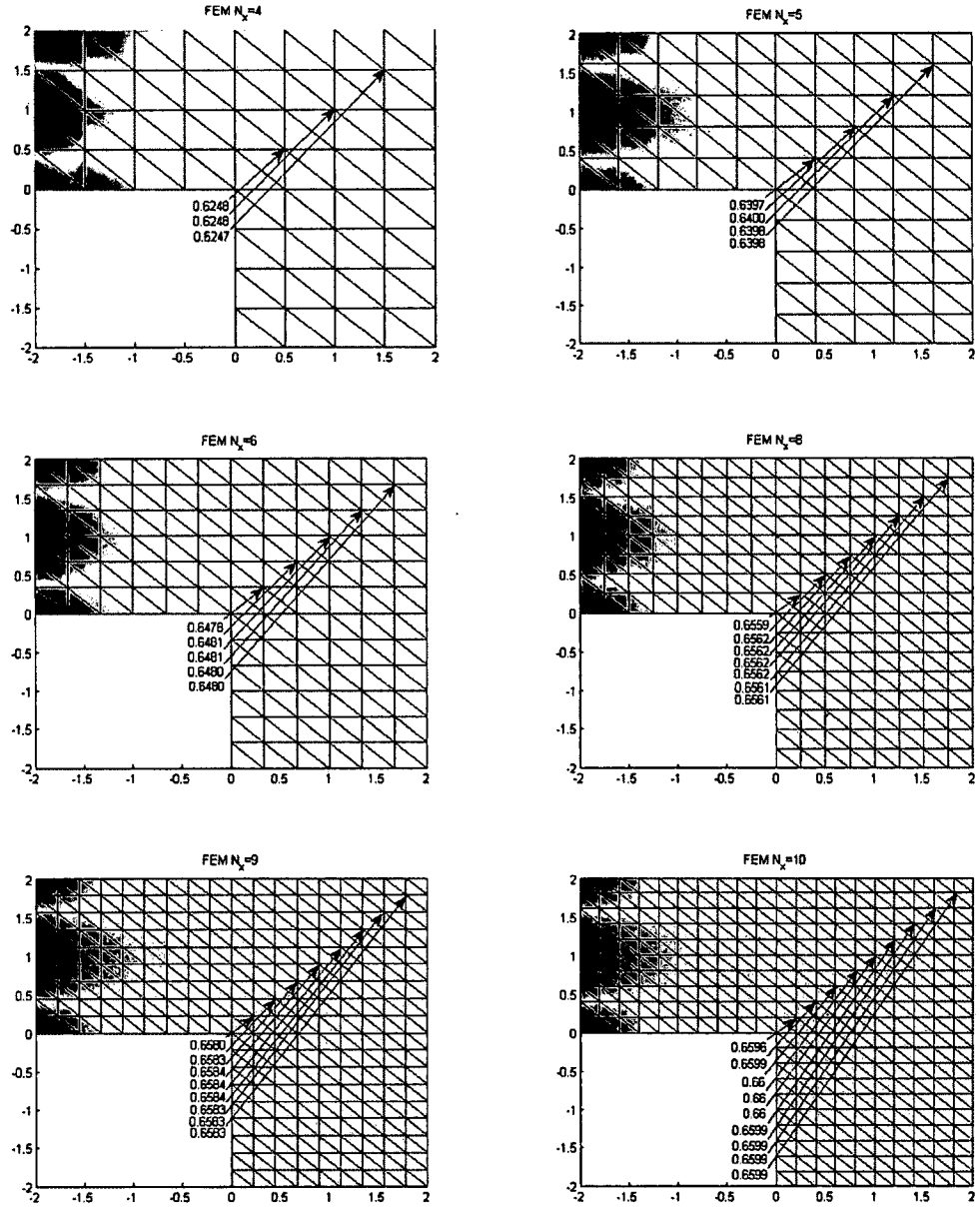


FIGURE. 2.4: A typical h-method FEM: the accuracy of solution is increased by increasing the number of elements

p-method FEM

In the p-method FEM, rather than changing the number of elements, the order of the hierarchical shape function is increased to handle high order displacement functions. The advantage of using a hierarchical basis shape function is that it obviates the reconstruction of the entire basis when the polynomial degree is increased. Shape functions for a one-dimensional domain are defined as integrals of Legendre polynomials [76]. A definition of Legendre polynomials is forthcoming (see Eq.2.4.4). This selection leads to sparse and well-conditioned stiffness matrices. For rectangular and parallelepiped elements, shape functions are defined as tensor product of their one-dimensional counterparts. Such a shape function, also has the desired orthogonality properties which can be described as $\int_{-1}^{1} P_m(x)P_N(x)dx = \frac{2}{2n+1}\delta_{nm}$ where δ_{nm} is the Kronecker delta; therefore this integral is zero if $m \neq n$. However, for triangular and tetrahedral elements, such a selection will result in a condition number of the stiffness matrix that increases exponentially with increased p.

A method was proposed to reduce this exponential growth to quadratic growth [1]. Also, A hierarchical basis has been developed for triangles and tetrahedron that improves conditioning [14]. These bases are constructed so that edge, face, and region shape functions of degree p are orthogonal to those of degrees not exceeding $p-2$, $p-3$ and $p-4$ respectively. One and two-dimensional shape functions are given as follows [1]:

One-dimensional shape functions

The basis is defined on the element $K := \{\xi | -1 \leq \xi \leq 1\}$. Two shape

functions are associated with the vertices at $\xi = (-1, 1)$:

$$\hat{\phi}_1^0(\xi) = \frac{1-\xi}{2}, \quad \hat{\phi}_2^0(\xi) = \frac{1+\xi}{2}, \quad \xi \in K \tag{2.3.10}$$

and the $p-1$ shape functions associated with the region $-1 \leq \xi \leq 1$ are:

$$\hat{\phi}_i^1(\xi) = \sqrt{\frac{2i+1}{2}} \int_{-1}^{\xi} P_i(t)dt, \quad \xi \in K, \quad i = 1, 2, 3, \cdots, p-1 \tag{2.3.11}$$

where $P_i(\xi)$ is the Legendre polynomial of degree i.

Two-Dimensional shape functions

The basis is defined in terms of the barycentric coordinates on the reference triangle k.

$$L_1 = \frac{1}{2}(1 - \xi - \frac{\eta}{\sqrt{3}}), \quad L_2 = \frac{1}{2}(1 + \xi - \frac{\eta}{\sqrt{3}}), \quad L_3 = \frac{\eta}{3} \tag{2.3.12}$$

The work of Szabo-Babuska [76] consists of three vertex shape functions, $3(p-1)$ edge shape functions and $\frac{(p-2)(p-1)}{2}$ face shape functions. Therefore the total number of shape functions is:

$$n_p = \frac{(p+1)(p+2)}{2} \tag{2.3.13}$$

$$\hat{\phi}_1^0(\xi, \eta) = L_i, \quad i = 1, 2, 3. \tag{2.3.14}$$

while the edge shape functions are given by:

$$\hat{\phi}_1^{1,j}(\xi, \eta) = L_{j1} L_{j2} \varepsilon_i(L_{j1}, L_{j2}) \quad i = 1, 2, 3, \cdots, p-1, \quad j_1 = j, \quad j_2 = 1 + j_1 mod 3, \tag{2.3.15}$$

$$\varepsilon_i(t_1, t_2) = -\frac{8\sqrt{4i+2}}{i(i+1)} P_i'(t_2 - t_1) \tag{2.3.16}$$

Moreover, the face shape functions are defined by:

$$\hat{\phi}^{2,j}_{i_k+r} = L_1 L_2 L_3 F_{r_1 r_2}(L_1, L_2, L_3), \tag{2.3.17a}$$

$$i_k = \frac{(k-3)(k-2)}{2}, \tag{2.3.17b}$$

$$r = 1, 2, \cdots, k-2 \tag{2.3.17c}$$

$$r_1 + r_2 = k - 3 \tag{2.3.17d}$$

$$k = 3, 4, \cdots, p \tag{2.3.17e}$$

$$F_{r_1 r_2}(t_1, t_2, t_3) = \sum_{i=0}^{r_2}\sum_{j=0}^{r_1}(-\frac{1}{2})^{i+j}\binom{r_1}{j}\binom{r_1+1}{j}\binom{r_2}{i}\binom{r_2+1}{i} \times \frac{i!j!(i+j)!t_1^{r_1-j}t_2^{r_2-j}}{\prod_{k=1}^{i+j}(k(r_1+r_2+2)-\frac{k(k-1)}{2})} \tag{2.3.18}$$

The Carnevali basis is similar to the Szabo-Babuska basis [76]. The difference is in definition of ε in the edge shape function. Instead of Eq.2.3.16, the following equation is used for ε_i:

$$\varepsilon_i(t_1, t_2) = \sum_{k=0}^{i}\frac{(-1)^k}{k+1}\binom{i}{k}\binom{i+1}{k}t_1^k t_2^{i-k}. \tag{2.3.19}$$

2.3.2 IGA SHAPE FUNCTIONS

As described in the previous section and depicted in Fig.2.3, the Lagrangian basis functions are most commonly used in describing the geometry and the unknown field in an isoparametric FEM. The errors in the geometry representation however, will affect the accuracy of results. Also, after generating the discretization from of the CAD model using Lagrangian polynomials, the lost information is not retrievable which leads to lower efficiency specifically in an iterative process. IGA is also an

isoparametric process which is constructed by the same CAD basis functions used to describe the geometry.

In IGA, the exact geometry is used at all stages of analysis which eliminates geometrical error even after refinement of the unknown field. The meshing process is greatly reduced or eliminated in this method. In IGA, the geometry is described through parametric functions. In other words, the geometry is mapped from a parameter space to a geometrical space. To illustrate this concept, consider the following simple geometric example. The implicit form of the equation of a circle with a unit radius is given by: $x^2 + y^2 = 1$. The associated parametric form is given by: $\mathbf{f}(t) = (cos(t), sin(t))$, which is a function of a single parameter t and corresponds to a mapping $f : [0, 2\pi] \rightarrow R^2$.

The parametric function is much more convenient to plot. Likewise, several CAD transformation algorithms become simpler when described by parametric functions. B-Splines and *Non-Uniform Rational B-Spline (NURBS)* are both based on parametric functions. In IGA, knots divide the parameter space into elements. A knot vector is defined as a set of non-decreasing coordinates and represented by $\Xi = \{\xi_1, \xi_2, , \xi_{(n+p+1)}\}$ where ξ_i is the i^{th} knot and i is the knot index, $i = 1, 2, \cdots, n + p + 1$. Note that p is the polynomial order and n is the number of basis functions used to construct the B-Spline curve. In an open knot vector, the first and the last knot values appear $p + 1$ times. If the knots are spaced equally within the knot vector, the knot vector is called *uniform* otherwise it is *non-uniform.*

The boundary of a d-dimensional B-Spline curve constructed using open knot

vectors is a $(d-1)$-dimensional B-Spline. At the ends of each parameter space interval $[\xi_1, \xi_{n+p+1}]$ and at the corner of the patches in multiple dimensions, the basis function created by an open knot vector is interpolatory. However, they cannot be considered interpolatory at the interior knots. This is in contrast with interpolatory nodes in the conventional finite element method.

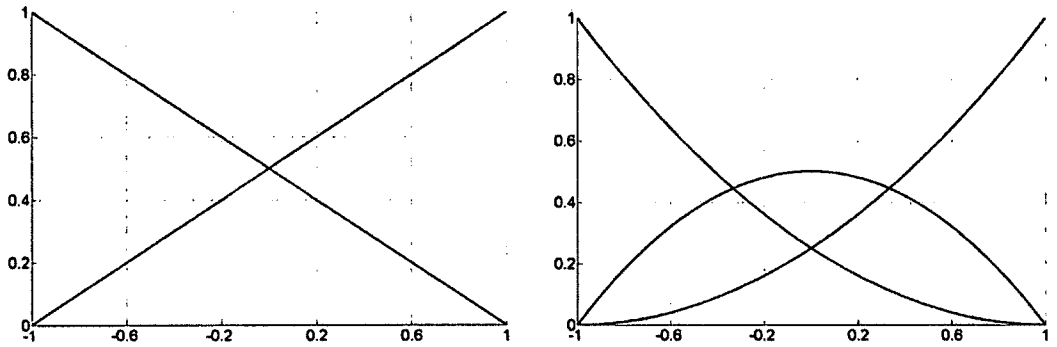
Note that using a unique discretization for both geometry and analysis impose a constraint. By considering two separate spline parametrization in a way that both can span the domain of interest, one can freely apply local refinement in solution domain without altering the parametrization used for the initial CAD description [84]. One can use alternative discretization in the solution domain such as T-splines [11, 69, 71] which can create a single-patch water-tight geometries. T-splines have used in IGA [4, 27] and conventional FEM [27] context. Alternatively, polycube splines [82], Polynomial splines over Hierarchical T-meshes (PHT splines) [24,58,59] and Locally Refined (LR) splines [25] , hierarchical B-Splines [10,67,79] and Powell-Sabin spline [75] can be used.

B-Splines

A B-Spline basis function is defined recursively by the Cox-de Boor recursion formula starting with the zeroth order ($p = 0$) basis function [61]:

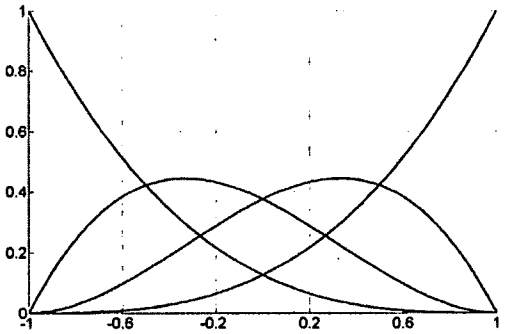
$$\text{for } p = 0,\ N_{i,p}(\xi) = \begin{cases} 1 & \xi_i \leq \xi \leq \xi_{i+1}, \\ 0 & \text{otherwise}, \end{cases} \tag{2.3.20}$$

$$\text{for } p = 1, 2, 3, \cdots \quad N_{i,p}(\xi) = \frac{\xi - \xi_i}{\xi_{i+p} - \xi_i} N_{i,p-1}(\xi) + \frac{\xi_{i+p+1} - \xi}{\xi_{i+p+1} - \xi_{i+1}} N_{i+1,p-1}(\xi), \tag{2.3.21}$$

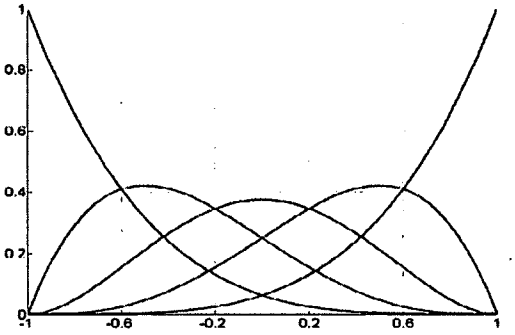


(a) 1^{st} order B-Spline shape function

(b) 2^{nd} order B-Spline shape function



(c) 3^{rd} order B-Spline shape function



(d) 4^{th} order B-Spline shape function

FIGURE. 2.5: Shape functions - IGA

where $\frac{0}{0}$ is defined to be zero. To compute the tangent and the normal functions, it is required to calculate the derivative of B-Splines. The required equations and efficient algorithms to calculate B-Splines and their derivatives can be found in [61]. The first, second, third, and fourth order B-Spline shape functions are depicted in Fig.2.5. Note that the first-order B-Spline functions are identical to their Lagrangian (FEM) counterparts, while the B-Splines number of required shape functions for a specific order is similar to the Lagrangian. For example, three shape functions are required to construct second-order Lagrangian or B-Spline shape functions. In contrast to FEM, shape functions in IGA are non-negative.

B-Spline basis properties

The B-Spline functions possess the following properties:

- *Partition of Unity:*

$$\forall \xi \quad \sum_{i=1}^{n} N_{i,p}(\xi) = 1, \tag{2.3.22}$$

- *Point-wise non negativity*

$$N_{i,p}(\xi) \geq 0, \quad i = 1, 2, \cdots, n, \tag{2.3.23}$$

- B-Spline shape functions are *linearly independent.*

- They exhibit C^{p-m_i} continuity across knot ξ_i where m_i is the multiplicity of the knot ξ_i.

B-Spline curves

Similarly to FEM, the piecewise-polynomial B-Spline curves in R^d are constructed as a linear combination of B-Spline basis functions:

$$\boldsymbol{C}(\xi) = \sum_{i=1}^{n} N_{i,p}(\xi)\mathbf{B}_i. \tag{2.3.24}$$

The vector-valued coefficients of the basis functions are called *control points* ($B_i \in R^d, i = 1, 2 \cdots, n$). Since $C(\xi)$ in Eq.2.3.24 is representing a curve, the corresponding control points,B_i , are analogous to nodal coordinates in FEM. The resulting curve is C^{p-1} continuous everywhere except at the location of the repeated knots with m_i multiplicity, where it is C^{p-m_i}. Therefore when utilizing open knot vectors the resulting curve is C^0-continuous, tangent to the control polygon, and interpolatory at the

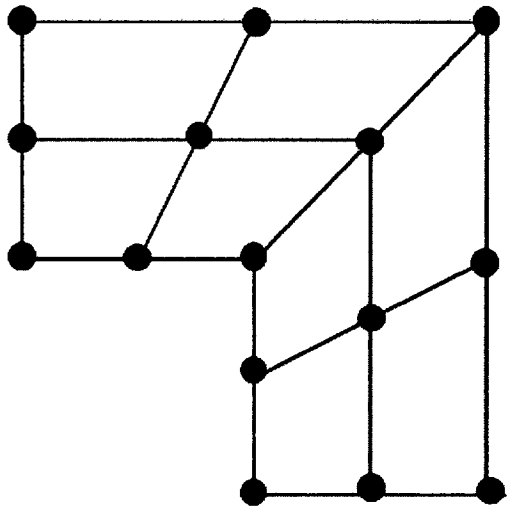


FIGURE. 2.6: Selection of control points- L shape domain

start and the end of the curve and where the multiplicity is equal to the polynomial order. Figure 2.6 shows the selected control points to represent a simple L-shaped geometry in IGA. The required meshing is automatically created and refined. The resulting two-dimensional shape functions are shown in Fig.2.7.

B-Spline curve properties

B-Spline curves inherit most of the properties of their basis functions, as described below.

- *Locality*: The support of a B-Spline basis function is local. In other words, moving a single control point can affect the geometry of no more than $p+1$ elements of the curve.

- *Continuity of B-Spline curve derivatives*: B-Spline curves of degree p have $p-1$ continuous derivatives in the absence of repeated knots or control points.

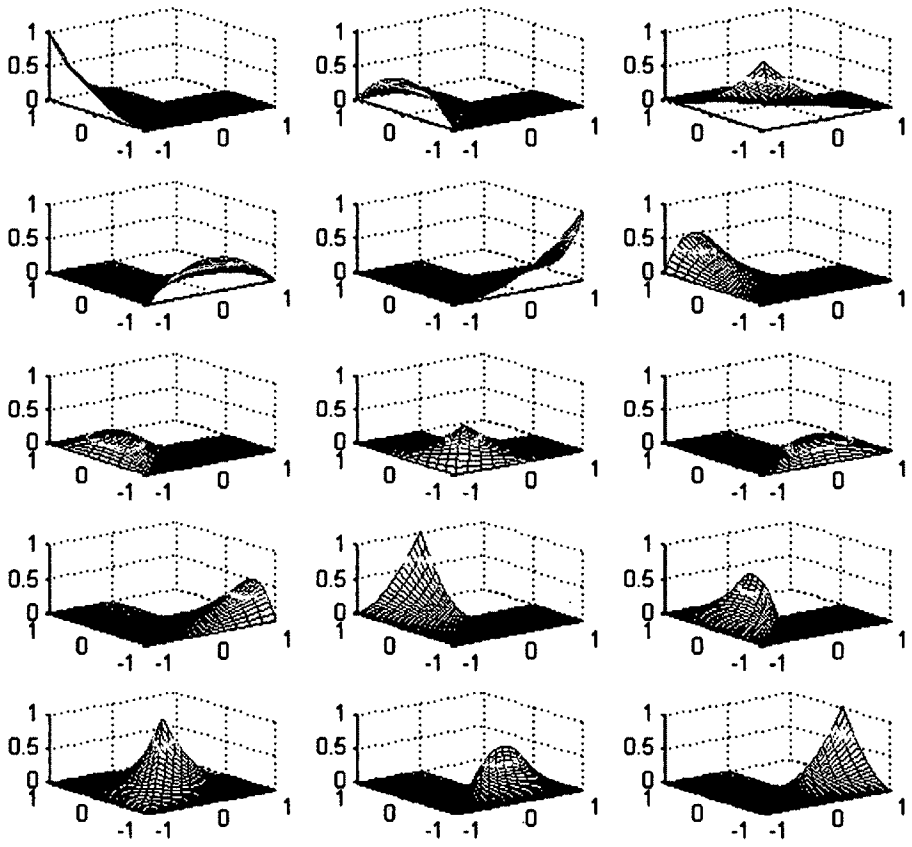


FIGURE. 2.7: Two-dimensional basis functions- L shaped domain

- *Convex hull containment property*: for a curve of degree p, the convex hull is defined as the union of all the convex hulls formed by $p+1$ successive control points. A B-Spline curve is completely contained within the convex hull defined by its control points. This property is the result of non-negativity, Partition of Unity, and the compact support properties of these functions.

- *Control of continuity*: the continuity of the basis function at a knot value with multiplicity k is C^{p-k}. For $k = p$, the basis is C^0-continuous and interpolatory

at that location. This property provides great flexibility in modeling discontinuities such as crack and material interfaces, where higher-level (derivative) continuity would otherwise lead to unwanted undulations.

- *Variation-diminishing property*: No plane has more intersection with the curve than it has with the control polygon.

B-Spline surfaces and volumes

A tensor product B-Spline surface is defined as:

$$\mathbf{S}(\xi,\eta) = \sum_{i=1}^{n}\sum_{j=1}^{m} N_{i,p}(\xi) M_{j,q}(\eta) \mathbf{B}_{i,j}, \tag{2.3.25}$$

where $\mathbf{B}_{i,j} = 1, 2, \cdots, n, \quad j = 1, 2, \cdots, m$ are control points, and $N_{i,p}(\xi)$ and $M_{j,q}(\eta)$ are univariate B-Spline functions of orders p and q corresponding to the knot vectors $\Xi^1 = \{\xi_1, \xi_2, \cdots, \xi_{n+p+1}\}$ and $\Xi^2 = \{\eta_1, \eta_2, \cdots, \eta_{m+q+1}\}$, respectively. Similarly, a B-Spline volume is constructed as a tensor product of three B-Spline basis functions:

$$\mathbf{V}(\xi,\eta,\zeta) = \sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{k=1}^{l} N_{i,p}(\xi) M_{j,q}(\eta) L_{k,r}(\zeta) \mathbf{B}_{i,j,k}. \tag{2.3.26}$$

NURBS

There are certain geometries, such as circles and ellipsoids, which B-Spline functions and Lagrange polynomials cannot represent exactly. In the CAD community, B-Spline functions are replaced with a more general form to overcome this shortcoming. This new form is called Non-Uniform Rational B-Splines (NURBS). NURBS shape functions are defined as:

$$R_i^p(\xi) = \frac{N_{i,p}(\xi) w_i}{W(\xi)} = \frac{N_{i,p}(\xi) w_i}{\sum_{i=1}^{n} N_{i,p}(\xi) w_i}, \tag{2.3.27}$$

where $\{N_{i,p}\}$ is a set of B-Spline basis functions and $\{w_i\}$ is a set of positive NURBS weights. By appropriate weight selection both polynomials and circular arcs can be described. Efficient algorithms to calculate NURBS basis functions and their derivatives are provided in [61]. NURBS curves are defined as follows:

$$C(\xi) = \sum_{i=1}^{n} R_i^p(\xi)\mathbf{B}_i, \tag{2.3.28}$$

Similarly, rational surfaces and volumes are defined in terms of rational basis functions as:

$$R_{i,j}^{p,q}(\xi,\eta) = \frac{N_{i,p}(\xi)M_{j,q}(\eta)w_{i,j}}{\sum_{i=1}^{n}\sum_{j=1}^{m} N_{i,p}(\xi)M_{j,q}(\eta)w_{i,j}}, \tag{2.3.29}$$

$$R_{i,j}^{p,q}(\xi,\eta,\zeta) = \frac{N_{i,p}(\xi)M_{j,q}(\eta)L_{k,r}(\zeta)w_{i,j,k}}{\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{k=1}^{l} N_{i,p}(\xi)M_{j,q}(\eta)L_{k,r}(\zeta)w_{i,j,k}}. \tag{2.3.30}$$

Note that if the weights are all equal, NURBS basis functions will reduce to their B-Spline counterparts and the corresponding curve becomes a non-rational polynomial again. The main strengths of NURBS consist of the following characteristics:

- Convenient for free form surface modeling.
- Can exactly represent conic sections like circles, cylinders, spheres, etc.
- Many efficient and numerically stable algorithms are available to create NURBS objects.

There are two types of meshes:

1. A *control mesh* is defined by control points and does not conform to the actual geometry. The control variables (degrees of freedom) are located at the *control points*. For a sufficiently smooth NURBS, the control mesh can be severely distorted and even inverted, while still representing a valid geometry.

2. A *physical mesh* is a decomposition of the actual geometry presented as non-zero knot spans or elements. Patches can be considered as macro elements.

2.3.3 MESHLESS METHOD SHAPE FUNCTIONS

Meshless Methods (MMs) eliminate the need for meshing the domain by building the approximation from nodes only. The approximation of a scalar function u in terms of material (Lagrangian) coordinates can be written as $u(x,t) = \sum_{I \in S} \phi_I(\mathbf{x}) u_I(t)$ where $\phi_I : \Omega \rightarrow R$ are the shape functions and $u_I(t)$ is the value at point I located at position x_I. S is a set of points, I, for which $\phi_I(x) \neq 0$.

Kernel weight functions

The shape functions $\phi_I(x)$ are obtained from kernel functions. Kernel weight functions ($w_I : \Omega \rightarrow R$) have compact support [57]. The support size is defined by the so-called dilatation parameter or smoothing length. Dilatation is equivalent to the element size in FEM and is considered critical to solution accuracy and stability. The weight function is required to be positive and continuous in its support. The continuity of the shape function is determined by continuity of this kernel function.

The focus in this section will be on meshless methods based on it global weak forms, especially the Element Free Galerkin (EFG) method and Reproducing Kernel Particle Method (RKPM). Another class of meshless methods are methods based on *local weak forms* such as Meshless Local Petrov-Galerkin (MLPG) and moving point methods which generate and integrate local weak forms on overlapping sub-domains. The EFG and RKPM are similar in their equations and both can

be considered as methods with an intrinsic basis. Other meshless methods, such as *hp*-cloud, implement an extrinsic basis to be able to increase the approximation order similar to FEM *p*-method. Some commonly used weight functions are the following [57].

The cubic spline weight function:

$$w(r) = \begin{cases} \frac{2}{3} - 4r^2 + 4r^3 & r \leq \frac{1}{2}, \\ \frac{4}{3} - 4r + 4r^2 - \frac{4}{3}r^3 & x_a \leq x \leq a_{a+1}, \\ 0 & r \geq 1, \end{cases} \tag{2.3.31}$$

The quadratic spline weight function:

$$w(r) = \begin{cases} 1 - 6r^2 + 8r^3 - 3r^4 & r \leq 1, \\ 0 & r > 1, \end{cases} \tag{2.3.32}$$

with

$$r = \frac{\| X_I - X \|}{d_I}, \tag{2.3.33}$$

where d_I is the support size of node I. In two dimensions, circular and rectangular supports are usually used.

Circular support

$$w(\mathbf{x} - \mathbf{x}_I) = w\left(\frac{\| X_I - X \|}{d_I}\right). \tag{2.3.34}$$

Rectangular support

$$w(\mathbf{x} - \mathbf{x}_I) = w\left(\frac{|x_I - x|}{d_I^x}\right) w\left(\frac{|y_I - y|}{d_I^y}\right). \tag{2.3.35}$$

The derivatives of the weight functions can be computed using the chain rule; for circular supports, one has:

$$w_k(r) = w_r(r)r_k = w_r\frac{x_k - x_{Ik}}{rd_I^2}. \tag{2.3.36}$$

The order of completeness of the solution space (defined as $u(x,t) = \sum_{I\in S}\phi_I(x)u_I(t)$) can be increased either by increasing the order of completeness of the shape functions (intrinsic) or by increasing the number of low order shape functions (Partition of Unity concept). Famous intrinsic meshless methods include: Smooth Particle Hydrodynamic Method (SPH) [22], Reproducing Kernel Particle Method (RKPM), and Moving Least Squares (MLS) approximation . The most common extrinsic methods are: the Partition of Unity Finite Element Method (PUFEM) and *hp*-clouds method [57].

Comparing shape functions in FEM, MMs and IGA

Advantages of meshless methods compared to FEM

- *h*-adaptivity is simpler to incorporate than in mesh-based FEM methods.
- It is easier to treat problems with discontinuity such as crack propagation, shear bands, phase transformation.
- Large deformations can be handled more robustly.
- Shape functions of higher order continuity are available.
- Non-local interpolation character.

- No sensitivity to mesh alignment.

Disadvantages of MMs compared to FEM

- Because of the rational shape function, a higher order integration scheme is required.
- Since MMs shape functions are not interpolative, the treatment of boundary condition is not as straightforward as in FEM.
- MMs do not satisfy the Kronecker delta property. Therefore direct imposition of Dirichlet boundary conditions is not possible.
- In general, the computational cost of MMs is higher than FEM [57].

Disadvantages of MMs compared to IGA

- In general, a higher order integration scheme is required in MMs due to the use of rational shape function. IGA requires rational shape functions only in case of conical shapes for exact representation of the geometry.
- MMs analysis usually contains errors due to an inaccurate geometric representation.
- MMs shape functions are not interpolatives, however IGA shape functions are interpolative at the first and last knots and satisfy the Partition of Unity property. This simplifies the treatment of boundary conditions.
- In general, the computational cost of MMs is higher than IGA.

Comparing shape functions used in FEM and IGA

B-Spline functions generated with $p = 0$ and $p = 1$ orders will result in the same piecewise constant and linear functions as standard FEM shape functions. The differences are found for cases using higher order shape functions, which are described as follows.

- *Homogeneity*: Quadratic and higher order B-Spline shape functions are homogenous. In other words, they are all identical but shifted relative to each other. However, their counterparts in conventional FEM are not homogenous as the shape functions at the end nodes are different from those at the internal nodes.

- *Positivity*: Each B-Spline basis is point-wise non-negative over the entire domain ($\forall \xi, N_{i,p}(\xi) \geq 0$). Because of this property, all of the entries of the mass matrix will be positive.

- *Higher order continuity*: The p^{th} order B-Spline function has $p - 1$ continuous derivatives across the element boundaries.

- *Spatial support*: The support of a B-Spline function of order p is always $p + 1$ knot spans. Therefore, the higher order B-Spline function has support over much larger portion of the domain than classical FEM functions do. Note that the total number of functions that any given function shares support with (including itself) is $2p + 1$ regardless of whether a FEM basis or a B-Splines is used. Therefore, using higher order B-Spline basis functions will provide

improved support compared to standard FEM without increasing the number of required shape functions and computational cost.

- *Interpolatory characteristics*: B-Spline basis functions are non-interpolatory. The resulting B-Spline curve is only interpolatory at the first and last control points and where multiplicity of the knot is equal to the polynomial order. Therefore, the Kronecker delta property is not guaranteed at the control points. As a result, imposing non-homogeneous Dirichlet boundary conditions is not straight-forward in IGA.

- *Robustness to oscillations*: Increasing the order of Lagrangian polynomials will increase the amplitude of oscillations in FEM. This problem is eliminated if the same data is used as control points of a B-Spline. This property is a result of non-negativity and non-interpolatory nature of B-Splines shape functions which is proven to be useful in analysis, specifically if noise is involved.

2.4 INTEGRATION

The focus of this section will be the integration method of Gaussian quadratures which is commonly used in FEM and IGA as well as methods that utilize a background mesh for MMs. The method of Gaussian quadrature is considered more accurate compared to other numerical integration methods. This method is ideally applicable to small and moderate deformations. Other methods such as MLPG, which are based on local weak forms, adopt an integration approach defined over the shape function supports or the intersection of supports [21, 61, 74]. Nodal and stress point integrations are mostly used in dynamics and where large deformations

are expected.

2.4.1 INTEGRATION-FEM

Numerical integrations in FEM can be performed by implementing an intermediate variable that maps the element domain over the parent domain. The parent domain is a unit length, unit square or unit cube for one, two, and three-dimensional problems, respectively. The method of Gaussian quadratures is considered as an optimal method of integration for conventional FEM. Accuracy of order $2n_{int}$ is achieved by n_{int} integration points [36].

General Gaussian quadratures rule:

The general Gaussian quadrature rule for a one-dimensional domain is defined by:

$$\int_{-1}^{1} g(\xi)d(\xi) = \sum_{l=1}^{n_{int}} g(\bar{g}_l)W_l + R \tag{2.4.1}$$

where n_{int} is the number of integration points, $\bar{g}_I$ is the coordinate of the l^{th} coordination point, W_l is the weight of the l^{th} coordination point, and R is the residual:

$$W_l = \frac{2}{[(1-\bar{\xi}_l^2)(P'_{n_{int}}(\bar{\xi}_l)^2)]}, \qquad 1 \leq l \leq n_{int}, \tag{2.4.2}$$

$$R = \frac{2^{2n_{int}+1}(n_{int}!)^4}{(2n_{int}+1)[2(n_{int})!]^3} g_{,\xi_1,\cdots,\xi_n}(\bar{\xi}), \tag{2.4.3}$$

$\bar{\xi}_l$ is the l^{th} zero of the Legendre polynomial defined by Eq.2.4.4.

$$P_{n_{int}}(\xi) = \frac{1}{2^{n_{int}} n_{int}} \frac{d^{n_{int}}}{d\xi^{n_{int}}}(\xi^2-1)^{n_{int}}, \tag{2.4.4}$$

where $P'_{n_{int}}$ denotes the derivative of $P_{n_{int}}$.The method of Gaussian quadratures can be applied to problems with multiple dimensions. In this case, the method of Gaussian quadrature is implemented along each dimension.

2.4.2 INTEGRATION-IGA

Similar to FEM, the common method of integration used in IGA is the method of Gaussian quadratures. However, this method is not considered the optimal method of integration of NURBS basis functions. Recently a new method, the half-point rule, have been proposed [2,38] which is considered nearly optimal. To perform numerical quadrature integration, integrals are mapped into the parametric space first and then onto the parent space. In the parent space, integration is performed using a classical change of variables formulation similar to the one used by the conventional FEM. The parameter and the parent spaces are defined in the following section. Since in IGA integrations are carried out at the knot span level, each nonzero knot span is equivalent to one element in FEM.

Required spaces

In FEM, in addition to the physical space, a parent space should be defined for integration. In IGA, however, four different spaces are defined namely index, parameter, physical, and parent spaces. *Index space* is defined by assigning each knot of the knot vector a unique coordinate. Even repeated knots have their distinct coordinate in this space and each knot is uniquely identified. Assigning distinct coordinates to repeated knots will result in discrimination among knots having multiplicities greater than one. However, this would create regions of zero parametric areas in this space. *Parameter* space is created by considering nonzero knot spans only. Normalization can be used to reduce this space to a unit interval, unit square, or a unit cube for

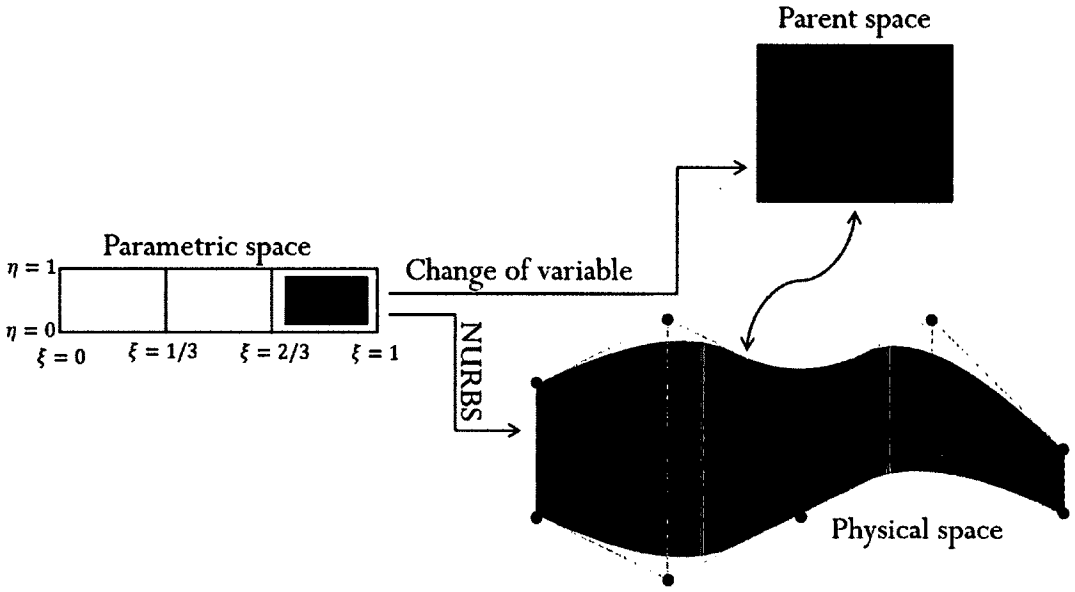


FIGURE. 2.8: Required spaces in Isogeometric Analysis

one-, two-, and three- dimensional spaces, respectively.

Each element in the *physical space* is the image of the corresponding element in the parameter space which can be calculated by the mappings described by NURBS (Eq.2.3.24 and Eq.2.3.28). For a two-dimensional geometric mapping, $\boldsymbol{F}$, defined over a single patch one has:

$$\boldsymbol{F} = \sum_{i=1}^{l_u} \sum_{j=1}^{l_v} R_{i,j}^{p,q}(\xi, \eta) \mathbf{C}_{i,j}, \tag{2.4.5}$$

where l_u and l_v are the number of knots in u and v directions and C_{ij} are control points. Since the same shape functions are used in IGA, the numerical solution is given by [80]:

$$u^h(x, y) = \sum_{i=1}^{l_u} \sum_{j=1}^{l_v} R_{i,j}^{p,q}(\mathbf{F}^{-1}(x, y)) \mathbf{C}_{i,j}. \tag{2.4.6}$$

Integrals over physical domain,Ω, can be transferred to parametric space Ω_0, namely:

$$\int_{\Omega} f(x,y)dxdy = \int_{\Omega_0} f(\mathbf{F}(u,v))|det\mathbf{DF}(u,v)|dudv, \tag{2.4.7}$$

where $\mathbf{DF}(u,v) = \begin{pmatrix} \frac{\partial F_1}{\partial u} & \frac{\partial F_1}{\partial v} \\ \frac{\partial F_2}{\partial u} & \frac{\partial F_2}{\partial v} \end{pmatrix}$ is the Jacobian matrix used for change of variable so the integration is performed on the parent domain.

Comparing IGA with FEM

In FEM, each mesh in the physical space is mapped into a single element in the parent space with a unique mapping. However, in IGA, the B-Spline parameter space is local to patches rather than elements and B-Spline mapping takes a patch of multiple elements from the parameter space into the physical space. In other words, the number of elements in the physical and the parameter spaces are equal but as shown in Fig.2.8 a mapping is global to the patch. Since a single mapping is used over a patch, material models are assumed to be uniform in each patch. The most straight-forward approach to define a physical space with multiple materials is to use a separate patch for each material.

A significant disadvantage of MMs using the Galerkin method is the complexity of the numerical integration of the weak form which is due to the non-polynomial (rational) form of commonly used meshless shape functions. This numerical approach creates difficulties in imposing exact integration. The most common techniques to overcome such difficulties are: cell structure integration, direct nodal integration, stabilized nodal integration, stress point integration, and support-based integration.

These methods are briefly discussed in the following sections.

Cell structure integration (background mesh)

In the cell structure approach, the domain is divided into integration cells over which the method of the Gaussian quadrature is performed [57]:

$$\int_{\Omega} f(X)d\Omega = \sum_{J} f(\boldsymbol{\xi}_J) w_J det\mathbf{J}^{\xi}(\boldsymbol{\xi}) \tag{2.4.8}$$

where $\boldsymbol{\xi}$ are local coordinates and $det\boldsymbol{J}^{\boldsymbol{\xi}}(\boldsymbol{\xi})$ is the determinant of the Jacobian, i.e.the mapping of the parent space into physical space. When implementing background cells, the nodes and vertices of the integration usually coincide (similar to conventional FEM). When cell structures are utilized, a regularly arrayed domain is created, independently of the node position [26]. For a two-dimensional domain, each of these cells is a four node quadrilateral element. Let the corresponding nodal coordinates be denoted as $x_{0I}, (I = 1, 2, 3, 4)$ and the corresponding shape function with N_I. For each Gauss point (ξ_{Gp}, w_{Gp}), the global coordinate, x_{Gp}, is found by the following isoparametric mapping:

$$x_{Gp} = \sum_{I=1}^{4} N_I(\xi_{Gp}) x_I \tag{2.4.9}$$

where $\boldsymbol{J}$ is the Jacobian of the physical to parent space transformation.

Direct nodal integration

In the direct nodal method, the integrals are evaluated only at the nodes, which serve as integration points [57]:

$$\int_{\Omega} f(\mathbf{X})d\Omega = \sum_{J \in S} f(X_J) V_J \tag{2.4.10}$$

The quadrature weights V_J, each represent a volume associated with the node that is obtained from a Voronoi diagram. This method is more efficient than the full integration discussed in Section 2.4.2 but suffers from instabilities due to rank deficiency. This approach is very similar to the collocation methods.

Stabilized nodal integration

Mostly used in MMs, the stabilized nodal method is based on the fact that vanishing derivatives of the shape functions at the nodes are the cause of instabilities [57]. Chen et al. [16] proposed a *strain smoothing* procedure to avoid shape function derivative evaluation at the nodes and therefore eliminate defective modes. In this method the nodal strains are computed as the divergence of the spatial average of the strain field.

The combination of stabilized conforming integration with conventional FEM is called the *smoothed finite element method (SFEM).* This method can provide higher stress accuracy, insensitivity to volumetric locking, and super-convergence. However this method will results in instability in some instances.

Stress point integration

In the method of finite spheres, the integration is performed on every intersection of overlapping support. A truly meshless method for integrating the weak form over overlapping support is called moving least squares quadrature. Another method is Partition of Unity quadrature which is improved to take discontinuities such as cracks into account.

2.5 ASSEMBLY

The assembly in the IGA is very similar to the conventional FEM. In both methods the support of each basis function is localized and therefore the resulting stiffness matrix,$\boldsymbol{K}$, is sparse. This will reduce the amount of work required in building and solving the algebraic system. The assembly of the stiffness matrix and the force vector is done by looping over the elements and building element stiffness matrices. Then each of these elemental stiffness matrices is added to the appropriate spot of the global stiffness matrix through the implemention of the connectivity array. The advantage of this process over calculating a global stiffness matrix is that it avoids the integration of the regions with zero basis functions. The difference in overhead is the additional mapping to parameter space to perform the integration during the assembly of the stiffness matrix in IGA.

In MMs the assembly procedure is performed in the domain of influence of the point under consideration [81], which often serves as a Gauss point as well. For total n nodes, if nodal unknowns are stored as follows:

$$\boldsymbol{u}^T = [u_1 \; v_1 \; u_2 \; v_2 \ldots u_n \; v_n] \tag{2.5.1}$$

the node I will contribute to the $(2I-1)^{th}$ row and $2I^{th}$ column in the global system.

2.6 ESSENTIAL BOUNDARY CONDITIONS

The Kronecker delta property of the conventional FEM shape function provides a straight-forward method for imposing essential boundary conditions. However, as discussed in the previous sections, IGA and MMs shape functions are not

guaranteed to have this property. In this section other methods of imposing essential boundary conditions are briefly discussed.

2.6.1 ESSENTIAL BOUNDARY CONDITIONS-IGA

Homogenous as well as uniform inhomogeneous boundary conditions can be imposed directly in IGA due to the interpolatory property of the first and last knots in open knot vectors as well as the Partition of Unity property. Since NURBS basis are not generally interpolatory and therefore do not satisfy the Kronecker delta property, any other type of Dirichlet boundary condition cannot be imposed directly. Inhomogeneous Dirichlet boundary conditions are usually imposed using Lagrangian multipliers, the Augmented Lagrangian Method, the penalty method and the least square minimization method. A brief description of the Lagrangian multipliers and the penalty methods are presented in Section 2.6.3 and Section 2.6.2 respectively.

Due to the Partition of Unity property of IGA basis functions, the imposition of homogeneous as well as inhomogeneous boundary conditions is straight-forward. For both of these cases, the prescribed boundary conditions are assigned to the corresponding control variables directly. Assuming the use of open knot vectors, NURBS shape functions satisfy the Kronecker delta property only at the corner points. Direct imposition cannot be used to treat non-uniform Dirichlet boundary conditions which vary from point to point due to non interpolatory nature of NURBS basis functions. Therefore, direct application of the inhomogeneous Dirichlet boundary condition values to the control variables may result in significant solution errors.

Implementation of the non-uniform inhomogeneous Dirichlet boundary conditions in IGA is not trivial and it is still a matter of research. Available methods to impose such a boundary condition include the Lagrangian, the least square, the penalty, and the transformation methods, and the projection method.

Transformation Method

The interior NURBS basis functions, which are constructed using open knot vectors, vanish on the boundary. Therefore, it is possible to partition control variables into interior and boundary sets [6]. The unknown solution can be written in terms of the collection of N_B control points representing the boundary and a collection N_I control points representing the interior of the domain, thus,

$$u^h(x) = \sum_{a=1}^{N_I} R_a(\xi,\eta)c_a + \sum_{b=1}^{N_B} R_b(\xi,\eta)c_b \tag{2.6.1}$$

which can be represented in vector form as

$$u^h(x) = \boldsymbol{R}^{I^T}\boldsymbol{c}^I + \boldsymbol{R}^{B^T}\boldsymbol{c}^B \tag{2.6.2}$$

with $\boldsymbol{R}^I$, $\boldsymbol{c}^I$, $\boldsymbol{R}^B$, $\boldsymbol{c}^B$ being the column vectors consisting of interior and boundary basis function and control variables respectively. To obtain the values of the control variables this equation can be collocated at a set of boundary collocation points $\boldsymbol{x}_c$ on Ω, $c = 1, 2, \cdots, N_B$, that is

$$u^h(\boldsymbol{x}_c) = \boldsymbol{R}^I(\boldsymbol{x}_c)\boldsymbol{c}^I + \boldsymbol{R}^B(\boldsymbol{x}_c)\boldsymbol{c}^B = \boldsymbol{R}^B(\boldsymbol{x}_c)\boldsymbol{c}^B, \quad \boldsymbol{x}_c \in \Gamma_D \tag{2.6.3}$$

since the control points may not lay on the problem boundary, the boundary collocation points, $\boldsymbol{x}_c$, may not coincide with the corresponding boundary control points.

In the matrix form one has:

$$\boldsymbol{u}^B = \boldsymbol{T}^T \boldsymbol{c}^B, \quad T_{BC} = R_B(\boldsymbol{x}_c) \quad \boldsymbol{x}_c \in \Gamma \tag{2.6.4}$$

The control variables, $\boldsymbol{c}^B$, are obtained at discretized boundary collocation points using the corresponding physical values, $\boldsymbol{u}^B$. That is

$$\boldsymbol{c}^B = \boldsymbol{T}^{-T} \boldsymbol{u}^B \tag{2.6.5}$$

Note that the collection of boundary collocation points, $\boldsymbol{x}_c$, is flexible and the transformation matrix $\mathbf{T}$ is well defined. Therefore, $\boldsymbol{x}_c$ can be selected as mesh knots, Gaussian integration points, or a set of points obtained through interpolating the boundary. Substituting $\mathbf{c}^B = \mathbf{T}^{-T}\mathbf{u}^B$ into $u^h(\mathbf{x}) = \mathbf{R}^I\mathbf{c}^I + \mathbf{R}^B\mathbf{c}^B$ one can get

$$u^h(\mathbf{x}) = \mathbf{R}^I\mathbf{c}^I + \mathbf{R}^B\mathbf{T}^{-T}\mathbf{u}^B \tag{2.6.6}$$

The Bubnov-Galerkin approximation gives:

$$\delta u^h(\mathbf{x}) = \mathbf{R}^I\delta\mathbf{c}^I + \mathbf{R}^B\mathbf{T}^{-T}\delta\mathbf{u}^B \tag{2.6.7}$$

Substituting, since $\mathbf{u}^B$ is representing physical boundary values, the $\delta\mathbf{u}^B$ would be kinematically admissible and therefore vanishes on the essential boundary $\boldsymbol{u}^h$. Substituting the $\delta\boldsymbol{u}^h$ into the weak form of the problem yields a set of linear equations:

$$\bar{\mathbf{K}}\mathbf{d} = \bar{\mathbf{f}} \tag{2.6.8}$$

Since the resulting linear system described by Eq.2.6.8 is different from the one obtained by the conventional FEM, this method of applying Dirichlet boundary condition will change the conventional structure of FEM.

Quasi interpolation (projection) Method

Another method is to impose the Dirichlet boundary condition is called quasi interpolation method [20,43,78]. This method offers low computational cost approximation to a given set of data or a given function. Quasi-interpolation is well-studied in approximation theory [13,47,48]. In order to apply the Dirichlet boundary conditions, a proper approximation of the boundary data $\boldsymbol{g}$, called $\boldsymbol{g}_h$ is used. Since in IGA, any physical boundary is a map of parameter domain boundary, the boundary can be described as

$$\begin{aligned} \Gamma_D &= F(\hat{\Gamma}_D), \\ \boldsymbol{x} = F(\boldsymbol{u},\boldsymbol{v}) &= \sum_{i=1}^{n}\sum_{j=1}^{m} N_{i,p}(\xi)M_{j,q}(\eta)\boldsymbol{c}_{ij} = \sum_{k=1}^{(j-1)n+i} R_k(\xi,\eta)\boldsymbol{c}_k, \end{aligned} \tag{2.6.9}$$

where

$$\boldsymbol{u} := \{0 = \xi_1 = \cdots, \xi_p < \cdots < \xi_{n+1} = \cdots = \xi_{n+p}\}$$

$$\boldsymbol{v} := \{0 = \eta_1 = \cdots, \eta_q < \cdots < \eta_{m+1} = \cdots = \eta_{m+q}\}$$

also $\hat{\Gamma}_D$ is representing the parameter domain boundary of the Dirichlet sides. Consider a two dimensional domain such as a unit rectangle in which side-1, corresponding to $\eta = 0$, is a Dirichlet side. The parameter domain of boundary can be represented as:

$$\Gamma_{1D} = F(\xi, 0) \tag{2.6.10}$$

by denoting the basis function for Dirichlet side with $\Phi_i(\boldsymbol{x})|_{\Gamma_{1D}}$, one has:

$$\Phi_k(\boldsymbol{x})|_{\Gamma_{1D}} = N_k \circ F^{-1}(\boldsymbol{x})|_{\Gamma_{1D}} = N_k(\xi, 0). \tag{2.6.11}$$

For control variable dedicated to Dirichlet degrees of freedom in Galerkin discretization $k = n_{eq+1} \dots n_{np}$ one has:

$$\boldsymbol{x} = F(\xi, 0), \tag{2.6.12}$$

also,

$$N_k(\xi, 0) = N_{i,p}(\xi) M_{j,q}(0) = \begin{cases} N_{i,p}(\xi) & j = 1, \\ 0 & \text{otherwise} \end{cases} \tag{2.6.13}$$

thus, $g_D^h|_{\Gamma_1}$ can be expressed in the form:

$$g_D^h|_{\Gamma_1} \circ F(., 0) := \sum_{i=1}^{n} g_i N_{i,p}(\xi) \tag{2.6.14}$$

the unknown g_k can be obtained by solving following interpolation problem:

$$(g_k)\big(F(s_k^p, 0)\big) = (g_D)\big(F(s_k^p, 0)\big), \quad i = 1, \dots, n, \tag{2.6.15}$$

where s_k^p is a set of suitable evaluation points such as the abscissas of the associated control polygon or the quadrature points. Note that this system of linear functions achieves full approximation power of the underlying space. This system of linear equation should be solved for all Dirichlet sides simultaneously. Using this method one can identify Dirichlet degrees of freedom on all Dirichlet sides and compute the stated value in quadrature points. Equation 2.6.15 can be solved for g_k by constructing the following integral:

$$\int_{\Gamma_D} g_k \, v \, d\Gamma_D = \int_{\Gamma_D} g_D \, v \, d\Gamma_D \tag{2.6.16}$$

Imposing essential boundary conditions in MMs is even more challenging. This is due to the lack of the Kronecker delta property of MMs shape functions.

To overcome this shortcoming, several methods have been proposed, which can be classified in two categories [57]:

1. methods based on the modification of the weak form, such as the Lagrange multiplier and the penalty methods

2. methods using modified shape functions [6]

A variational principle can be symbolically defined by an integral:

$$\prod = \int_{\Omega} F(\boldsymbol{u})\ d\Omega + \int_{\Gamma} \lambda E(\boldsymbol{u}) d\Gamma \tag{2.6.17}$$

where u is the unknown function and F and E are differential operators and λ is the Lagrangian multiplier. The solution of the continuum problem is a function of u that makes $\prod$ stationary with any arbitrary variations δu. In other words, $\delta \prod = 0$ with any variation δu.

2.6.2 LAGRANGE MULTIPLIERS

The solution that makes the functional $\prod$ stationary subject to the constraints:

$$\mathbf{C}(\mathbf{u}) = 0 \quad on\ \Omega \tag{2.6.18}$$

will make the following functional stationary:

$$\bar{\prod}(\boldsymbol{u}, \boldsymbol{\lambda}) = \prod(\boldsymbol{u}) + \int_{\Gamma} \boldsymbol{\lambda}^T \boldsymbol{C}(u)\ d\Gamma \tag{2.6.19}$$

where $\boldsymbol{\lambda}(x)$ are Lagrange multipliers. To derive the corresponding discrete equations, the Lagrange multipliers must be approximated as:

$$\lambda(\boldsymbol{x}) = \sum_{I=1}^{l} N_I^L(x) \lambda_I \tag{2.6.20}$$

The approximation space for Lagrange multipliers ($N_I^L(\boldsymbol{x})\lambda_I$) can be the finite element approximation of the boundary Γ, the meshless approximation on this boundary or the point collocation method that uses the Dirac delta function:

$$N_I^L(\boldsymbol{x}) = \delta(\boldsymbol{x} - \boldsymbol{x}_I^L) \tag{2.6.21}$$

where $\boldsymbol{x}_I^L$ is a set of points located along the boundary Γ. The obvious drawback of the Lagrange multipliers method is the introduction of additional unknowns to the problem and added zero terms on the diagonal of the coefficient matrix in the final matrix equations, which makes the matrix no longer positive-definite. The Augmented Lagrangian method is suggested by Ventura [78] for meshless methods. This method is effective and stable, especially in contact problems.

2.6.3 PENALTY FUNCTIONS

The advantage of the penalty-based method over the Lagrange multipliers method is that no additional unknown are added to the problem. However the conditioning of the matrix depends on the choice of the penalty parameter. Also in this method, constraints are satisfied approximately [57]. In this study, the quasi interpolation method, which is described in Section 2.6.1, is used to apply the inhomogeneous boundary conditions.

2.7 REFINEMENT

The purpose of refinement is to achieve higher accuracy in numerical solution and to confirm the convergence of the solution. Increasing the accuracy is possible by increasing the number of elements or by increasing the order of shape function used to estimate the solution domain. As previously discussed, these two methods of increasing accuracy have resulted in two versions of FEM, which will be compared with each other in the following sections. Similar refinement methods are also used for the IGA and the MMs, which are described in this Section 2.7.2.

2.7.1 REFINEMENT IN FEM

The h-method improves the solution accuracy by adding more elements, while the p-method improves accuracy by increasing the order of the shape functions. It is desirable to only increase the number of elements (h-method) or polynomial order (p-method) in the complex areas of the domain to achieve more accurate results that will add a minimal computational cost.

Comparing h- and p-methods

The differences between h and p version can be summarized as follows [3]:

1. *Asymptotic rate of convergence (in the energy norm)* with respect to the number of degrees of freedom:

    - In contrast to h-method FEM, the rate of convergence of the p-method FEM is not limited by a fixed polynomial degree.

- In the case of non-smooth solutions, when the h-version is based on quasi-uniform mesh refinement, the rate of convergence of the h-method FEM is similar to p-method FEM. However, when a singularity exists in the corners the rate of convergence of the p-version is twice as of h-version.

2. *Optimal mesh design* can be implemented with h-method to obtain a higher convergence rate. However the mesh design will not significantly affect the p-method convergence rate.

3. *The size of input data* is smaller in the p-method than for the h-method which is the direct result of smaller number of mesh cells.

4. *Flexibility*, which is less for the p-method than the h-method due to the *assumption of constant coefficients* over large finite element domains.

5. *Solution time* is similar in both versions for a given number of degrees of freedom.

2.7.2 REFINEMENT IN IGA

In the IGA, the basis function may be enriched by refinement without changing the geometry and parameterization [21]. Refinement in the IGA is considered superior to its FEM counterpart because the element size and the order of the basis as well as the continuity of the basis can be refined.

Knot insertion

Knot insertion is based on providing additional knots to the knot vector to increase the number of elements. Efficient algorithms are available in the Piegle and Tiller reference [61]. This refinement does not affect the geometry and the parameter space of the problem. First, an extended knot vector is introduced and then the new set of control points is calculated as a linear combination of the original control points. If a knot is repeated in the extended knot vector, the continuity of the basis functions will be reduced. However the continuity of the curve can be preserved [21]. Knot refinement is analogous to h-refinement in FEM but not identical. The main difference is the possibility of decreasing the continuity without adding elements, simply by repeating a repeated knot value.

Degree elevation

Degree elevation refinement is done by increasing the order of the NURBS basis function. To preserve the discontinuities of the model, the number of repeated knots should be increased by one for each degree of order elevation. No new knot is added in this process and the geometry and parameterization of the problem remains unchanged. This method is analogous to p-refinement in FEM, however unlike p-refinement which always begins with C^0 basis, order elevation can be applied to any combination of continuities in the B-Spline mesh [21].

The order elevation has three major steps. The first step is increasing the multiplicity of the knots to the degree of the polynomial, which is equivalent to dividing

the curve into separate elements. The second step involves increasing the order of the polynomials. Finally, the third step is the removal of excess knots to combine individual curve into one curve. This process results in increasing the number of basis functions, which are C^0 across element boundaries. Efficient algorithms are available in [61].

k-refinement

k-refinement is a combination of degree evaluation and knot insertion and is only feasible for IGA. First, the order of the basis is increased and then additional knots are inserted. The extent of knot insertion is dependent on the order of polynomials. This method results in a limited increase in the number of basis functions compared to p-method, where each function becomes C^{P-1}-continuous across element boundaries.

Domain refinement in MMs

H-adaptivity is easier to incorporate in meshless methods due to the absence of a mesh. This characteristic also holds true for r-adaptivity, in which a node is relocated in the desired region, and p-adaptivity as well.

2.8 TIME-DEPENDENT PROBLEMS

There are two broad approaches for computing the solution of a time-dependent problem. In the first approach, the *semi-discrete method*, time is considered continuous and space is discretized. In the second, *space-time finite elements*,

both time and space are discretized.

2.8.1 SEMI-DISCRETE METHODS

The Galerkin method is used for the discretization of the geometry while time is modeled as continuous. The shape functions are defined as a function of the spatial domain only, in a manner similar to the static case, and corresponding coefficients are defined as functions of time. As a result, the solution is a linear combination of space-dependent shape functions and time- dependent coefficients. The function space of test functions, V, is similar to the static case but the function space of trial functions is time-dependent, satisfying the dynamic conditions [21]:

$$S_t = \{\mathbf{u}(.,t) | u_i(\mathbf{x},t) = g_i(\mathbf{x},t)\ , \mathbf{x} \in \Gamma_D\ , u(.,t) \in H^1(\Gamma)\} \tag{2.8.1}$$

In this method, integration is done with respect to spatial coordinates only.

Predictor/ multicorrector Newmark Algorithms

To solve a semi-discrete equation in the form of $\ddot{\boldsymbol{d}} + \boldsymbol{C}\dot{\boldsymbol{d}} + \mathbf{K}\mathbf{d} = \boldsymbol{F}$, a time step,$\Delta t$, is defined and an iterative process is used to calculate acceleration $a_{n+1} \approx \ddot{d}(t_{n+1})$, velocity $v_{n+1} \approx \dot{d}(t_{n+1})$, and displacement $v_{n+1} \approx d(t_{n+1})$ at each time step. This iteration starts ($n = 0$) with a predictor phase in which the approximation is initialized:

$$\mathbf{d}^i_{n+1} = \tilde{\mathbf{d}}_{n+1} \tag{2.8.2}$$

$$\mathbf{v}^i_{n+1} = \tilde{\mathbf{v}}_{n+1} \tag{2.8.3}$$

$$\mathbf{a}^i_{n+1} = \tilde{\mathbf{a}}_{n+1} \tag{2.8.4}$$

This initialization should remain consistent to the Newmark formulas:

$$\tilde{\mathbf{d}}_{n+1} = \mathbf{d}_n + \Delta t \mathbf{v}_n + \frac{(\Delta t)^2}{2}\big((1-2\beta)\mathbf{a}_n + 2\beta\tilde{\mathbf{a}}_{n+1}\big) \tag{2.8.5}$$

$$\tilde{\mathbf{v}}_{n+1} = \mathbf{v}_n + \Delta t\big((1-\gamma)\mathbf{a}_n + \gamma\ \tilde{\mathbf{a}}_{n+1}\big), \tag{2.8.6}$$

where β and γ are parameters. Commonly used predictors are listed below.

1. *Constant-displacement predictor*: mostly used in nonlinear solid mechanics, contact problems and problems involving large deformations.

$$\tilde{\mathbf{d}}_{n+1} = \mathbf{d}_n, \tag{2.8.7}$$

$$\tilde{\mathbf{v}}_{n+1} = \mathbf{v}_n + \Delta t\big((1-\gamma)\mathbf{a}_n + \gamma\ \tilde{\mathbf{a}}_{n+1}\big), \tag{2.8.8}$$

$$\tilde{\mathbf{a}}_{n+1} = -\frac{1}{\beta\Delta t}\mathbf{v}_n - \frac{1-\beta}{2\beta}\mathbf{a}_n \tag{2.8.9}$$

2. *Constant-velocity predictor*: mostly used in problems of fluid mechanics and fluid structure interaction.

$$\tilde{\mathbf{d}}_{n+1} = \mathbf{d}_n + \Delta t \mathbf{v}_n + \frac{(\Delta t)^2}{2}\big((1-2\beta)\mathbf{a}_n + 2\beta\tilde{\mathbf{a}}_{n+1}\big) \tag{2.8.10}$$

$$\tilde{\mathbf{v}}_{n+1} = \mathbf{v}_n \tag{2.8.11}$$

$$\tilde{\mathbf{a}}_{n+1} = -\frac{(1-\gamma)}{\gamma}\mathbf{a}_n \tag{2.8.12}$$

3. *Zero-acceleration predictor*:: preferred in linear analysis.

$$\tilde{\mathbf{d}}_{n+1} = \mathbf{d}_n + \Delta t\ \mathbf{v}_n + \frac{(\Delta t)^2}{2}(1-2\beta)\mathbf{a}_n \tag{2.8.13}$$

$$\tilde{\mathbf{v}}_{n+1} = \mathbf{v}_n + \Delta t(1-\gamma)\mathbf{a}_n \tag{2.8.14}$$

After selecting the predictor type, the residual value of the equation is calculated at time t_{n+1}:

$$\Delta \mathbf{F}^i_{n+1} = \mathbf{F}_{n+1} - \mathbf{M}_{n+1}\ \mathbf{a}^i_{n+1} - \mathbf{C}_{n+1}\ \mathbf{v}^i_{n+1} - \mathbf{K}_{n+1}\ \mathbf{d}^i_{n+1} \tag{2.8.15}$$

Then the correction to the acceleration term is calculated by solving

$$\mathbf{M}^*_{n+1}\ \Delta a = \Delta F^i_{n+1} \tag{2.8.16}$$

where M^* is calculated from the following equation:

$$\mathbf{M}^*_{n+1} = \mathbf{M}_{n+1} + \gamma\ \Delta t \mathbf{C}_{n+1} + \beta(\Delta t)^2 \mathbf{K}_{n+1} \tag{2.8.17}$$

where the values of β and γ affect the properties of this method; i.e. By maintaining $2\beta \geq \gamma \geq \frac{1}{2}$, this method will be unconditionally stable, while setting $\gamma = \frac{1}{2}$ will result in second-order accuracy and conditional stability [36]:

$$\gamma \geq \frac{1}{2} \tag{2.8.18}$$

$$\beta \leq \frac{\gamma}{2} \tag{2.8.19}$$

$$\frac{\Delta t}{T} \leq \frac{\Omega_{critical}}{2\pi} \tag{2.8.20}$$

where

$$\Omega_{critical} = \frac{\xi(\gamma - \frac{1}{2}) + [\frac{\gamma}{2} - \beta + \xi^2(\gamma - \frac{1}{2})^2]^{\frac{1}{2}}}{(\frac{\gamma}{2} - \beta)} \tag{2.8.21}$$

After obtaining Δa , a corrector phase is used to update the solution:

$$\mathbf{a}^{i+1}_{n+1} = \mathbf{a}^i_{n+1} + \Delta \mathbf{a}, \tag{2.8.22}$$

$$\mathbf{v}^{i+1}_{n+1} = \mathbf{v}^i_{n+1} + \gamma\ \Delta t\ \Delta \mathbf{a}, \tag{2.8.23}$$

$$\mathbf{d}_{n+1}^{i+1} = \mathbf{d}_{n+1}^{i} + \beta\ (\Delta t)^2\ \Delta \mathbf{a}, \tag{2.8.24}$$

Then the residual is calculated and checked for convergence using a user defined tolerance ε:

$$\| \Delta F_{n+1}^{i} \| \leq \varepsilon \| \Delta F_{n+1}^{0} \| \tag{2.8.25}$$

If convergence is not achieved, a new iteration begins from Eq.2.8.4 for the same time step; otherwise the evaluation at the next time step begins. For convergence analysis, the reader is referred to the reference by Hughes [36]. There are several well-known cases in the Newmark family:

- *Average acceleration method* ($\beta = 1/4, \gamma = 1/2$) : an implicit method providing unconditional stability and widely used in structural dynamics.
- *Linear acceleration method* ($\beta = 1/6, \gamma = 1/2$) : an implicit method with conditional stability, $\Omega_{critical} = 2\sqrt{3}$.
- *Fox-Goodwin method* ($\beta = 1/12, \gamma = 1/2$) : an implicit method with conditional stability, $\Omega_{critical} = \sqrt{6}$.
- *Central differences method* ($\beta = 0, \gamma = 1/2$) : an explicit method (M and C need to be diagonal) with conditional stability, $\Omega_{critical} = 2$.

The average acceleration method is implicit and unconditionally stable. This technique is relatively computationally inexpensive for large scale systems. The central differences method is considered as the most economical technique and is widely used in literature. The only limitation of this method is the time step restriction.

2.8.2 SPACE-TIME FINITE ELEMENTS

In the space-time finite element method, both the space and the time are discretized. To represent discretized space and time, the shape function is defined as a tensor product of the basis describing the geometry and a one-dimensional basis describing the time. Therefore the shape function used to describe a d-dimensional space will be $(d+1)$-dimensional. This means that an additional space has been added to the meshing to enable the computation of a transient solution as indicated by Eq.2.8.20 as it is applicable to Newmark method. The size of mesh in time is limited in order to ensure a stable and convergent solution. This added dimension considerably increases the computational cost. To reduce this cost, time is partitioned, enabling an analysis based on a sequence of space-time slabs [21].

CHAPTER 3

ELECTROMAGNETISM

The electro-magnetic fields were originally introduced by means of the force equations. A force acting between localized charges was studied by Coulomb. Similarly, the mutual forces of current-carrying loops were studied by Ampere. Introduction of electric and magnetic fields ($\boldsymbol{E}$ and $\boldsymbol{B}$) conceptually decouples the charges from the bodies experiencing the forces. Note that the electro-magnetic field can exist in regions of space where there are no forces. Provided the charge and current sources, Maxwell's equations describe the relation between electric and magnetic fields and the sources in the medium.

3.1 TIME DOMAIN MAXWELLS DIFFERENTIAL EQUATIONS

Faraday's law:

$$\frac{\partial \boldsymbol{B}}{\partial t} = -\nabla \times \boldsymbol{E} \tag{3.1.1}$$

Ampere's law:

$$\frac{\partial \boldsymbol{D}}{\partial t} = \nabla \times \boldsymbol{H} - \boldsymbol{J} \tag{3.1.2}$$

Gauss's law for electric field:

$$\nabla \cdot \boldsymbol{D} = 0 \tag{3.1.3}$$

Gauss's law for magnetic field:

$$\nabla \cdot \boldsymbol{B} = 0 \tag{3.1.4}$$

The field variable are defined as:

$\boldsymbol{E}$	*electric field intensity*	$[Volts/meter; Vm^{-1}]$
$\boldsymbol{H}$	*magnetic field intensity*	$[amperes/meter; Am^{-1}]$
$\boldsymbol{B}$	*magnetic flux density*	$[Tesla; T]$
$\boldsymbol{D}$	*electric displacement*	$[coulombs/m^2; Cm^{-2}]$
$\boldsymbol{J}$	*electric current density*	$[amperes/m^2; Am^{-2}]$
$\boldsymbol{\rho}$	*electric charge density*	$[coulombs/m^3; Cm^{-3}]$

According to Eq.3.1.4, the divergence of the magnetic flux density, $\boldsymbol{B}$ is zero and therefore the magnetic flux density is not expanding or contracting in space. Similarly, Eq.3.1.3 shows the divergence-free property of the electrical displacement, $\boldsymbol{D}$. Equation 3.1.1 shows that any change of magnetic flux density $\boldsymbol{B}$ in time will cause the electric field $\boldsymbol{E}$ to move in space which can be visualized as a current. Finally, Eq.3.1.2 explains how a time variation of electric displacement or the field $\boldsymbol{E}$ can be related to the space displacement of the magnetic field, $\boldsymbol{H}$ and to the current density $\boldsymbol{J}$. Maxwell's equations are composed of one scalar and five vector quantities. These equations are related as described by the following constitutive equations which relates $\boldsymbol{E}$ and $\boldsymbol{H}$ fields to the corresponding fluxes $\boldsymbol{D}$ and $\boldsymbol{B}$ as well as the impressed current $\boldsymbol{J}^{imp}$ and the free current $\boldsymbol{J}$:

$$\boldsymbol{D} = \varepsilon \boldsymbol{E} \tag{3.1.5}$$

$$\boldsymbol{B} = \mu \boldsymbol{H} \tag{3.1.6}$$

$$\boldsymbol{J} = \boldsymbol{J}^{imp} + \sigma \boldsymbol{E} = \rho \boldsymbol{v} \tag{3.1.7}$$

where

ε	permittivity	$[Farads/meter; Fm^{-1}]$
μ	permeability	$[Henries/meter; Hm^{-1}]$
σ	conductivity	$[Siemens/meter; Siemens \cdot m^{-1}]$

also, v in Eq.3.1.7 is the velocity of the net charge density. In general, constitutive equations Eqs. 3.1.5, 3.1.6, and 3.1.7 might be nonlinear (ferromagnetic materials) and may depend on the history (hysteresis). For anisotropic media these parameters become tensors. Also, they can be functions of position, frequency, and field intensities. In a lossy medium, electric and magnetic fields attenuate $\boldsymbol{E}$ and $\boldsymbol{H}$ via conversion to heat energy similar to resistive heating of a wire. From Eqs. 3.1.1 and 3.1.6 one has:

$$\frac{\partial \boldsymbol{H}}{\partial t} = -1/\mu \nabla \times \boldsymbol{E} \tag{3.1.8}$$

also, from Eqs.3.1.2, 3.1.5, and 3.1.7 one has:

$$\frac{\partial \boldsymbol{E}}{\partial t} = \frac{1}{\varepsilon} \nabla \times \boldsymbol{H} - \frac{1}{\varepsilon} (\boldsymbol{J}^{imp} + \sigma \boldsymbol{E}) \tag{3.1.9}$$

which yields; assuming that the material properties are constant,

$$\varepsilon \frac{\partial^2 \mathbf{E}}{\partial t^2} = \nabla \times (-\frac{1}{\mu} \nabla \times \mathbf{E}) - \frac{\partial (\boldsymbol{J}^{imp} + \sigma \boldsymbol{E})}{\partial t} \tag{3.1.10}$$

or

$$\varepsilon \boldsymbol{E}_{,tt} + \sigma \boldsymbol{E}_{,t} + \nabla \times (1/\mu \nabla \times \boldsymbol{E}) = -\boldsymbol{J}^{imp}_{,t} \tag{3.1.11}$$

Equation 3.1.11 is called Maxwell equation in second order form and commonly solved as a time dependent electromagnetic equation [54]. Computational electromagnetism is commonly performed using the edge finite elements in which the normal component of the computed solution is discontinuous. Such a discontinuity will result in a discontinuous solution in physical domain, which is considered useful to represent the change of material in the domain. In IGA, a separate patch can be considered to represent each material.

3.2 TIME-HARMONIC MAXWELL EQUATIONS

The time-harmonic field $\boldsymbol{E}$ has the general form:

$$\boldsymbol{E}(\boldsymbol{x},t) = \boldsymbol{A}(\boldsymbol{x})\ cos(\omega t) + \boldsymbol{B}(\boldsymbol{x})\ sin(\omega t) \tag{3.2.1}$$

where ω is the angular frequency and $\boldsymbol{A}(\boldsymbol{x})$ and $\boldsymbol{B}(\boldsymbol{x})$ are real vector fields which are considered position-dependent but time-independent. Note that $\boldsymbol{E}$ can be written as the real part of its complex form $\boldsymbol{E}_c = (\boldsymbol{A}(\boldsymbol{x}) - j\boldsymbol{B}(\boldsymbol{x}))\ e^{j\omega t}$ at a single frequency as follows:

$$\boldsymbol{E}(\boldsymbol{x},t) = \Re(\boldsymbol{E}_c(\boldsymbol{x})) \tag{3.2.2}$$

where $\boldsymbol{E}_c(\boldsymbol{x})$ is the unknown complex-valued phasor which is only a function of position $\boldsymbol{x}$ and the frequency ω. The phasor can be viewed as the Fourier transformation of the real-valued electrical field $\mathbf{E}(\mathbf{x},\mathbf{t})$ in time. Consequently, the $j\omega$ will appear in time derivatives of the transient Maxwell equations described in the previous section. For simplicity $\boldsymbol{E}_c$ is denoted with $\boldsymbol{E}$ hereafter. Hence, time-harmonic Maxwell

equation can be written as:

$$\nabla \times \boldsymbol{E} = -j\omega \boldsymbol{B} \qquad \text{Faraday's law,} \qquad (3.2.3a)$$

$$\nabla \times \boldsymbol{H} = \boldsymbol{J} + j\omega \boldsymbol{D} \qquad \text{Ampere's law,} \qquad (3.2.3b)$$

accompanied with Gauss's laws:

$$\nabla \cdot \boldsymbol{D} = 0, \qquad (3.2.4a)$$

$$\nabla \cdot \boldsymbol{B} = 0. \qquad (3.2.4b)$$

The Maxwell equations are overdetermined. Constitutive equations presented in equations (3.1.5, 3.1.6) can be used to eliminate $\boldsymbol{D}$, $\boldsymbol{B}$, and $\boldsymbol{J}$, resulting in only two vector unknowns: $\boldsymbol{E}$, and $\boldsymbol{H}$. The electeric field E is subjected to the divergance equation $\nabla \cdot \boldsymbol{E} = 0$ bacause of Eqs.3.1.3 and 3.1.5.This divergence equations is not considered directly in numerical analysis. However, eliminating divergence equations in numerical calculations is known to generate spurious modes and inaccurate solutions in computational electro-magnetics. To overcome this shortcoming, the discretized spaces should be modified to $\boldsymbol{H}(curl)$-conforming elements satisfying the divergence free conditions of the electric and magnetic field implicitly. B-Spline discretization for electromagnetic field introduced and extensively described in [12]. The curl-conserving transforms in this study are applied by following the construction presented in [23].

Second Order Wave Equations: The second order Wave equations which are also called the *curl-curl* form can be formulated by combining Eq. 3.2.3a, and Eq.3.2.3b

as follows:

$$\nabla \times \left(\frac{1}{\mu}\nabla \times \boldsymbol{E}\right) - (\omega^2\varepsilon - j\omega\sigma)\boldsymbol{E} = -j\omega\boldsymbol{J}^{imp}. \tag{3.2.5}$$

In this study, equations are formulated with respect to electrical field. The magnetic field can be estimated by using Faraday's law described in Eq.3.2.3a.

3.3 VARIATIONAL FORMULATION

To obtain the variational formulation, it is necessary to combine Ampere's law and Faraday's law. The electric field formulation is generated by multiplying a test function $\boldsymbol{F}$ to the Ampere's equation 3.2.3b and integration over the domain Ω. Therefore, the Ampere's law is weakly satisfied while the Faraday's law is imposed strongly. Integrating by parts one can find the following equation:

$$\int_\Omega \boldsymbol{H}\cdot\nabla\times\boldsymbol{F}d\Omega + \int_\Gamma(\mathbf{n}\times\boldsymbol{H})\cdot\mathbf{F}d\Gamma = \int_\Omega \boldsymbol{J}^{imp}\cdot\boldsymbol{F}d\Omega + \int_\Omega(\sigma+j\omega\varepsilon)\boldsymbol{E}\cdot\boldsymbol{F}d\Omega. \tag{3.3.1}$$

Then, the Faraday's law is imposed point-wise to remove the magnetic field $\mathbf{H}$:

$$\boldsymbol{H} = -\frac{1}{j\omega\mu}(\nabla\times\boldsymbol{E}), \tag{3.3.2}$$

Substituting Eq.3.3.2 into Eq. 3.3.1 yields:

$$\begin{aligned}\int_\Omega \frac{1}{\mu}\nabla\times\boldsymbol{E}\cdot\nabla\times\boldsymbol{F}d\Omega - \int_\Omega(\omega^2\varepsilon - j\omega\sigma)\boldsymbol{E}\cdot\boldsymbol{F}d\Omega \\ = -j\omega\int_\Omega \boldsymbol{J}^{imp}\cdot\boldsymbol{F}d\Omega + j\omega\int_\Gamma(\mathbf{n}\times\boldsymbol{H})\cdot\boldsymbol{F}d\Gamma\end{aligned} \tag{3.3.3}$$

This variational equation can be rewritten using relative parameters. The free space parameters are:

$$\varepsilon_0 = \text{free space permittivity } (8.854\times10^{-12}F/m)$$

$$\mu_0 = \text{free space permeability } (4.0\pi \times 10^{-7} H/m)$$

The relative parameters are defined as follows:

$$\varepsilon_r = \frac{\varepsilon}{\varepsilon_0}, \quad \varepsilon_\sigma = \frac{\sigma}{\omega\varepsilon_0}, \quad \mu_r = \frac{\mu}{\mu_0}, \quad k_0 = \omega\sqrt{\varepsilon_0\mu_0}, \quad Z_0 = \sqrt{\frac{\mu_0}{\varepsilon_0}}, \tag{3.3.4}$$

recasting Eq. 3.3.3 in terms of free-space wave number, k_0, and free-space impedance, Z_0, one can get:

$$\begin{aligned} &\int_\Omega \frac{1}{\mu_r} \nabla \times \boldsymbol{E} \cdot \nabla \times \boldsymbol{F} d\Omega - \int_\Omega k_0^2 (\varepsilon_r - j\varepsilon_\sigma) \boldsymbol{E} \cdot \boldsymbol{F} d\Omega \\ &= -jk_0 Z_0 \int_\Omega \boldsymbol{J}^{imp} \cdot \boldsymbol{F} d\boldsymbol{x} + jk_0 Z_0 \int_\Gamma (\boldsymbol{n} \times \boldsymbol{H}) \cdot \boldsymbol{F} d\Gamma. \end{aligned} \tag{3.3.5}$$

Two types of boundary conditions are very common in computational electromagnetics namely the Dirichlet and the Neumann boundary conditions.

Perfect Electric Conductor: The Dirichlet boundary condition is used when the boundary is assumed to be a perfect conductor. In a perfect conductor the electric field can not hold therefore, the tangential component of the electric field should be zero:

$$\boldsymbol{n} \times \boldsymbol{E} = 0 \quad \text{on} \quad \Gamma_D \tag{3.3.6}$$

Impressed surface current: The Neumann boundary conditions can be used to describe the impressed surface current on an antenna:

$$\boldsymbol{n} \times \boldsymbol{H} = \boldsymbol{J}_s^{imp} \quad \text{on} \quad \Gamma_N \tag{3.3.7}$$

The boundary condition at the interface of two media can be expressed as:

$$\boldsymbol{n} \times (\boldsymbol{E}_1 - \boldsymbol{E}_2) = 0, \tag{3.3.8}$$

where $\boldsymbol{n}$ is a unit normal vector from medium 1 to medium 2, ρ is the imposed interface charge density and $\boldsymbol{J}_s^{imp}$ is the imposed electric current density on the

boundary. Assuming that the domain boundary $\partial\Omega$ is composed of disjoint Dirichlet, Γ_D, and Neumann, Γ_N, boundaries, the variational formulation described in 3.3.5 can be re-casted as:

$$a(\boldsymbol{E},\boldsymbol{F}) = l(\boldsymbol{F}), \quad \forall \boldsymbol{F} \in \boldsymbol{H}(curl,\Omega) \quad \text{and} \quad \boldsymbol{n}\times\boldsymbol{F} = 0 \quad \text{on} \quad \Gamma_D \tag{3.3.9}$$

where

$$\boldsymbol{E} \in \boldsymbol{H}(curl,\Omega) \quad \text{and} \quad \boldsymbol{n}\times\boldsymbol{E} = 0 \quad \text{on} \quad \Gamma_D, \tag{3.3.10a}$$

$$a(\boldsymbol{E},\boldsymbol{F}) = \int_\Omega \Big(\frac{1}{\mu_r}\nabla\times\boldsymbol{E}\cdot\nabla\times\boldsymbol{F} - k_0^2(\varepsilon_r - j\epsilon_\sigma)\boldsymbol{E}\cdot\boldsymbol{F}\Big)d\Omega, \tag{3.3.10b}$$

$$l(\boldsymbol{F}) = -jk_0Z_0\int_\Omega \boldsymbol{J}^{imp}\cdot\boldsymbol{F}d\Omega + jk_0Z_0\int_{\Gamma_N}\boldsymbol{J}_s^{imp}\cdot\boldsymbol{F}d\Gamma. \tag{3.3.10c}$$

where $\boldsymbol{H}(\mathbf{curl},\Omega)$ is defined as:

$$\boldsymbol{H}(\mathbf{curl},\Omega) := \{\boldsymbol{u}\in L^2(\Omega)|\nabla\times\boldsymbol{u}\in L^2(\Omega)\} \tag{3.3.11}$$

Please refer to Eq. 2.1.4 for the definition of Sobolev spaces.

3.3.1 NON-DIMENSIONALIZATION

The variational equation can be rewritten in terms of non-dimensional variables defined as follows:

$$\boldsymbol{x} := \frac{\boldsymbol{x}}{a}, \quad \boldsymbol{\omega} := k_0 a, \quad \boldsymbol{E} := \frac{\boldsymbol{E}}{E_0}, \quad \boldsymbol{J}^{imp} := \frac{aZ_0}{E_0}\boldsymbol{J}, \quad \boldsymbol{J}_s^{imp} := \frac{Z_0}{E_0}\boldsymbol{J}_s, \tag{3.3.12}$$

where a and E_0 are the characteristic length and the electric field intensity respectively. The size scale is determined based on the size of the domain under consideration and the intensity scale is determined based on the intensity of the incident electric field. For instance, if finding the electric field in a human brain section is

desired, one can determine a and E based on the size of the head and the intensity of the incident electric field. In this configuration, the non-dimensional angular frequency, ω, coincides with the non-dimensional free space number, k. The non-dimensionalized second-order wave equation becomes:

$$\nabla \times \left(\frac{1}{\mu_r}\nabla \times \boldsymbol{E}\right) - \omega^2(\varepsilon_r - j\varepsilon_\sigma)\boldsymbol{E} = -j\omega \boldsymbol{J}^{imp} \tag{3.3.13}$$

The variational form can be obtained from Eq.3.3.10 as follows:

$$\boldsymbol{E} \in \boldsymbol{H}(curl, \omega) \quad \text{and} \quad \boldsymbol{n} \times \boldsymbol{E} = \boldsymbol{n} \times \boldsymbol{E}_D \quad \text{on} \quad \Gamma_D, \tag{3.3.14a}$$

$$a(\boldsymbol{E}, \boldsymbol{F}) = \int_\Omega \left(\frac{1}{\mu_r}\nabla \times \boldsymbol{E} \cdot \nabla \times \boldsymbol{F} - \omega^2(\varepsilon_r - j\varepsilon_\sigma)\boldsymbol{E} \cdot \boldsymbol{F}\right) d\Omega, \tag{3.3.14b}$$

$$l(\boldsymbol{F}) = -j\omega \int_\Omega \boldsymbol{J}^{imp} \cdot \boldsymbol{F} d\boldsymbol{x} + j\omega \int_{\Gamma_N} \boldsymbol{J}_S^{imp} \cdot \boldsymbol{F} d\Gamma. \tag{3.3.14c}$$

3.4 ELECTRO-MAGNETIC WAVE SCATTERING

In electro-magnetic scattering problems when a EM wave, called an incident wave, strikes the interface of two media, a part of its energy is reflected (scattered wave) and another part is transmitted to the second medium (transmitted wave). For lossy materials, in addition to these two parts, a third part of the wave is absorbed by the medium. A medium is considered lossless if $\sigma << \omega\varepsilon$. If $\sigma >> \omega\varepsilon$, the material is considered a good conductor otherwise the material is considered lossy.

The amount and the angular distribution of the scattered wave, as well as the amount absorbed, depends on the shape, size, and material of the scatterer (the illuminated body). In this study, the focus will be on plane harmonic incident waves. Note that any arbitrary field can be decomposed into its Fourier components which

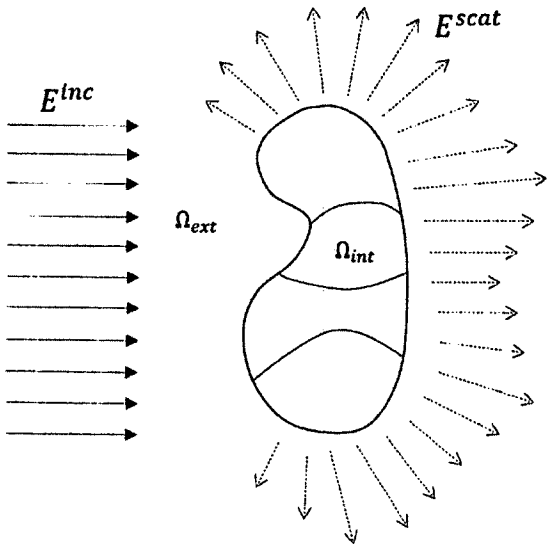


FIGURE. 3.1: Electro-magnetic scattering

are plane waves [9]. It is desired to determine the characteristics of the reflected and transmitted waves from those of the incident wave. For this purpose, the total field on the interface of two medium is forced to fulfill the boundary conditions. Figure 3.1 shows a scatterer which is illuminated by a incident wave. The incident EM wave $(\boldsymbol{E}^{inc}, \boldsymbol{H}^{inc})$ should satisfy the Maxwell equations in the entire domain Ω. The interior domain of the scatterer is separated as $\Omega_{int} = \cup_{i=1}^{n} \Omega_i$ where Ω_i denotes a single patch inside the scatterer. One can also define several sub-domains composed of one to several patches sharing identical material properties.

Since fairly complicated geometries can be represented as a single patch in IGA, each sub-domain can be modeled by a single patch in most of engineering applications. The exterior domain, which is outside of the scatterer, is denoted by Ω_{ext}. In this study, the material inside each patch is assumed to be homogeneous and isotropic. The total electric field is decomposed into the scattered wave $\boldsymbol{E}$ and

the incident wave $\boldsymbol{E}^{inc}$:

$$\boldsymbol{E}^{total} = \boldsymbol{E} + \boldsymbol{E}^{inc} \tag{3.4.1}$$

The total electric field $\boldsymbol{E}^{total}$ should satisfy Maxwell equations presented in Eq.3.2.3 subjected to the boundary conditions imposed on $\partial\Omega$. Moreover, the incident wave satisfies the reduced wave equation in the free space. That is:

$$\nabla \times \nabla \times \boldsymbol{E}^{inc} - \omega^2 \boldsymbol{E}^{inc} = 0, \tag{3.4.2}$$

where ω corresponds to the free space wave number. Assuming there is no current source in the scatterer, $\mathbf{J}^{imp} = 0$, one can obtain the following equation by substituting the Eq.3.4.2 into Eq.3.4.1:

$$\begin{aligned}\nabla \times \Big(\frac{1}{\mu_r}\nabla \times \boldsymbol{E}\Big) - \omega^2(\varepsilon_r - j\varepsilon_\sigma)\boldsymbol{E} &= -\nabla \times \Big(\frac{1}{\mu_r}\nabla \times \boldsymbol{E}^{inc}\Big) + \omega^2(\varepsilon_r - j\varepsilon_\sigma)\boldsymbol{E}^{inc} \\ &= -\frac{1}{\mu_r}\Bigg(\nabla \times \nabla \times \boldsymbol{E}^{inc} - \omega^2 \boldsymbol{E}^{inc} - \omega^2\Big(\big((\varepsilon_r - j\varepsilon_\sigma)\mu_r - 1\big)\boldsymbol{E}^{inc}\Big)\Bigg).\end{aligned} \tag{3.4.3}$$

The complex relative permittivity is defined as $\hat{\varepsilon_r} = \varepsilon_r - j\varepsilon_\sigma$. Since

$$\nabla \times \nabla \times \boldsymbol{E}^{inc} - \omega^2 \boldsymbol{E}^{inc} = 0,$$

Eq.3.4.3 can be rewritten as:

$$\begin{aligned}\nabla \times \Big(\frac{1}{\mu_r}\nabla \times \boldsymbol{E}\Big) - \omega^2\hat{\varepsilon}_r\boldsymbol{E} &= -\nabla \times \Big(\frac{1}{\mu_r}\nabla \times \boldsymbol{E}^{inc}\Big) + \omega^2\hat{\varepsilon}_r\boldsymbol{E}^{inc} \\ &= -\frac{1}{\mu_r}\Bigg(\nabla \times \nabla \times \boldsymbol{E}^{inc} - \omega^2\boldsymbol{E}^{inc} - \omega^2\Big(\big(\hat{\varepsilon}_r\mu_r - 1\big)\boldsymbol{E}^{inc}\Big)\Bigg) \\ &= \omega^2(\hat{\varepsilon_r} - \frac{1}{\mu_r})\boldsymbol{E}^{inc},\end{aligned} \tag{3.4.4}$$

The right hand side of Eq.3.4.3 can be used to define the equivalent volume current, $\mathbf{J}^{imp}$, in the scatterer:

$$-j\omega\boldsymbol{J}^{imp} = \omega^2(\hat{\varepsilon_r} - \mu^{-1})\boldsymbol{E}^{inc} \tag{3.4.5}$$

Equation 3.4.4 transforms the homogeneous wave equation of the total electric field to non-homogeneous wave equation for the scattered field. Note that the boundary conditions should be satisfied for the total electric field, $\boldsymbol{E}^{total}$: the Dirichlet boundary conditions (perfect electric conductor condition) require that the tangential component of the total field $\boldsymbol{E}$ vanish on the boundary. That is:

$$\boldsymbol{n} \times \left(\boldsymbol{E} + \boldsymbol{E}^{inc}\right) = 0 \quad \text{on} \quad \Gamma_D.$$

In other words, the initial homogeneous Dirichlet boundary condition for the total field is changed to an inhomogeneous one for the scattered field. Another requirement to obtain a unique solution is to satisfy the Silver-Muller radiation conditions at infinity [41]. However, the unbounded domain can not be discretized using finite element meshes. Also, increasing the size of the discretized domain will increase the computational cost and is not guaranteed to yield accurate results. To mimic the far field, the domain of interest must be truncated with an absorbing layer, called the Perfectly Matching Layer, which is discussed in the next section.

3.5 PERFECTLY MATCHED LAYER

In scattering problems, Maxwell equations should be solved in the unbounded domain while the scattered field is required to satisfy the Silver-Muller boundary condition at infinity:

$$\lim_{r\to\infty} r[\nabla \times \boldsymbol{E} - jk_0 \boldsymbol{e}_r \times (\boldsymbol{e}_r \times \boldsymbol{E})] = 0,$$

where r is representing the radius measured from the center of coordinates located

inside the scatterer and $\mathbf{e}_r$ is the unit vector along the radial direction under consideration. By satisfying this condition, the electric and magnetic waves will remain transverse in the far field region which also results in a radial Poynting vector ($\boldsymbol{E} \times \boldsymbol{H}$). Therefore, the resulting wave will propagate outward. Imposing the Silver-Muller radiation condition when solving electro-magnetic scattering problems will ensure the uniqueness of the solution.

Since the discretized domain in FEM/IGA is bounded, the infinite domain has to be truncated with an artificial boundary condition to mimic the far-field radiation condition. Such a boundary should be as transparent as possible to the scattered/radiated field to minimize non-physical reflections from this artificial boundary. Truncation of the wave propagation problems are not trivial. Due to the slow decay of these problems, truncation of the boundary with Neumann or Dirichlet boundary conditions will result in unacceptable artifacts from boundary reflections. Any coordinate remapping to represent the infinite space will result in infinitely fast oscillations as the wave get closer to the boundary. Such a behavior can not be captured using a finite number of grids in discretized space.

Therefore, to truncate the solution of the wave equation a *layer* should be defined to absorb the wave with no reflection from the boundary. Numerous methods have been invented to construct this artificial boundary: The *Infinite Elements* [15], *Non-Reflecting Boundary Conditions* [33], *Absorbing Boundary Conditions (ABCs)* [29], etc. The ABCs precede other methods, in which the absorbing layer extrapolates from the interior to the boundary of the layer mimicking infinite

extension. This method works perfectly for one-dimensional problems where the wave has only two possible direction of propagation. However, in more practical 2D and 3D problems the infinite number of possible propagation directions results in cumbersome application of ABCs. Another shortcoming is the limited number of angles at which existing ABCs can absorb the wave efficiently. Finally, most ABCs are formulated for homogeneous materials at the boundaries and may become unstable otherwise.

In 1994, Berenger introduced the *absorbing boundary layer* [7]. In this method, an artificiality absorbing material is placed adjacent to the edges of the scatterer. The wave entering this absorbing layer is attenuated by absorption and it decays exponentially. Berenger introduced the *Perfectly Matched Layer (PML)* to avoid reflections at the interface between the absorbing layer and the scatterer. Later a more general form of the PML was introduced through *complex coordinate stretching* [18, 64, 77]. The PML method which is commonly used in computational electromagnetism and is considered the most effective method available. The construction of PML in the FEM context [39, 41, 50] will be extended to IGA in this study.

Complex coordinate stretching: The region of interest, which may be composed of several patches, is centered at the origin of the coordinate system and is subjected to the wave equation. The space outside the region of interest should be truncated mimicking the infinite space as shown in Fig.3.2. The equations and the solution in the infinite space should be analytically continued to complex coordinates. To visualize the effects of complex stretching the far field space is assumed to be linear,

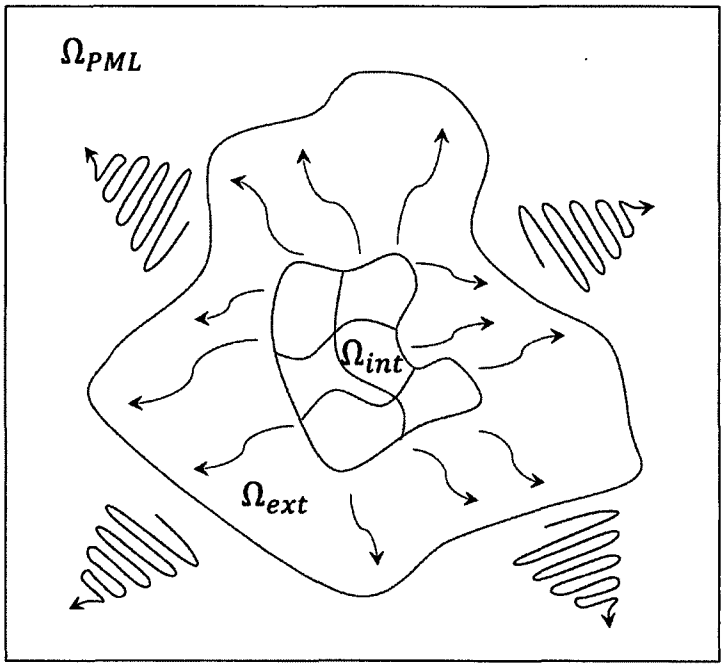


FIGURE. 3.2: PML Layer

homogeneous, and time-invariant, then, the radiating solution in the far field can be represented as a superposition of plane waves:

$$w(x,t) = \sum_{k,\omega} W_{k,\omega} e^{j(k\cdot x - \omega t)} \tag{3.5.1}$$

where $W_{k,\omega}$ are some constant amplitudes, ω is the angular frequency, and k is the wave vector. This analytical function can be evaluated at complex values of x. The original wave equation evaluated for real values of x has an oscillating term e^{jkx}. However, if a linearly growing imaginary part is added, the real part of the wave start decaying exponentially because:

$$e^{jk(\Re(x)+j\Im m(x))} = e^{jk\Re(x)} e^{-k\Im m(x)}.$$

In FEM, the evaluation in the complex plane is usually started at a certain distance (radius) from the origin. However, in IGA it is possible to start the PML layer

at the exact boundary of the region of interest exactly. Therefore, the PML in IGA can represent the geometry. The patches defining the region of interest should be terminated by PML patches in which the evaluation is performed in a complex plane. Note that at the interface of the PML with the region of interest, no complex streaching ia applied. Consequently, the PML is a reflection-less absorbing layer.

To overcome the inconvenience of solving the governing differential equation in complex plane, a coordinate transformation from complex to real coordinates is performed. The absorbing layer can be viewed as a layer in the real coordinates with an artificial anisotropic absorbing material. In other words, the Maxwell's equations described in complex coordinates are re-casted as the Maxwell's equations in Cartesian coordinates with the transformed materials [39]. The domain of this new coordinate system may be truncated with any type of boundary conditions such as homogeneous Dirichlet boundary conditions. Provided enough traveling distance, the PML layer can decay the propagating wave to machine zero resulting in no or exponentially small reflections from the boundary. Let the complex coordinate stretching be denoted with $\tilde{\boldsymbol{T}}$ as: $\tilde{\boldsymbol{T}} : \mathbb{R}^n \ni \boldsymbol{x} \rightarrow \tilde{\boldsymbol{x}} \in \mathbb{C}^n$ where $n = 1, 2, 3$ is the dimension of the space. Also, denote the corresponding Jacobian matrix by $\tilde{\boldsymbol{J}}$. where

$$\tilde{J}_{ij} = \frac{\partial \tilde{x}^i}{\partial x^j} \quad i, j, = 1, ..., n.$$

Since Maxwell's equations include divergence and curl of the unknown solution these operators should be described in the complex space. For $\tilde{u} = \tilde{u}(\tilde{\boldsymbol{x}})$ the divergence,

$\nabla_{\tilde{x}}\tilde{u}$ can be described in terms of real coordinates by:

$$\nabla_x u^* = \nabla_{\tilde{x}}\tilde{u} \circ \tilde{\boldsymbol{T}}$$

representing the modified solution in the real space with $u^* = \tilde{u} \circ \tilde{\boldsymbol{T}}$ and by using the original notation, $u := u^*$, one has:

$$\frac{\partial u}{\partial x^i}(\boldsymbol{x}) = \frac{\partial \tilde{u}}{\partial \tilde{x}^j}[\tilde{\boldsymbol{T}}(\boldsymbol{x})]\frac{\partial \tilde{x}^j}{\partial x^i}(\boldsymbol{x})$$

or

$$\nabla_{\boldsymbol{x}} u = \tilde{\boldsymbol{J}}^T(\nabla_{\tilde{\boldsymbol{x}}}\tilde{u} \circ \tilde{\boldsymbol{T}})$$

multiplying both sides by $\tilde{\boldsymbol{J}}^{-T}$ yields:

$$\nabla_{\tilde{\boldsymbol{x}}}\tilde{u} \circ \tilde{\boldsymbol{T}} = \tilde{\boldsymbol{J}}^{-T}\nabla_{\boldsymbol{x}} u$$

The same transformation can be used for $\boldsymbol{H}(curl)$-conforming fields [50]:

$$\tilde{\boldsymbol{E}} \circ \tilde{\boldsymbol{T}} = \tilde{\boldsymbol{J}}^{-T}\boldsymbol{E} \tag{3.5.2}$$

where $\boldsymbol{E}$ is the new unknown in the PML layer and $\tilde{\boldsymbol{E}}$ is the complex stretch of the original electrical field. Equation.3.5.2 is identical to Piola maps which transform matter elements onto physical ones. Following the same sequence one can obtain the required equation for the curl operator [50]:

$$(\nabla_{\tilde{\boldsymbol{x}}} \times \tilde{\boldsymbol{E}}) \circ \tilde{\boldsymbol{T}} = \tilde{J}^{-1}\tilde{\boldsymbol{J}}(\nabla_{\boldsymbol{x}} \times \boldsymbol{E}) \tag{3.5.3}$$

where $\tilde{J}$ is the determinant of the matrix $\tilde{\boldsymbol{J}}$. The PML layer should also satisfy the Maxwell equations. The bilinear form for the PML patches can be modified

accordingly:

$$\begin{aligned}\tilde{\boldsymbol{a}}(\tilde{\boldsymbol{E}},\tilde{\boldsymbol{F}}) &= \int_{\Omega}\big(\frac{1}{\mu_r}\nabla_{\tilde{\boldsymbol{x}}}\times\tilde{\boldsymbol{E}}\cdot\nabla_{\tilde{\boldsymbol{x}}}\times\tilde{\boldsymbol{F}}-\omega^2(\varepsilon_r-j\varepsilon_\sigma)\tilde{\boldsymbol{E}}\cdot\tilde{\boldsymbol{F}}\big)\tilde{J}d\Omega\\ &= \int_{\Omega}\big(\frac{1}{\mu_r}(\tilde{J}^{-2}\tilde{\boldsymbol{J}}^T\tilde{\boldsymbol{J}})\nabla_{\boldsymbol{x}}\times\boldsymbol{E}\cdot\nabla_{\boldsymbol{x}}\times\boldsymbol{F}-\omega^2(\varepsilon_r-j\varepsilon_\sigma)(\tilde{\boldsymbol{J}}^{-1}\tilde{\boldsymbol{J}}^{-T})\boldsymbol{E}\cdot\boldsymbol{F}\big)\tilde{J}d\Omega\\ &= \int_{\Omega}\big(\frac{1}{\mu_r}(\tilde{J}^{-1}\tilde{\boldsymbol{J}}^T\boldsymbol{J})\nabla_{\boldsymbol{x}}\times\boldsymbol{E}\cdot\nabla_{\boldsymbol{x}}\times\boldsymbol{F}-\omega^2(\varepsilon_r-j\varepsilon_\sigma)(\tilde{J}\tilde{\boldsymbol{J}}^{-1}\tilde{\boldsymbol{J}}^{-T})\boldsymbol{E}\cdot\boldsymbol{F}\big)d\Omega\end{aligned}$$

By defining the tensor $\boldsymbol{A}=\tilde{J}\tilde{\boldsymbol{J}}^{-1}\tilde{\boldsymbol{J}}^{-T}$, the above bilinear form can be rewritten as:

$$\tilde{\boldsymbol{a}}(\tilde{\boldsymbol{E}},\tilde{\boldsymbol{F}})=\int_{\Omega}\big(\frac{1}{\mu_r}\mathrm{A}^{-1}\nabla_{\boldsymbol{x}}\times\boldsymbol{E}\cdot\nabla_{\boldsymbol{x}}\times\boldsymbol{F}-\omega^2(\varepsilon_r-j\varepsilon_\sigma)\boldsymbol{A}\boldsymbol{E}\cdot\boldsymbol{F}\big)d\Omega \qquad (3.5.4)$$

The tensor $\boldsymbol{A}$ can be viewed as material anisotropic property. It can be shown that Maxwell equations under any coordinate transformation can be rewritten as Maxwell's equation in Cartesian coordinates with transformed materials [42,83].

3.6 SCATTERING OF INFINITE LONG CYLINDER

The scattering and the absorption of a domain with complex shape and composition are usually performed by experimental measurements. The electromagnetic scattering of the brain is usually studied by measuring the scattered field of a typical phantom of the head. However, these methods are very expensive. Also, experimental methods are not customizable to represent the geometrical detail of a specific case/patient. On the other hand, most of the *computational methods* such as FEM suffer from inaccurate geometrical representation/loss during discretization. In addition, creating a patient specific model using conventional FEM is very time-consuming. IGA can represent the geometry accurately. Also, it is possible generate a patient-specific model from medical images. Another advantage of IGA is the automatic meshing which can be used to generate patient specific models.

Theoretical scattering solutions are used by many researchers. The scattering of light by cells in tissue [34], the radiative properties of aerosols [46], the scattering of electromagnetic waves by sphere (Mie theory) and layered spheres [40, 53], the multiple scattering Mie solution [85] and the generalized Lornez-Mie solution [31] are good examples in which theoretical solutions were used. However, the exact solutions are limited to simple geometries such as spheres and cylinders.

The exact solutions of scattering and absorption by a sphere and an infinite right cylinder were discussed in [9]. Many researchers [9, 51, 65, 66] have provided computer codes to calculate the scattered field using Mie series. Schafer et al. [66] studied the near field magnetic and electric scattering of a monochromic plane wave. The derivation given in this section followed the construction of the exact solution discussed in the [9]. Let the scalar wave equation $\nabla^2\psi + k^2\psi = 0$ be considered in cylindrical polar coordinates (r, ϕ, z):

$$\frac{1}{r}\frac{\partial}{\partial r}\Big(r\frac{\partial \psi}{\partial r}\Big) + \frac{1}{r^2}\frac{\partial^2 \psi}{\partial \phi^2} + \frac{\partial^2 \psi}{\partial z^2} + k^2\psi = 0 \quad \text{in}\Omega \tag{3.6.1}$$

Separable solutions of the wave equation (Eq.3.6.1) can be found as:

$$\psi_n(r, \phi, z) = Z_n(\rho)e^{jn\phi}e^{jhz} \tag{3.6.2}$$

where $\rho = r\sqrt{k^2 - h^2}$ and Z_n is the solution of the Bessel equation. By considering $\hat{\boldsymbol{e}}_z$ along the cylinder axis, the cylindrical vector harmonics are:

$$\begin{aligned} \boldsymbol{M}_n &= \sqrt{k^2 - h^2}\Big(jn\frac{Z_n(\rho)}{\rho}\hat{\boldsymbol{e}}_r - Z_n'(\rho)\hat{\boldsymbol{e}}_\phi\Big)e^{j(n\phi+hz)}, \\ \boldsymbol{N}_n &= \frac{\sqrt{k^2 - h^2}}{k}\Big(jhZ_n'(\rho)\hat{\boldsymbol{e}}_r - hn\frac{Z_n(\rho)}{\rho}\hat{\boldsymbol{e}}_\phi + \sqrt{k^2 - h^2}Z_n(\rho)\hat{\boldsymbol{e}}_z\Big)e^{j(n\phi+hz)}. \end{aligned} \tag{3.6.3}$$

Consider a plane wave illuminating the cylinder shown in Fig.3.3. Note the ξ is defined as the angle between the axis of the cylinder and the incident field. The

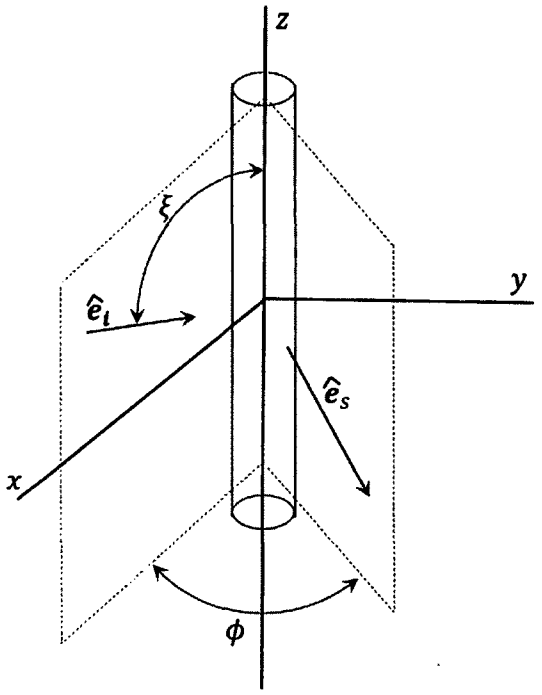


FIGURE. 3.3: Oblique illumination of an infinite cylinder by plane wave

incident field is polarized either parallel (plane of incidence) or perpendicular to the $x-z$ plane. The total field is a linear combination of the these components which should satisfy the following boundary conditions at the surface of the cylinder:

$$[\boldsymbol{E}_2(\boldsymbol{x}) - \boldsymbol{E}_1(\boldsymbol{x})] \times \hat{\boldsymbol{n}} = 0,$$

$$[\boldsymbol{H}_2(\boldsymbol{x}) - \boldsymbol{H}_1(\boldsymbol{x})] \times \hat{\boldsymbol{n}} = 0,$$

For a infinite cylinder illuminated by homogeneous plane wave $\boldsymbol{E}_i = \boldsymbol{E}_0 e^{jk\hat{\mathbf{e}}_i \cdot \boldsymbol{x}}$ which is polarized parallel to $x-z$ plane, the scattered field is described as:

$$\begin{aligned}
\boldsymbol{E}_{s\|} &= -\sum_{n=\infty}^{\infty} E_n\big(b_{n\|}\boldsymbol{N}_n + ja_{n\|}\boldsymbol{M}_n\big), \\
\boldsymbol{H}_{s\|} &= \frac{jk}{\omega\mu}\sum_{n=-\infty}^{\infty} E_n\big(b_{n\|}\boldsymbol{M}_n + ja_{n\|}\boldsymbol{N}_n\big), \\
\boldsymbol{E}_n &= \boldsymbol{E}_0\frac{(-1)^n}{k\sin\xi},
\end{aligned} \tag{3.6.4}$$

Similarly, for the polarization vector perpendicular to the x-z plane, the scatter field is:

$$\begin{aligned} \boldsymbol{E}_{s\perp} &= \sum_{n=-\infty}^{\infty} E_n\big(ja_{n\perp}\boldsymbol{M}_n + b_{n\perp}\boldsymbol{N}_n\big), \\ \boldsymbol{H}_{s\perp} &= -\frac{jk}{\omega\mu}\sum_{n=-\infty}^{\infty} E_n\big(ja_{n\perp}\boldsymbol{N}_n + b_{n\perp}\boldsymbol{M}_n\big), \end{aligned} \tag{3.6.5}$$

where

$$\begin{aligned} a_{n\parallel} &= \frac{C_nV_n - B_nD_n}{W_nV_n + jD_n^2}, \qquad b_{n\parallel} = \frac{W_nB_n + jD_nC_n}{W_nV_n + jD_n^2} \\ a_{n\perp} &= -\frac{A_nV_n - jC_nD_n}{W_nV_n + jD_n^2}, \quad b_{n\perp} = -j\frac{C_nW_n + A_nD_n}{W_nV_n + jD_n^2} \end{aligned} \tag{3.6.6}$$

and A_n, B_n, C_n, D_n, V_n, and W_n are defined in terms of $\zeta = x\sin\xi$, and $\eta = x\sqrt{m^2 - \cos^2\xi}$:

$$\begin{aligned} A_n &= j\zeta[\zeta J_n'(\eta)J_n(\zeta) - \eta J_n(\eta)J_n'(\zeta)], \\ B_n &= \zeta[m^2\zeta J_n'(\eta)J_n(\zeta) - \eta J_n(\eta)J_n'(\zeta)], \\ C_n &= n\cos(\xi)\eta J_n(\eta)J_n(\zeta)\Big(\frac{\zeta^2}{\eta^2} - 1\Big), \\ D_n &= n\cos(\xi)\eta J_n(\eta)H_n^{(1)}(\zeta)\Big(\frac{\zeta^2}{\eta^2} - 1\Big), \\ V_n &= \zeta[m^2\zeta J_n'(\eta)H_n^{(1)}(\zeta) - \eta J_n(\eta)H_n^{(1)'}(\zeta)], \\ W_n &= j\zeta[\eta J_n(\eta)H_n^{(1)'}(\zeta) - \zeta J_n'(\eta)H_n^{(1)}(\zeta)]. \end{aligned} \tag{3.6.7}$$

The $H_n^{(1)}$ is the Hankel function: $H_n^{(1)} = J_n + jY_n$ where J_n and Y_n are the Bessel functions of first and second kind of integral order n. The relation between the incident and the scattered field can be written in matrix form:

$$\begin{pmatrix} E_{\parallel s} \\ E_{\perp s} \end{pmatrix} = e^{j\frac{3\pi}{4}}\sqrt{\frac{2}{\pi kr\sin\xi}}e^{ik(r\sin\xi - z\cos\xi)}\begin{pmatrix} T_1 & T_4 \\ T_3 & T_2 \end{pmatrix}\begin{pmatrix} E_{\parallel i} \\ E_{\perp i} \end{pmatrix}. \tag{3.6.8}$$

Defining the change of variable: $\Theta = \pi - \phi$, the elements $T_1, T_2, T_3, andT_4$ of the

amplitude scattering matrix,$(T_1 - T_4)$ are given as:

$$T_1 = \sum_{-\infty}^{\infty} b_{n\parallel} e^{-jn\theta} = b_{0\parallel} + 2\sum_{n=1}^{\infty} b_{n\parallel} \cos(n\Theta), \tag{3.6.9a}$$

$$T_2 = \sum_{-\infty}^{\infty} a_{n\perp} e^{-jn\Theta} = a_{0\perp} + 2\sum_{n=1}^{\infty} a_{n\perp} \cos(n\Theta), \tag{3.6.9b}$$

$$T_3 = \sum_{-\infty}^{\infty} a_{n\parallel} e^{-jn\Theta} = -2j\sum_{n=1}^{\infty} a_{n\parallel} \sin(n\Theta), \tag{3.6.9c}$$

$$T_4 = \sum_{-\infty}^{\infty} b_{n\perp} e^{-jn\Theta} = -2j\sum_{n=1}^{\infty} b_{n\perp} \sin(n\theta) = -T_3. \tag{3.6.9d}$$

The efficiencies for scattering, C_{scat}, and extinction C_{ext} are:

$$C_{scat\parallel} = \frac{4}{k}\Big[|b_{0\parallel}| + 2\sum_{n=1}^{\infty}\left(|b_{n\parallel}|^2 + |a_{n\parallel}|^2\right)\Big], \tag{3.6.10a}$$

$$C_{scat,\perp} = \frac{4}{k}\Big[|a_{0\perp}|^2 + 2\sum_{n=1}^{\infty}\left(|a_{n\perp}|^2 + |b_{m\perp}|^2\right)\Big]. \tag{3.6.10b}$$

Particularly, when the incident field is perpendicular to the cylinder axis, $\xi = \frac{\pi}{2}$, the two different polarization directions decouple and therefore can be considered separately:

$$\begin{pmatrix} E_{\parallel s} \\ E_{\perp s} \end{pmatrix} = e^{j\frac{3\pi}{4}}\sqrt{\frac{2}{\pi kr\sin\xi}}e^{ik(r-x)}\begin{pmatrix} T_1 & 0 \\ 0 & T_2 \end{pmatrix}\begin{pmatrix} E_{\parallel i} \\ E_{\perp i} \end{pmatrix}, \tag{3.6.11}$$

also, for this case ($\xi = \frac{\pi}{2}$) one has:

$$a_{n\parallel} = 0, \tag{3.6.12a}$$

$$b_{n\perp} = 0, \tag{3.6.12b}$$

$$a_n = a_{n\perp} = \frac{mJ_n(mx)J_n'(x) - J_n(x)J_n'(mx)}{mJ_n(mx)H_n^{(1)'}(x) - H_n^{(1)}(x)J_n'(mx)}, \tag{3.6.12c}$$

$$b_n = b_{n\parallel} = \frac{J_n(mx)J_n'(x) - mJ_n(x)J_n'(mx)}{J_n(mx)H_n^{(1)'}(x) - mH_n^{(1)}(x)J_n'(mx)}. \tag{3.6.12d}$$

In this study, the analytical solution will be used to evaluate the numerical results obtained by IGA analysis. Specifically, the near field solution of the scattered and the total electrical field is of particular interest.

CHAPTER 4

GENERATING ANALYSIS READY GEOMETRY

In Chapter 2, the B-Spline curves and surfaces were introduced. Recall that a B-Spline surface has a tensor product structure and carries the parametric discretization which can be used in the IGA context. The CAD industry has spent a great deal of effort to develop efficient algorithms to represent complex geometries. Specifically, numerous algorithms are available to *interpolate* or *estimate* a set of geometrical data points [61]. The interpolated curve or surface crosses all the given geometrical data precisely. This is not necessarily true in B-Spline approximation; approximation will only capture the shape of the curve/surface without necessarily crossing every data point. This property might be attractive when data noise is expected.

For most engineering applications, the analysis model is generated by a very coarse mesh and then refined to yield a desired solution accuracy. In this study, the curve/surface interpolation is used to generate the analysis domain. The interpolating B-Spline is defined by a knot vector in each spatial direction and a set of control points. Therefore, the resulting B-Spline already contains the discretization information required to perform analysis in IGA. This property is very attractive in generating models based on geometrical data obtained from coordinate measurement or imaging devices. Specifically, for a human brain section, it is possible to extract the geometrical data from a MRI image and then represent each layer of the brain

with one B-Spline curve which can be used to generate a surface covering each layer of the brain.

There are two main types of interpolations namely: *Global* and *Local* interpolations. Local interpolations are generated segment-wise using the local data in each step. Consequently, perturbing a single data point will only change the resulting B-Spline locally. In the global interpolation, perturbing a single data point would change the entire curve/surface. However, this change is more pronounced near the perturbation and will die out with an increasing distance. Local interpolations are less computationally expensive and can represent straight lines and also capture local deformations better than global interpolations. However, local methods tend to generate multiple interior knots. Consequently, achieving a desired continuity at the boundary of segments is far from trivial for local interpolations.

In this study the analysis model for IGA is generated by global interpolation. As a result, the number of meshes in the interpolated/estimated curve or surface will be a function of the number of data points. One can perform h-, p-, or k- refinements to generate the required number of the meshes and/or increase the order of underlying B-Splines in the model. It is possible to generate two dimensional discretization by constructing a surface between B-Spline curves obtained from curve interpolation. In this chapter the basics of curve and surface interpolation, the ruled surfaces, and the Coons patches, are described. Then, the brain section was discretized by defining multiple-patches. Finally, the requirements for multi-patch conforming analysis are discussed and applied on the brain section model.

4.1 B-SPLINE GLOBAL CURVE INTERPOLATION

To interpolate a set of points, $\{Q_k\}, k = 0, ..., n$, using a p^{th} degree non-rational B-Spline curve, a parameter value, $\bar{u}_k$, is assigned to each data point, Q_k. By selecting an appropriate knot vector $\boldsymbol{U} = \{u_0, ..., u_{n+p+1}\}$, a system of $(n+1)\times(n+1)$ linear equations can be set up to construct a B-spline curve as defined by Eq.2.3.24:

$$Q_k = C(\bar{u}_k) = \sum_{i=0}^{n} N_{i,p}(\bar{u}_k)\boldsymbol{P}_i. \tag{4.1.1}$$

The $n + 1$ unknowns of this system are the location of the control points in each spatial direction, $\boldsymbol{P}_i$. This system of equations were solved in each direction to obtain the coordination of the control points. Several methods exist to select $\bar{u}_k$, namely *equally spaced knots*, the *cord-length method*, and the *centripetal method*. A curve with equally spaced knots cannot to yield acceptable results if the data is unevenly spaced which is the case for the brain section data. Therefore, the focus will be on the cord-length and the centripetal methods [61]. The cord-length method is most commonly used method which generally yields adequate results.

The cord length method: let d be the total cord length:

$$d = \sum_{k=1}^{n} \left|\boldsymbol{Q}_k - \boldsymbol{Q}_{k-1}\right|, \tag{4.1.2}$$

where $\bar{u}_0 = 0$ and $\bar{u}_n = 1$, and

$$\bar{u}_k = \bar{u}_{k-1} + \frac{\left|\boldsymbol{Q}_k - \boldsymbol{Q}_{k-1}\right|}{d}, \quad k = 1, ..., n-1. \tag{4.1.3}$$

The centripetal method [44] is used when data has sharp edges and turns:

$$d = \sum_{k=1}^{n} \sqrt{|\mathbf{Q}_k - \mathbf{Q}_{k-1}|}, \tag{4.1.4}$$

where $\bar{u}_0 = 0$ and $\bar{u}_n = 1$, and

$$\bar{u}_k = \bar{u}_{k-1} + \frac{\sqrt{|\boldsymbol{Q}_k - \boldsymbol{Q}_{k-1}|}}{d}, \qquad k = 1, ..., n-1. \tag{4.1.5}$$

To select the $n+p+1$ elements of knot vector, $\boldsymbol{U}$ for a p^{th} order B-Spline curve, the averaging technique is used:

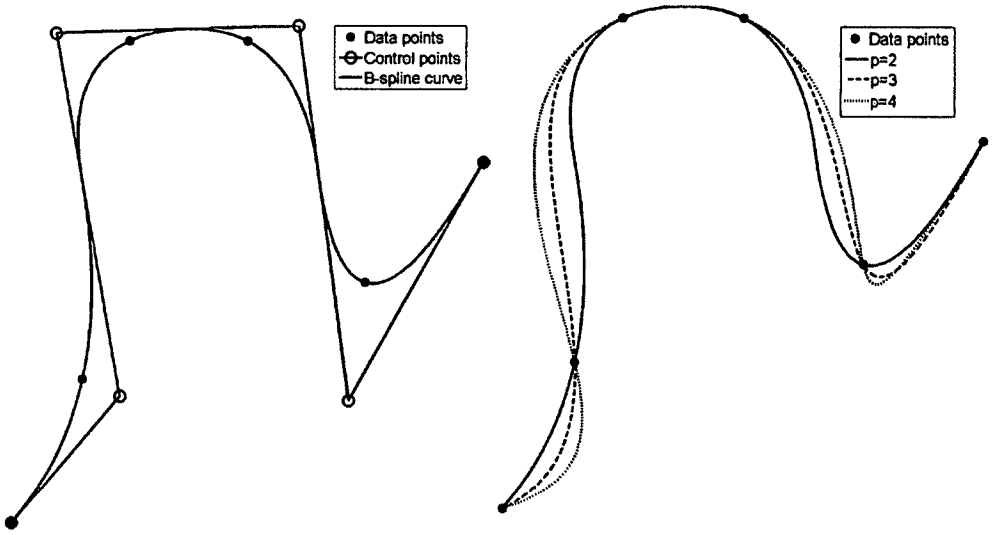
$$\begin{aligned} u_0 = ... = u_p = 0, \\ u_{m-p} = ...u_m = 1, \\ m = n + p + 1. \end{aligned} \tag{4.1.6}$$

The required knots are obtained using the following equation:

$$u_{j+p} = \frac{1}{p} \sum_{i=j}^{j+p-1} \bar{u}_i, \qquad j = 1, ..., n-p. \tag{4.1.7}$$

Using this method, assigned knots reflect the distribution of $\bar{u}_k$. Also, the resulting linear system of equations for control points, Eq.4.1.1, is totally positive and banded [13, 61]. A Matlab code (the Matworks, Natick, MA, USA) has been prepared to interpolate the geometrical data points with a p^{th} order B-Spline curve following the global interpolation algorithm and the cord length method. Providing the geometrical data and the desired regularity of the interpolating B-Spline, all the required control points and knot vectors are calculated automatically.

The resulting interpolating B-Spline will exactly cross all the given data points. For the purpose of illustration, a set of given geometrical data points, the calculated control points, and the interpolating B-Spline are shown in Figure 4.1a. The effect of the interpolating B-Spline order, p, on the resulted curve is shown in Fig.4.1b.



(a) Quadratic curve interpolation

(b) Effects of B-Spline order on curve interpolation

FIGURE. 4.1: Global B-Spline interpolation of given data points

To generate the brain section model, the geometrical data points of the boundary of each material inside the brain was extracted from a MRI image. Then, each layer was interpolated by a single B-Spline. Figure 4.2 depicts all the resulting B-Splines representing the material boundaries in a human brain section. Note that the data points shown in Fig.4.1a are sparse enough to show the effect of B-Spline order on the resulted curve. For the brain section however, the high density of the extracted data points will result in minimal effect of the selected B-Spline order on the resulted shape of the boundaries. Nevertheless, the selected order will directly affect the regularity of the interpolated curve.

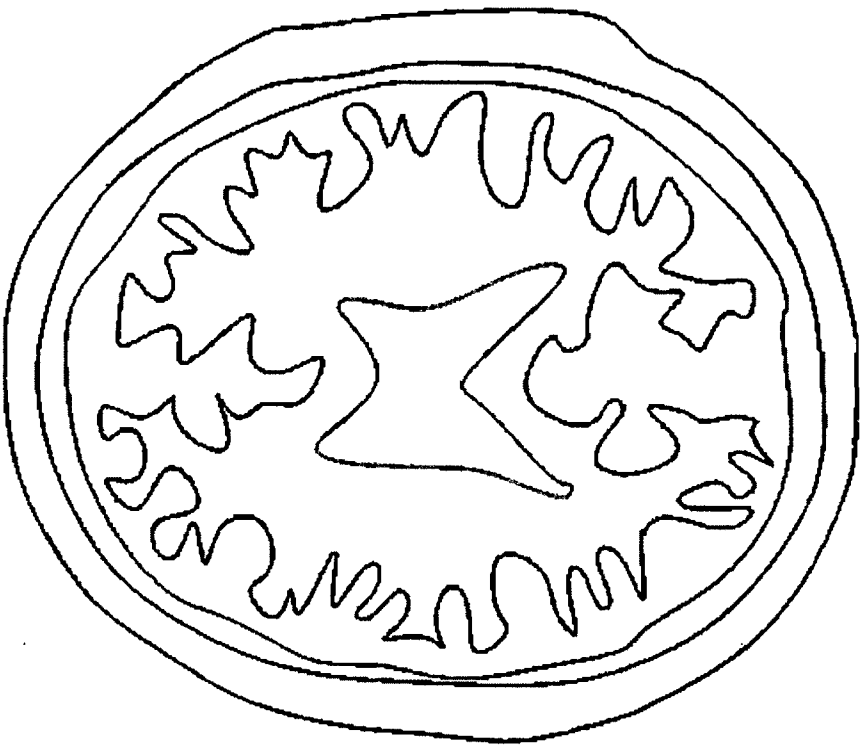


FIGURE. 4.2: Material boundary B-Spline interpolation

4.2 B-SPLINE GLOBAL SURFACE INTERPOLATION

In this section, the basics of surface interpolation to generate analysis domain are discussed. Given a given set of $(n+1) \times (m+1)$ data points, $\{\mathbf{Q}_{k,l}\}, k = 0, ..., n,$ and $l = 0, ..., m$, a non-rational (p, q)th degree surface interpolating the given data points is generated similar to Eq.2.3.25:

$$\mathbf{Q}_{k,l} = \mathbf{S}(\bar{u}_k, \bar{v}_l) = \sum_{i=0}^{n} \sum_{j=0}^{m} N_{i,p}(\bar{u}_k) N_{j,q}(\bar{v}_l) \mathbf{P}_{i,j}, \tag{4.2.1}$$

to calculate $\bar{u}_k$ and $\bar{v}_k$ the following equations based upon the averaging method can

be used:

$$\bar{u}_k = \frac{1}{m+1}\sum_{l=0}^{m} \bar{u}_k^l, \quad k = 0, ..., n, \tag{4.2.2a}$$

$$\bar{v}_l = \frac{1}{n+1}\sum_{k=0}^{n} \bar{v}_l^k, \quad l = 0, ..., m. \tag{4.2.2b}$$

Similar to curve interpolation, $v_0 = u_0 = ... = v_p = u_p = 0$ and $v_{m-p} = u_{m-p} = ...v_m = u_m = 0$. For each fixed l and k the $\bar{u}_k^l$, and $\bar{v}_l^k$ are calculated either by the cord-length, or the centripetal methods (Eq.4.1.3 and Eq.4.1.5 respectively) for curve interpolation. Then, the averaging technique is used to calculate $\bar{u}_k$ and $\bar{v}_k$ as described by Eq.4.2.2. Therefore the required knot vectors $\boldsymbol{U} = \{\bar{u}_0, \cdots, \bar{u}_{n+p+1}$ and $\boldsymbol{V} = \{\bar{v}_0, \cdots, \bar{v}_{m+q+1}\}$ can be constructed. The control points can be found by solving Eq.4.2.1 which can be re-written as:

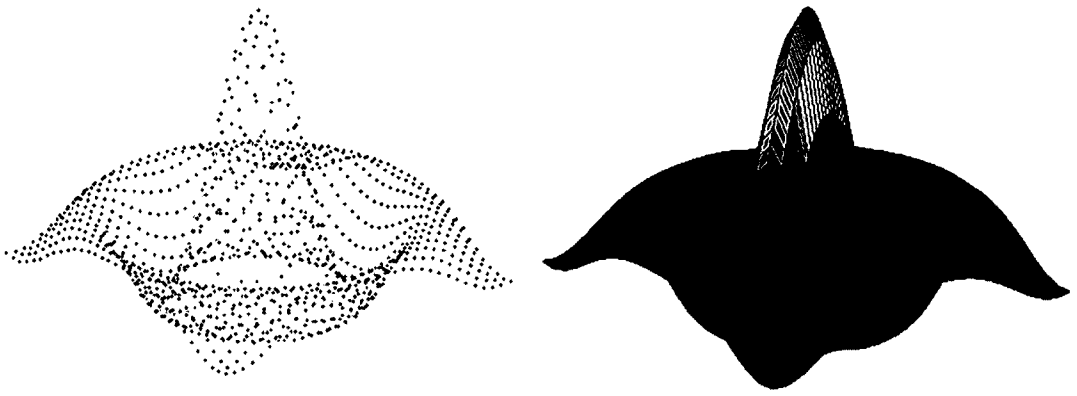
for each l

$$\mathbf{Q}_{k,l} = \sum_{i=0}^{n} N_{i,p}(\bar{u}_k)\left(\sum_{j=0}^{m} N_{j,q}(\bar{v}_l)\mathbf{P}_{i,j}\right) = \sum_{i=0}^{n} N_{i,p}(\bar{u}_k)\mathbf{R}_{i,l}, \quad k = 0, ...n, \tag{4.2.3}$$

where for each

$$\mathbf{R}_{i,l} = \sum_{j=0}^{m} N_{j,q}(\bar{v}_l)\mathbf{P}_{i,j}, \quad l = 0, ..., m. \tag{4.2.4}$$

Equation 4.2.3 is set of curve interpolations of the points $Q_{k,l}$ $k = 0, ..., n$, which is solved to find $\mathbf{R}_{i,l}$. Similarly, Eq.4.2.4 is another curve interpolation that should be solved to find the required control points $\mathbf{P}_{i,j}$. The surface interpolation can generate 3D surfaces crossing all the given geometrical data points. The resulted B-Spline surface has built upon the given discretization data and can be used for IGA analysis. Figure 4.3a, and Fig.4.3b are showing the given example data points, and the interpolated B-Spline surface respectively. The number control points in a



(a) Given data points (b) B-Spline surface interpolation

FIGURE. 4.3: Global surface interpolation of given data points

interpolated surface is a function of the number of data points in each parametric direction. The use of surface interpolation to generate a surface between brain material boundaries will result in limited number of the internal control points. The h- and/or p-refinement(s) can be used to increase the number of meshes and the control points to increase the accuracy of the solution.

4.3 THE RULED SURFACE

A ruled surface can be viewed as a linear interpolation between a pair of curves. Consider two B-Spline curves described as:

$$\boldsymbol{C}_k(u) = \sum_{i=1}^{n_k} \boldsymbol{N}_{i,p_k}(u)\boldsymbol{P}_i^k, \quad k = 1, 2, \tag{4.3.1}$$

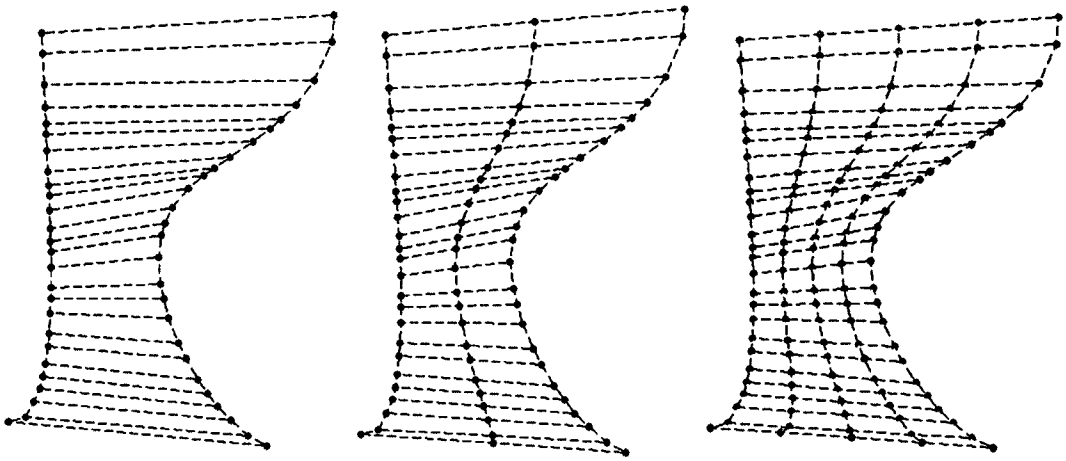
which are defined on knot vectors $\boldsymbol{U}^k = \{u_0^k, ..., u_{m_k}^k\}$. A linear interpolation between $\boldsymbol{C}_1(u)$ and $\boldsymbol{C}_2(u)$ is desired. This interpolation should be between points with equal parameter values so that for a fixed $\bar{u}$ a line is connecting $\boldsymbol{C}_1(\bar{u})$ and $\boldsymbol{C}_2(\bar{u})$. Such a

surface can formulated as:

$$\boldsymbol{S}(\boldsymbol{u},\boldsymbol{v}) = \sum_{i=0}^{n}\sum_{j=0}^{1} \boldsymbol{N}_{i,p;j,1}(\boldsymbol{u},\boldsymbol{v})\boldsymbol{P}_{i,j}, \tag{4.3.2}$$

where $\boldsymbol{V} = \{0,0,1,1\}$ is the knot vector in the linear direction, also, $n, \boldsymbol{U}, p$, and $\boldsymbol{P}_{i,j}$ should be specified. Because of tensor product nature of the B-Spline surfaces, both boundary curves should be defined by the same knot vector and B-Spline degree. Therefore, knot insertion (h-refinement) and degree elevation (p-refinement) might be necessary to create a ruled surface.

The knot vectors in $\boldsymbol{C}_1$ and $\boldsymbol{C}_2$ are obtained using the averaging method in the curve interpolation are not necessarily identical. To create an identical knot vector for both boundary curves, the knot vectors of both curves are merged to obtain a common knot vector $\boldsymbol{U}^c$. In other words, u_j is in $\boldsymbol{U}^c$ if it is in either $\boldsymbol{U}^1$ or $\boldsymbol{U}^2$. The maximum multiplicity of u_j in $\boldsymbol{U}^1$ or $\boldsymbol{U}^2$ is also carried over to $\boldsymbol{U}^c$. Then, $\boldsymbol{U}^c$ which is used to refine both curves which yields the final values $\boldsymbol{P}_{ij}$. Consequently, this process will increase the number of elements in analysis similar to h-refinement.



(a) Iinitial ruled surface (b) Degree evaluation, p = 2 (c) Degree evaluation, p = 3

FIGURE. 4.4: Degree elevation for ruled surfaces

The linear property of the ruled surfaces will limit the flexibility of this method to represent complex geometries. The ruled surface generated between two interpolated curves is shown in Fig.4.4a, in which the control points are shown with red dots. As discussed before, and clearly seen in Fig.4.4a a ruled surface is a linear interpolation between two B-Spline curves. By increasing the order of the resulted surface in one parametric direction, one can insert an additional set of internal control points inside the surface. The effect of degree elevation on the ruled surface is illustrated in Figs.4.4b and 4.4c.

The capacity of ruled surfaces to mesh the skull section as one patch is illustrated in Figure 4.5. Since the parametrization of the curves defining the boundaries of skull is a function of the geometrical location of data points, the resulting discretization of the boundaries are not identical. Consequently, the resulting one-patch surface will show distorted meshing. Multi-patch discretization is preferable

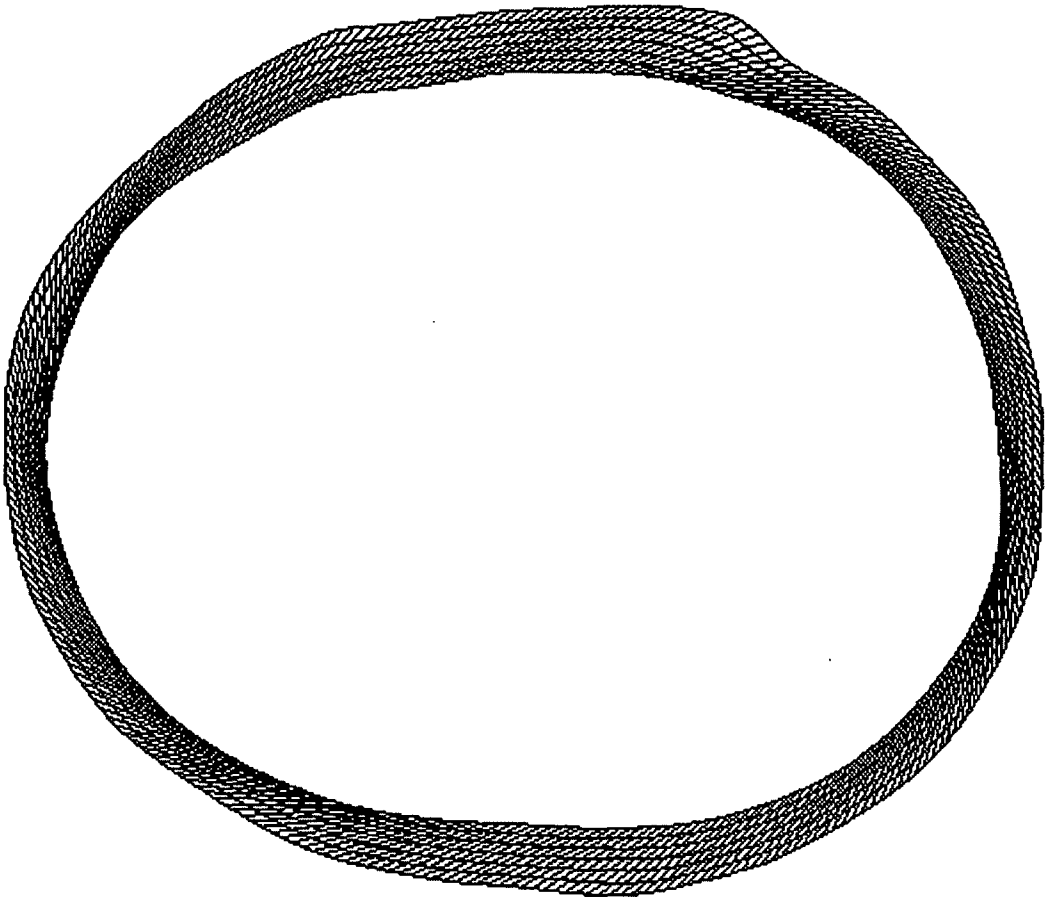


FIGURE. 4.5: A ruled surface representing the skull section as one patch

for complex geometries. The multi-patch discretization of the brain section will be discussed in the next section using the Coons patch method.

4.4 THE COONS PATCH

Coons patches are widely used in CAD industry [30] as well as in model generation for isogeometric analysis [62]. Each Coons patch will represent a patch in IGA which can be assigned a unique set of material properties. In multi-patch IGA, when two patches share a common interface, the continuity along the common

boundary is reduced to C^0 . This property is desirable in electromagnetism because of the predicted discontinuity of the field in the interface of different materials. In very complex geometries, such as the brain section, it might be necessary to consider multiple patches to represent a single material. Multi-patch analysis is unavoidable for practical applications. First developed in mid 1960s [19], a Coons surface is generated by the surrounding curves that satisfy the compatibility requirements. Given four curves:

$$\begin{aligned} \mathbf{C}_k(u) &= \sum_{i=0}^{n} N_{i,p}(u)\mathbf{P}_{k,i}, \qquad k = 0,1, \quad u \in [0,1], \\ \mathbf{C}_l(v) &= \sum_{j=0}^{m} N_{j,q}(v)\mathbf{P}_{l,j}, \qquad l = 0,1, \quad v \in [0,1]. \end{aligned} \tag{4.4.1}$$

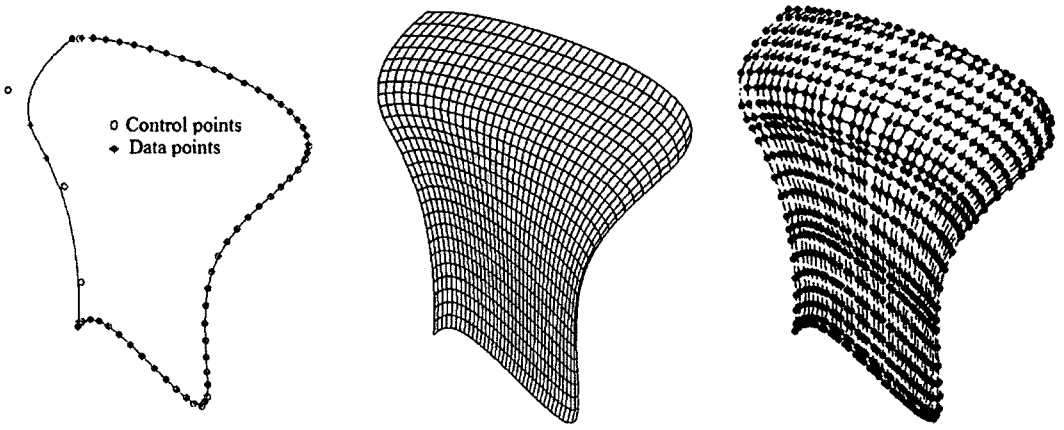
The comparability conditions for a Coons surface are:

- The opposite boundary B-Spline curves should be constructed on common knot vectors. In other words, both $C_K(u)$ are constructed over on a common knot vector, $\mathbf{U}$, and similarly, $C_l(v)$s also share a common knot vector $\mathbf{V}$;

- The adjacent curves are connected at the common ends;

$$\begin{aligned} \mathbf{S}_{0,0} &= \mathbf{C}_{k=0}(u=0) = \mathbf{C}_{l=0}(v=0) \\ \mathbf{S}_{1,0} &= \mathbf{C}_{k=0}(u=1) = \mathbf{C}_{l=1}(v=0) \\ \mathbf{S}_{0,1} &= \mathbf{C}_{k=1}(u=0) = \mathbf{C}_{l=0}(v=1) \\ \mathbf{S}_{1,1} &= \mathbf{C}_{k=1}(u=1) = \mathbf{C}_{l=1}(v=1) \end{aligned} \tag{4.4.2}$$

Denoting the ruled surface between $\mathbf{C}_{k=0}(u)$ and $\mathbf{C}_{k=1}(u)$ with $\mathbf{R}_1$ and the ruled surface between $\mathbf{C}_{l=0}(v)$ and $\mathbf{C}_{l=1}(v)$ with $\mathbf{R}_2$, the bilinear Coons surface can be



(a) B-Spline interpolation (b) Coons patch (c) Control Points

FIGURE. 4.6: Discretization using the Coons patch

defined as a bilinear blend of surfaces $\mathbf{R}_1$ and $\mathbf{R}_2$ as:

$$\mathbf{S}(u,v) = \mathbf{R}_1(u,v) + \mathbf{R}_2(u,v) - \mathbf{T}(u,v), \tag{4.4.3}$$

where $\mathbf{T}(u,v)$ is the bilinear tensor product surface defined as:

$$\mathbf{T}(u,v) = \begin{bmatrix} 1 & u \end{bmatrix} \begin{bmatrix} \mathbf{S}_{0,0} & \mathbf{S}_{0,1} \\ \mathbf{S}_{1,0} & \mathbf{S}_{1,1} \end{bmatrix} \begin{bmatrix} 1 \\ v \end{bmatrix}. \tag{4.4.4}$$

Provided that the B-Splines are compatible, the surface $\mathbf{S}(u,v)$ can be constructed by adding or subtracting the corresponding control points. That is:

$$\mathbf{P}_{i,j} = \mathbf{P}_{i,j}^{R_1} + \mathbf{P}_{i,j}^{R_2} - \mathbf{P}_{i,j}^{T}. \tag{4.4.5}$$

The use of the Coon patch in generating the analysis domain is shown in Figure 4.6a where the boundary of the region of interest is divided into four sections and interpolated by four B-Splines. The B-Spline interpolation does not necessarily generate compatible curves. h-refinement (knot insertion) can be used to modify the

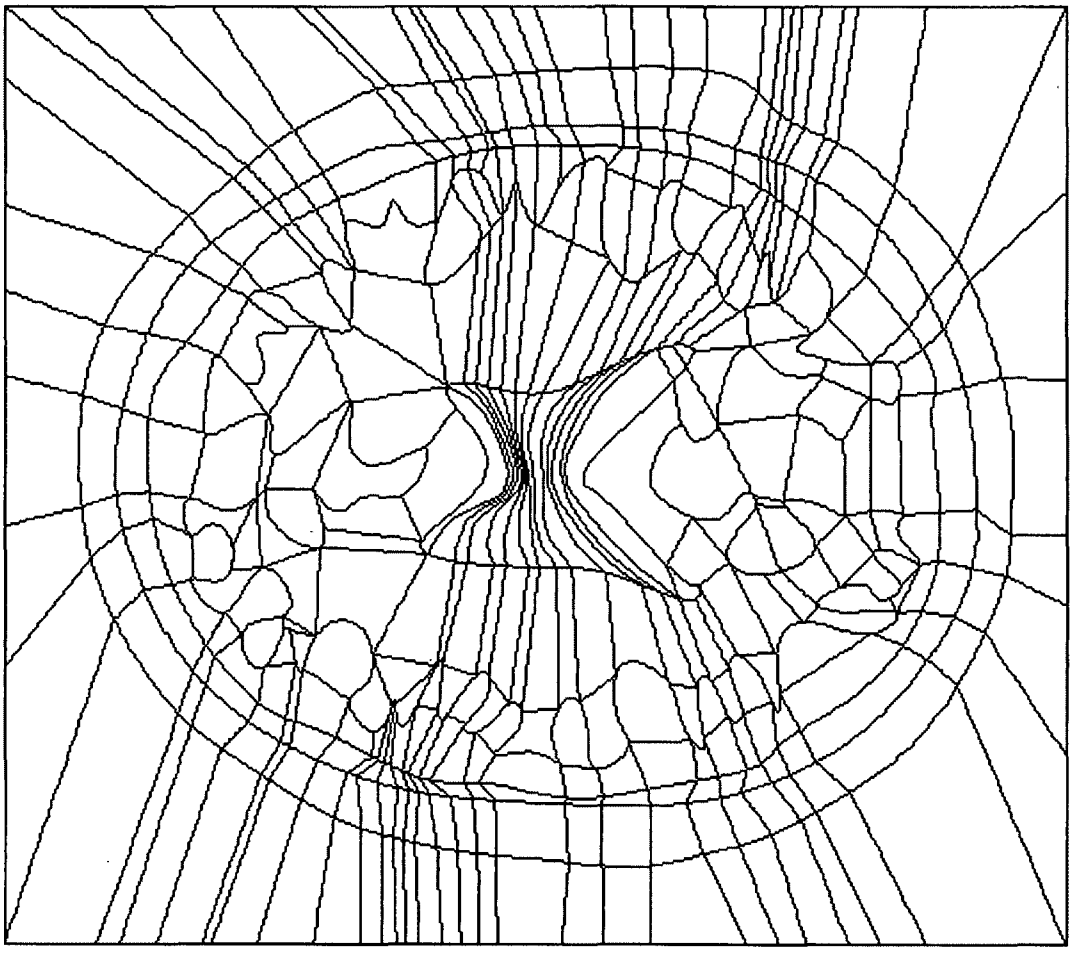


FIGURE. 4.7: The required patches for the brain section and the surrounding environment

knot vector of the curves without affecting the geometry of the curve. After knot insertion, a Coons patch is generated between these four curves as shown in Fig.4.6b. The control points of this Coons patch is shown in Fig.4.6c. Note that the internal control points are generated automatically without any additional refinement. To mesh the brain section and its surrounding environment, 399 Coons patches and 770 interfaces are defined as shown in Fig.4.7.

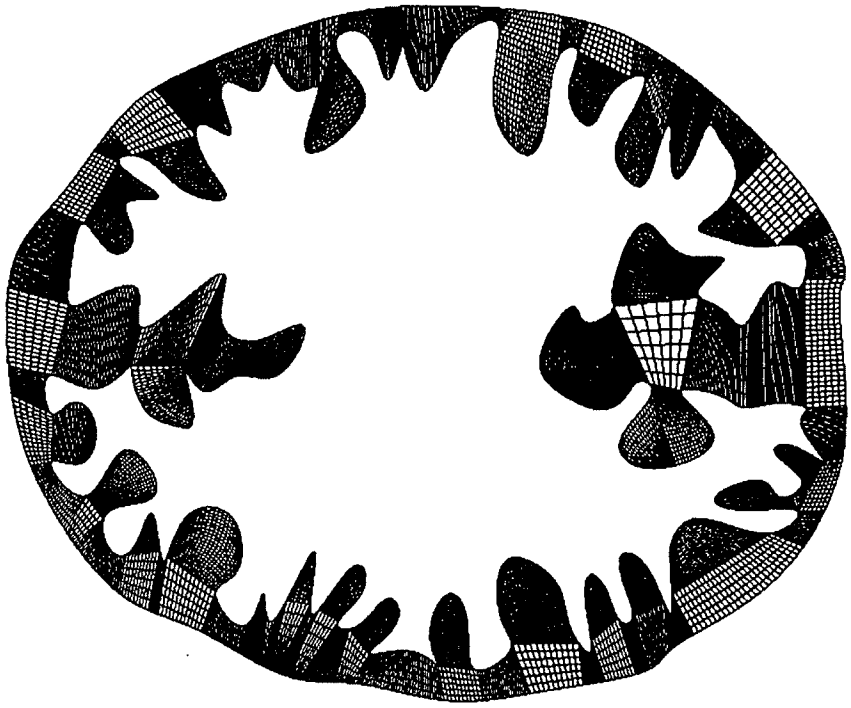


FIGURE. 4.8: Meshing the gray matter of the brain section

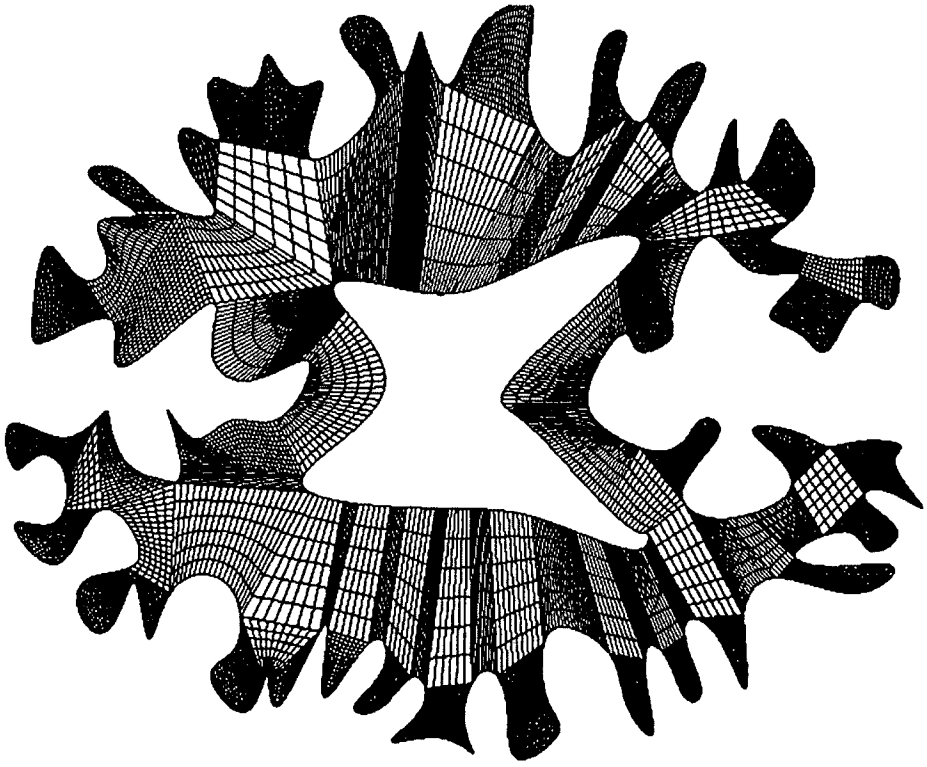


FIGURE. 4.9: Meshing the white matter of the brain section

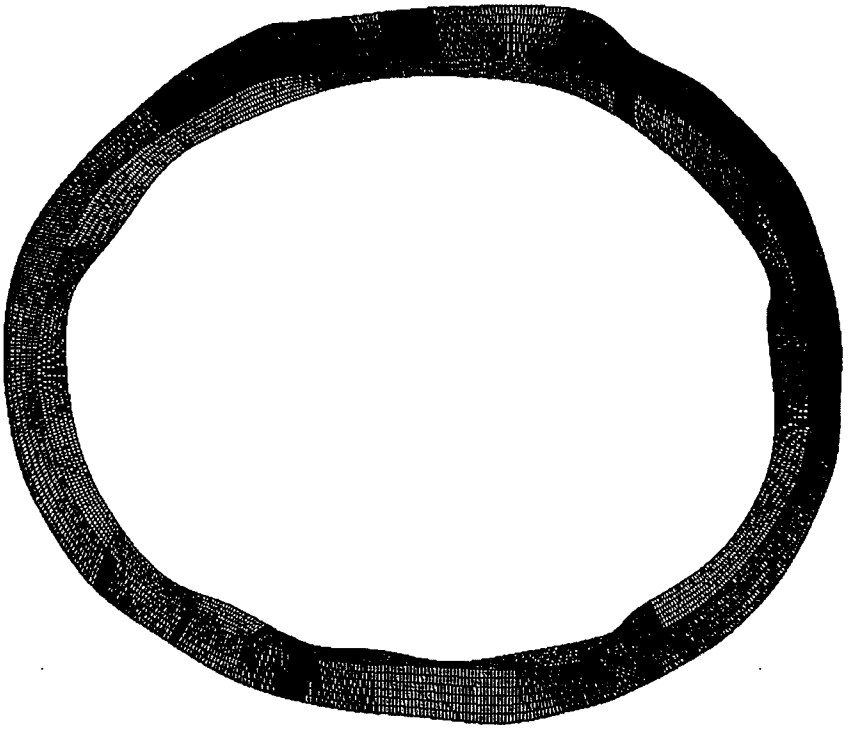


FIGURE. 4.10: Discretization of the skull and CSF using multiple Coons patches

The Coons patches used to represent the gray and white matters of the brain are shown in Figs.4.8, and 4.9. Also, the Coons patches of the skull and the CSF are shown in Fig.4.10. Note that the number of meshes at the patch interfaces are not necessarily equal. In the next section, the process of knot insertion to create comparable patches is described.

4.5 MULTI-PATCH CONFORMING GEOMETRIES

In this section, multi-patch analysis is performed for compatible patches in which the meshes and the control points coincide along the interface of two patches (even after refinement). The multi-patch domain, Ω, is formed by the union of n_p

patches such that:

$$\bar{\Omega} = \cup_{l=1}^{n_p} \bar{\Omega}_l,$$

If two patches share a common interface Γ_{ij} one has:

$$\bar{\Omega}_i \cap \bar{\Omega}_j = \Gamma_{ij}, \qquad i \neq j.$$

otherwise

$$\bar{\Omega}_i \cap \bar{\Omega}_j = 0, \qquad i \neq j.$$

Note that each patch is defined by its own mapping $\boldsymbol{F}_l : \hat{\Omega} \rightarrow \Omega_l$, with $\hat{\Omega}$ representing the unit square of the parent domain. Therefore, two patches, Ω_i and Ω_j, with a common interface $\Gamma_{ij} = \bar{\Omega}_i \cap \bar{\Omega}_j \neq 0$ coincide in their interface with identical multiplicity for the corresponding knot vectors.

Identical discretization of both patches in their interface results in identical basis functions along the common interface. In addition to the stated requirements, the parametric direction of the both meshes should match. Sharing a common knot vector, as well as control points along the interface, results in the same boundary mapping for both neighboring patches. Hence, a connectivity array can be defined to identify matching basis functions on each patch.

By identifying common control variables and control points in the interface, and by assigning a single entity for them in the global array in the assembly process, one can treat both patches as one patch with knots repeated p times at the interface. As discussed before, this repetition of knots will reduce the continuity to C^0 between two neighboring patches. The typical shape functions of two neighboring patches are

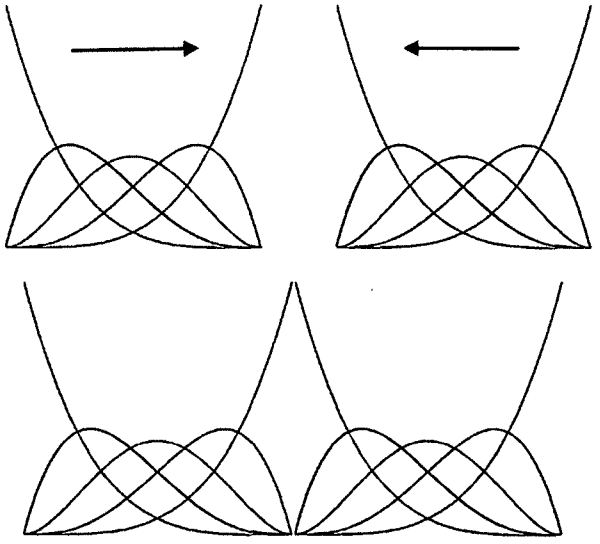
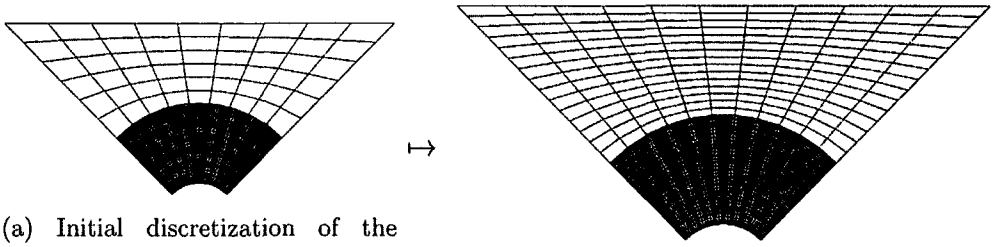


FIGURE. 4.11: Multi-patch conforming shape functions

shown in Fig.4.11. Multi-patch IGA is more complicated for vector field approximations such as electromagnetic field problems. This is because orientations for both tangential and normal components should be defined when applying curl-conserving transforms.

4.5.1 KNOT INSERTION FOR CONFORMING PATCHES

For simple geometries, one can define a set of identical control points and knot vectors along the interface of two neighboring patches. Recall that the brain section meshes was created by Coons patches. As discussed earlier, when generating a Coons patch, the opposite boundary curves are refined to have identical knot vectors. This refinement is done separately for each of two neighboring patches. Therefore, the common interface is defined over two separate knot vectors and consequently the resulting discretization does not satisfy the requirements of the multi-patch conforming



(a) Initial discretization of the neighboring patches

(b) Conforming patches after knot insertion

FIGURE. 4.12: Knot insertion used to generate compatible patches

geometries along the same interface.

To have conforming discretization on the interface, one can use knot insertion (similar to h-refinement) to redefine the underlying boundary knot of the interface. Figure 4.12 shows two neighboring Coons patches. Even though the same B-Spline curve used to define the interface of these two patches, the resulting discretization is non-conforming. The use of knot insertion to generate conforming patches is depicted in Fig.4.12b. After refinement, all the circular curves in Fig.4.12b are defined on a common knot.

Similarly, to create conforming patches for the brain section, 33 different sets of neighboring patches have been considered for refinement. After refinement, all the interfaces of each set are defined over a common knot vector. Figure 4.13 shows major refinement sets of patches used to modify the brain section discretization.

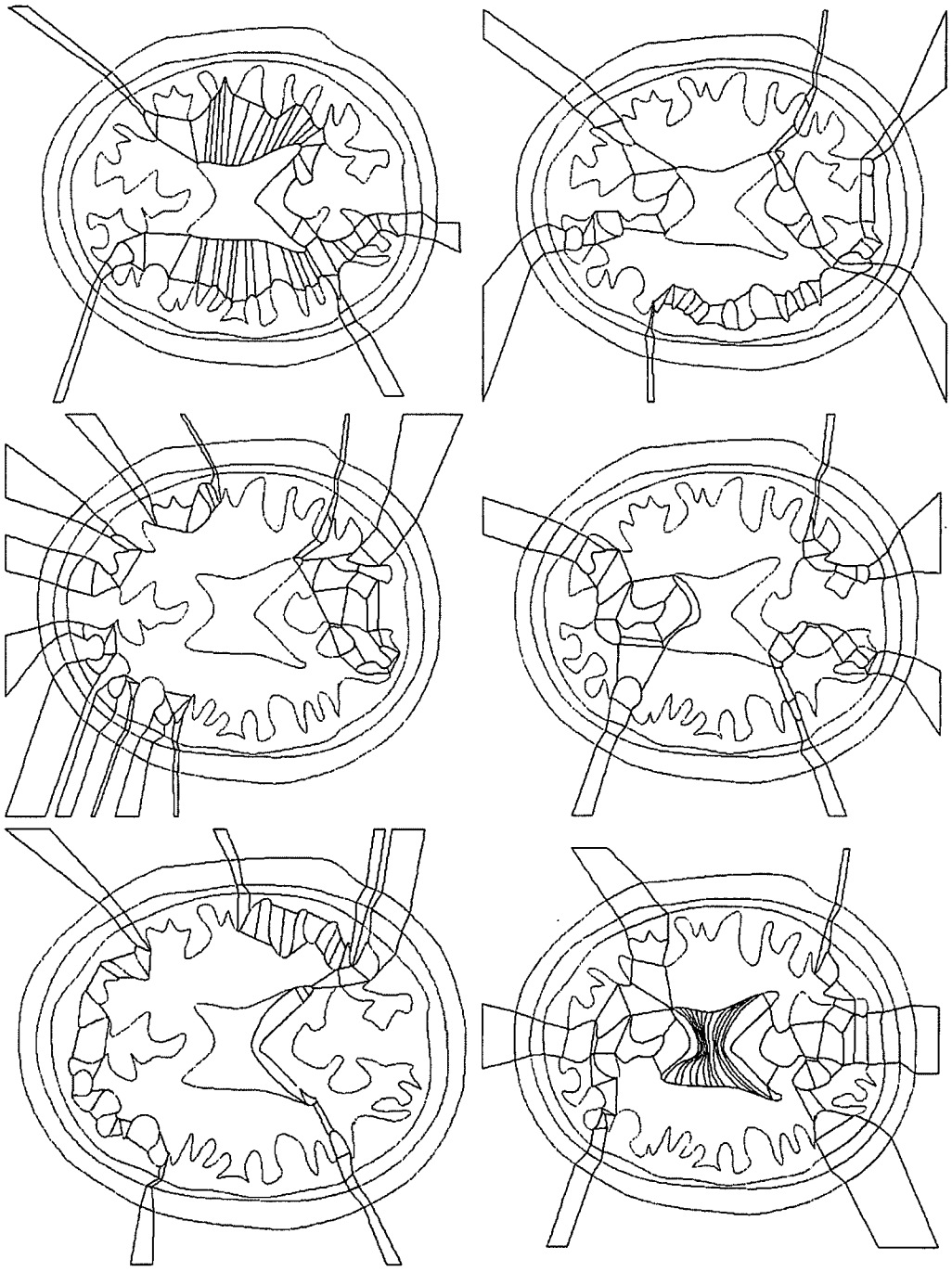


FIGURE. 4.13: Refinement is used for generation of conforming patches in the brain section model

CHAPTER 5

NUMERICAL RESULTS - EVALUATION OF IGA FOR ELECTROMAGNETISM

In this Chapter the IGA codes are developed based upon Chapters 3 for solving steady state and transient electromagnetic problems as well as electromagnetic scattering. The construction of the developed IGA package follows the excellent structure presented in [23]. The developed codes are evaluated by comparing the numerical results with analytical ones.

In Section 5.1, the steady state electric field is obtained on a rectangular domain and compared with the analytical solution. In Section 5.2, the time-dependent electromagnetic problem is described and numerical results are evaluated against the exact solution. Section 5.3 is dedicated to electromagnetic scattering.

The construction of the PML layer in the IGA context is explained in Section 5.4. The scattered and total electrical fields of an dielectric circular domain are obtained in Section 5.5 and numerical results are compared with the analytical ones. Then, the simplified circular model of the brain with realistic material properties is considered in Section 5.6. The scattered and total electrical fields for lossless and lossy materials are obtained using the developed IGA code for electromagnetic scattering which are evaluated against analytical solutions. The process described in Section 5.6 is then extended to a 2D human brain section. The results are presented in Section 5.7.

5.1 STEADY ELECTROMAGNETIC SOURCE PROBLEM

A Matlab code (the Matworks, Natick, MA, USA) has been prepared to estimate the steady and transient electromagnetic field over multiple-patch models. In this section, the simplified form of steady electromagnetic source problem is considered. The numerical results are compared against the analytical solution in a rectangular domain. The simplified strong form of the problem can be stated as:

$$\begin{aligned} \nabla \times (\nabla \times \boldsymbol{E}) + \boldsymbol{E} &= \boldsymbol{f} \qquad \text{in} \quad \Omega, \\ \boldsymbol{E} \times \boldsymbol{n} &= \boldsymbol{g} \qquad \text{on} \quad \Gamma_D, \\ (\nabla \times \boldsymbol{E}) \times \boldsymbol{n} &= \boldsymbol{h} \qquad \text{on} \quad \Gamma_N. \end{aligned}$$

The weak form of the problem with prescribed Dirichlet boundary on all sides, reads: Given $\boldsymbol{f} \in L^2(\Omega)$, find $\boldsymbol{u} \in \boldsymbol{H}_{0,\Gamma_D}(curl;\Omega)$ such that:

$$\int_\Omega \nabla \times \boldsymbol{u} \cdot \nabla \times \boldsymbol{w}\, d\Omega + \int_\Omega \boldsymbol{u} \cdot \boldsymbol{w}\, d\Omega = \int_\Omega \boldsymbol{f} \cdot \boldsymbol{w}\, d\Omega \quad \forall w \in \boldsymbol{H}_{0,\Gamma_D}(curl;\Omega), \quad (5.1.1)$$

The derivation of Eq.5.1.2 is provided in the Appendix.A. $\boldsymbol{H}_{0,\Gamma_D}(curl;\Omega)$ is the space of square integrable vectorial functions in Ω such that their curl is also square integrable and their tangential components are null on the boundary Γ_D:

$$H_0(curl;\Omega) := \{\boldsymbol{v} \in L^2 : \nabla \times \boldsymbol{v} \in L^2, \boldsymbol{n} \times \boldsymbol{v} = \boldsymbol{0} \text{ on } \Gamma_D\} \qquad (5.1.2)$$

This is reduced to finite dimensional problem by creating finite dimensional sub-spaces $\boldsymbol{U}^k \subset \boldsymbol{U}$, and $\boldsymbol{V}^k \subset \boldsymbol{V}$ both formed through B-Spline (NURBS) basis. Denoting $\boldsymbol{g}^k \subset \boldsymbol{U}^k$ as the finite dimensional representation of $\boldsymbol{g}$, the solution can be written as $\boldsymbol{u}^h = \boldsymbol{v}^h + \boldsymbol{g}^h$, which is valid for all $\boldsymbol{v}^h \in \boldsymbol{V}^h$. Note that the boundary data is built in the solution by inclusion of the term $\boldsymbol{g}^h$. The approximated solution

can be written in terms of NURBS basis functions as:

$$\boldsymbol{u}^h = \sum_{B=1}^{n_{eq}} d_B N_B + \sum_{B=n_{eq}+1}^{n_{np}} g_B N_B, \tag{5.1.3}$$

where n_{eq} is the number of unknowns (equations to be solved), and n_{np} is the total number of degrees of freedom. Also, $\{d_B\}_{B=1}^{n_{eq}}$ is a set of all unknown control variables and $\{g_B\}_{B=n_{eq}+1}^{n_{np}}$ is a set of known Dirichlet control variables. The test function is similarly discretized:

$$\boldsymbol{w}^h = \sum_{A=1}^{n_{eq}} c_A N_A. \tag{5.1.4}$$

Substituting Eq.5.1.4 and Eq.5.1.3 in the weak form of the problem, Eq.5.1.1 and noting that $\{c_A\}$ is arbitrary, for any index A, the discrete form becomes:

$$\sum_{B=1}^{n_{eq}} \int_\Omega d_B (\nabla \times N_A) \cdot (\nabla \times N_B) d\Omega + \sum_{B=1}^{n_{eq}} \int_\Omega d_B N_A N_B d\Omega$$
$$= \int_\Omega f_A N_A d\Omega - \sum_{B=n_{eq}+1}^{n_{np}} \int_\Omega g_B (\nabla \times N_B)(\nabla \times N_A) d\Omega - \sum_{B=n_{eq}+1}^{n_{np}} \int_\Omega g_B N_A N_B d\Omega$$

This is a linear system of equations that can be re-written as:

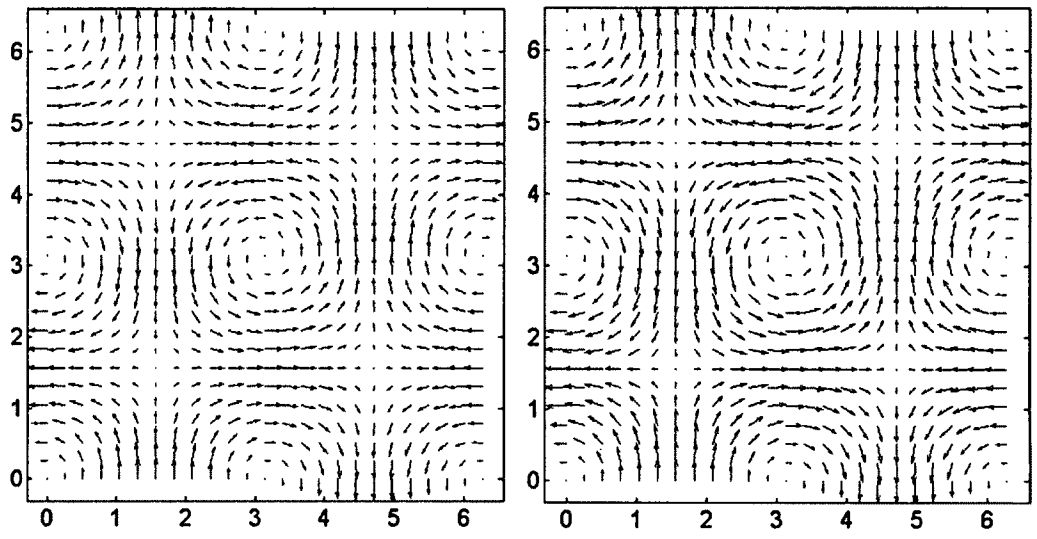
$$(\boldsymbol{A} + \boldsymbol{B})\boldsymbol{d} = \boldsymbol{f}.$$

To evaluate the IGA code, a manufactured problem is set up to yield the following solution:

$$E_x = -\frac{1}{\pi} \cos(x) \sin(y),$$
$$E_y = \frac{1}{\pi} \sin(x) \cos(y),$$

Consequently, the source term is defined as follows:

$$f_x = -\frac{3}{\pi} \cos(x) \sin(y),$$
$$f_y = +\frac{3}{\pi} \sin(x) \cos(y).$$



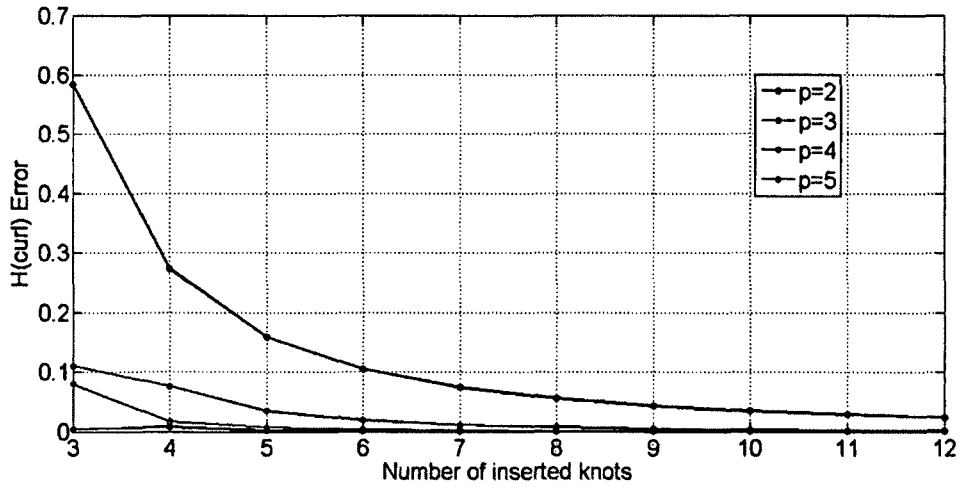
(a) Computed Solution (b) Exact Solution

FIGURE. 5.1: Comparing the computed electrical field with exact solution

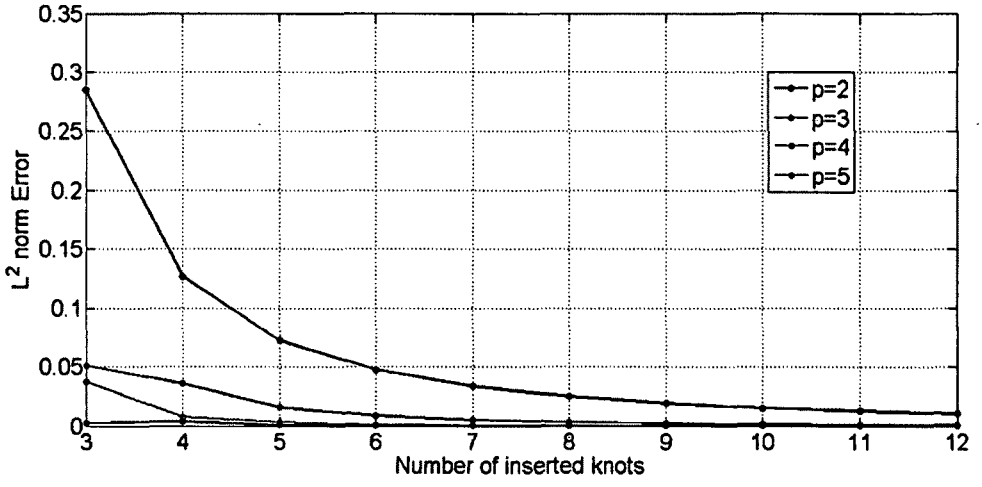
A square domain is considered with side length equal to 2π, $\Omega = [0, 2\pi] \otimes [0, 2\pi]$. The parameter space is defined by two open knot vectors: $u_\xi = u_\eta = \{0, 0, 0, 1, 1, 1\}$ with quadratic basis, $p = 2$. Dirichlet boundary conditions are applied using the projection method which was described in Section 2.6.1. Figure 5.1 compares the computed solution with the exact solution. The initial mesh was defined very coarse to show the effect of the refinement on the accuracy of the solution. Denoting the computed and the exact electrical fields with E_{comp}, and E_{exact}, the error is computed using the following equation:

$$Err = ||E_{exact} - E_{comp}|| \tag{5.1.5}$$

where $||\ .\ ||$ is either the $\mathbf{L}^2$-norm: $||\boldsymbol{E}_{exact} - \boldsymbol{E}_{comp}||_{0,\Omega}$ or the $\mathbf{H(curl)}$ norm: $||Curl(\boldsymbol{E}_{exact}) - Curl(\boldsymbol{E}_{comp})||_{curl,\Omega}$. Figure 5.2a and Fig.5.2b show the effects of



(a) The effects of order elevation and knot insertion on Hcurl error



(b) The effects of order elevation and knot insertion on L_2 error

FIGURE. 5.2: Increasing the accuracy of the solution by refinement

order elevation and knot insertion on the **H(curl)**-norm and $\mathbf{L}^2$-norm errors respectively. It can be seen that in the IGA, the order elevation and the knot insertion are very effective in increasing the convergence rates.

5.2 TRANSIENT ELECTROMAGNETIC SOURCE FIELD

Recall the second order time-dependent Maxwell equation (Eq.3.1.11), which is the second order form of this equation and is commonly solved in time dependent electromagnetic problems:

$$\varepsilon \boldsymbol{E}_{,tt} + \sigma \boldsymbol{E}_{,t} + \nabla \times (\frac{1}{\mu} \nabla \times \boldsymbol{E}) = \boldsymbol{J}_t^{imp}$$

In order to obtain the electrical field, the initial condition, $\boldsymbol{E}$, and $\frac{\partial \boldsymbol{E}}{\partial t}$ at time $t = 0$ must be prescribed. To evaluate the IGA code for transient electromagnetic field, an analytical problem is considered. The problem is imposed with Dirichlet boundary conditions only. The strong form of the problem reads:

$$\begin{aligned} \frac{\partial^2 \boldsymbol{E}}{\partial t^2} + \nabla \times \nabla \times \boldsymbol{E} = \boldsymbol{f} \quad \text{in} \quad \Omega, \\ \boldsymbol{E} \times \boldsymbol{n} = \boldsymbol{g} \quad \text{on} \quad \Gamma_D, \end{aligned} \tag{5.2.1}$$

also, the initial conditions are given by:

$$\boldsymbol{E}(\boldsymbol{x}, 0) = 0 \quad \text{on} \quad \Omega,$$

$$\frac{\partial \boldsymbol{E}}{\partial t}(\boldsymbol{x}, 0) = \begin{pmatrix} -\cos(x)\sin(y) \\ \sin(x)\cos(y) \end{pmatrix} \quad \text{on} \quad \Omega.$$

The source is described as:

$$\boldsymbol{f} = \begin{pmatrix} -\frac{2-\pi^2}{\pi}\cos(x)\sin(y)\sin(\pi t) \\ \frac{2-\pi^2}{\pi}\sin(x)\cos(y)\sin(\pi t) \end{pmatrix},$$

The exact solution of the problem is given by:

$$\boldsymbol{E} = \begin{pmatrix} -\frac{1}{\pi}\cos(x)\sin(y)\sin(\pi t) \\ \frac{1}{\pi}\sin(x)\cos(y)\sin(\pi t) \end{pmatrix}.$$

The homogeneous Dirichlet boundary conditions is applied on all four sides of the rectangle. The weak form of the problem reads:

Given $\boldsymbol{f} \in L^2(\Omega)$, find $u \in \boldsymbol{H}_{0,\Gamma_D}(curl;\Omega)$ such that

$$\int_\Omega (\nabla \times \boldsymbol{u}) \cdot (\nabla \times \boldsymbol{v}) + \frac{d^2}{dt^2} \int_\Omega \boldsymbol{uv} = \int_\Omega \boldsymbol{f} \cdot \boldsymbol{v} d\Omega \quad \forall v \in \boldsymbol{H}_{0,\Gamma_D}(curl;\Omega).$$

To solve this time dependent problem the semi-discrete method is used. As discussed in section 2.8.1, in this approach the space is discretized using the Galerkin method while, time is considered to remain continuous. Consequently, the shape functions are defined as functions of the space only (similar to static case) and corresponding coefficients are defined as functions of time. Therefore, the solution is a linear combination of space-dependent shape functions and time-dependent coefficients. The space of test functions, $\boldsymbol{V}$, will be similar to static case but, the space of trial functions will be time dependent satisfying the dynamic conditions:

$$S_t = \{\boldsymbol{u}(\cdot,t) | u_i(\boldsymbol{x},t) \times \boldsymbol{n} = g_i(\boldsymbol{x},t), \boldsymbol{x} \in \Gamma_D, u(\cdot,t) \in H_{0,\Gamma_D}(curl;\Omega)\}$$

The discretized solution can be written as:

$$\boldsymbol{u}^h = \sum_{B=1}^{n_{eq}} d_B N_B + \sum_{B=n_{eq}+1}^{n_{np}} g_B N_B,$$

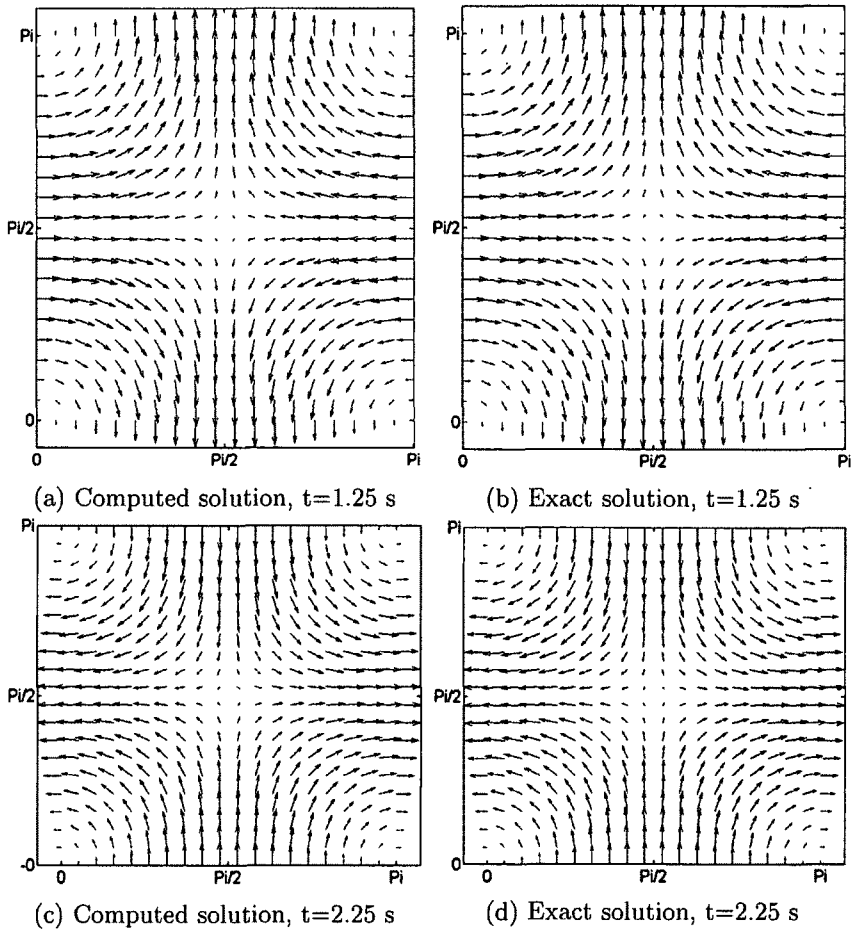
and the test functions are given as:

$$\boldsymbol{w}^h = \sum_{A=1}^{n_{eq}} c_A N_A,$$

$$\sum_{B=1}^{n_{eq}} \int_{\Omega} d_B(t)(\nabla \times N_A) \cdot (\nabla \times N_B) d\Omega + \frac{d^2}{dt^2} \sum_{B=1}^{n_{eq}} \int_{\Omega} d_B N_A N_B d\Omega$$
$$= \int_{\Omega} F N_A d\Omega - \sum_{B=n_{eq}}^{n_{np}} \int_{\Omega} g_B (\nabla \times N_B)(\nabla \times N_A) d\Omega$$
$$- \frac{d^2}{dt^2} \sum_{B=n_{eq}+1}^{n_{np}} \int_{\Gamma_D} g_B N_A N_B d\Gamma_D$$

which is reduced to an ordinary differential equation (ODE):

$$\boldsymbol{A} \frac{d^2}{dt^2} \boldsymbol{d}_B + \boldsymbol{B} \boldsymbol{d}_B = \boldsymbol{C}$$



(a) Computed solution, t=1.25 s (b) Exact solution, t=1.25 s

(c) Computed solution, t=2.25 s (d) Exact solution, t=2.25 s

FIGURE. 5.3: Transient electrical filed, comparing the computed solution with the exact solution

where

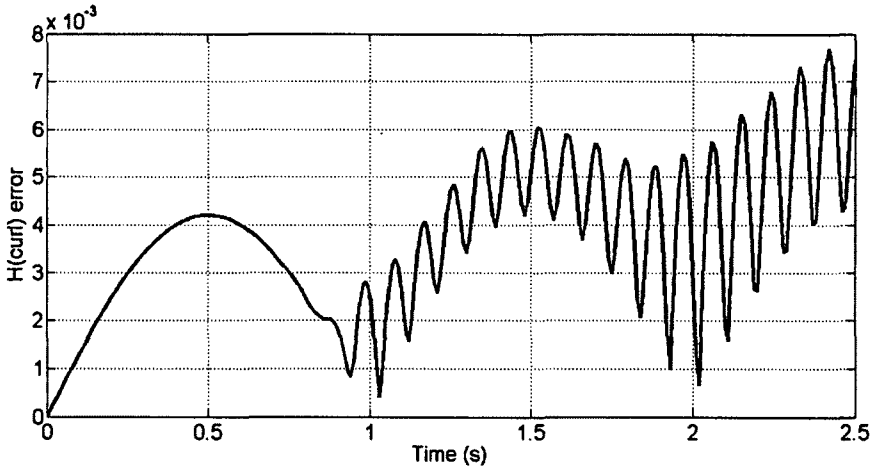
$$\boldsymbol{A} = \sum_{B=1}^{n_{eq}} \int_{\Omega} d_B N_A(x) N_B(x) d\Omega,$$

$$\boldsymbol{B} = \sum_{B=1}^{n_{eq}} \int_{\Omega} d_B(t) (\nabla \times N_A)(\nabla \times N_B) d\Omega,$$

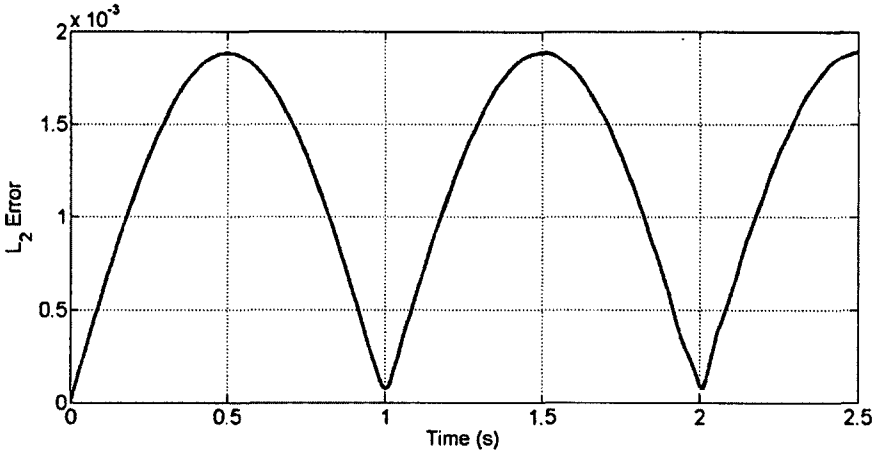
$$\boldsymbol{C} = \int_{\Omega} F N_A d\Omega - \sum_{B=n_{eq}}^{n_{np}} \int_{\Omega} g_B (\nabla \times N_B)(\nabla \times N_A) d\Omega$$

$$- \frac{d^2}{dt^2} \sum_{B=n_{eq}+1}^{n_{np}} \int_{\Gamma_D} g_B N_A N_B d\Gamma_D.$$

The computed solution is compared with the analytical solution at $t = 1.25\ s$ and



(a) Transient electric field - $H(curl)$ error vs time



(b) Transient electric field - L^2 error vs time

FIGURE. 5.4: Transient electrical field - $H(curl)$ and L^2 norm errors vs time

$t = 2.25\ s$ in Fig.5.3. The $H(curl)$ and L_2 errors versus time are depicted in Fig.5.4a and Fig. 5.4b respectively.

5.3 ELECTROMAGNETIC SCATTERING

In this study, the incident wave is modeled as a monochromatic plane wave:

$$\mathbf{E}_{inc} = \mathbf{E}_0 e^{-\jmath \mathbf{k}\mathbf{x}}, \tag{5.3.1}$$

the polarization vector, $\mathbf{E}_0$, and propagating wave vector, $\mathbf{k}$, should satisfy the following conditions:

$$\mathbf{E}_0 \cdot \mathbf{k} = 0. \tag{5.3.2}$$

where

$$\mathbf{E}_0 = (0,1,0), \quad \text{and} \quad \mathbf{k} = (\pi,0,0)^T.$$

Particularly this plane wave is traveling along x direction and polarized along y direction as shown in Fig.5.5. When an electromagnetic wave hits an scatterer made

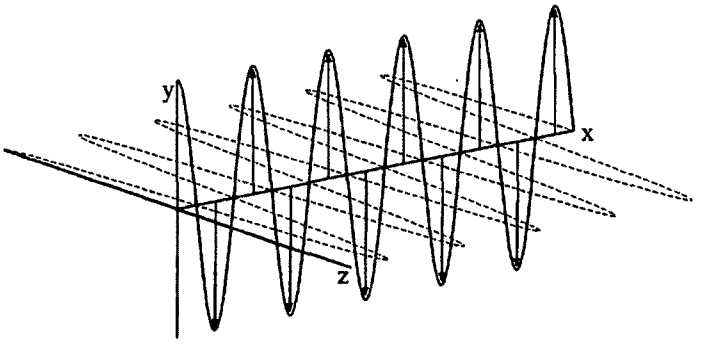


FIGURE. 5.5: Plane wave traveling along x direction, polarized along y direction

of lossless material a portion of it is reflected and another part is transmitted into the scatterer. In other words, the illuminated body will allow a portion of the wave to pass through and reflects the rest.

When the scatterer is made of lossy material, in addition to these two portions, a third portion of the electromagnetic wave is absorbed and converts to heat. It is desired to find the scattered and the total electric fields using IGA. The basics of electromagnetic scattering and the required formulation were presented in Section 3.4. The study of electromagnetic scattering within a finite domain is not possible without mimicking the infinite domain using a PML layer introduced in Section 3.5. The construction of the required PML layer is described in the next section.

5.4 PERFECTLY MATCHING LAYER IN IGA

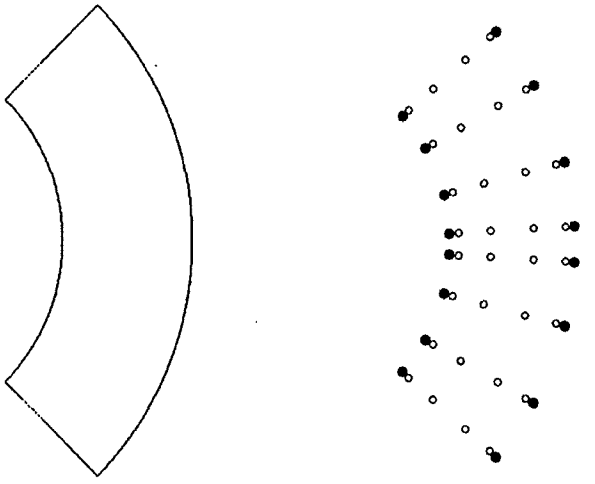
Recall the bilinear form of the scattering problem in the PML discussed in Section 3.5, the bilinear form involves the evaluation of Eq.3.5.4:

$$\tilde{\mathrm{a}}(\tilde{\boldsymbol{E}}, \tilde{\boldsymbol{F}}) = \int_{\Omega} \big(\frac{1}{\mu_r} \mathbf{A}^{-1} \nabla_{\boldsymbol{x}} \times \boldsymbol{E} \cdot \nabla_{\boldsymbol{x}} \times \boldsymbol{F} - \omega^2(\varepsilon_r - j\varepsilon_\sigma)\mathbf{A}\boldsymbol{E} \cdot \boldsymbol{F}\big) d\Omega$$

where $\mathbf{A} = \tilde{J}\tilde{\boldsymbol{J}}^{-1}\tilde{\boldsymbol{J}}^{-T}$ and $\tilde{\boldsymbol{J}}$ is the Jacobian of the complex stretching variable denoted by $\tilde{\boldsymbol{x}}$ and defined as:

$$\tilde{\boldsymbol{x}} = \boldsymbol{x}\big(1 - \frac{j}{k}(\frac{\boldsymbol{x} - \boldsymbol{x}_a}{\boldsymbol{x}_b - \boldsymbol{x}_a})^{\alpha}\big), \tag{5.4.1}$$

where $\boldsymbol{x}$ is a set of physical points (physical mapping) obtained in each parametric direction. Similarly, $\boldsymbol{x}_a$ and $\boldsymbol{x}_b$ are the physical boundaries of the PML patch, α is a tuning parameter and k is the free space wave number. The approach used in this study to construct the PML layer in IGA is illustrated in Figure 5.6. The PML patch is shown in Fig.5.6a where the wave is traveling from the inner radius to the outer boundary. Along the outer boundary, the wave should decay to zero without



(a) A single PML patch

(b) Evaluation points

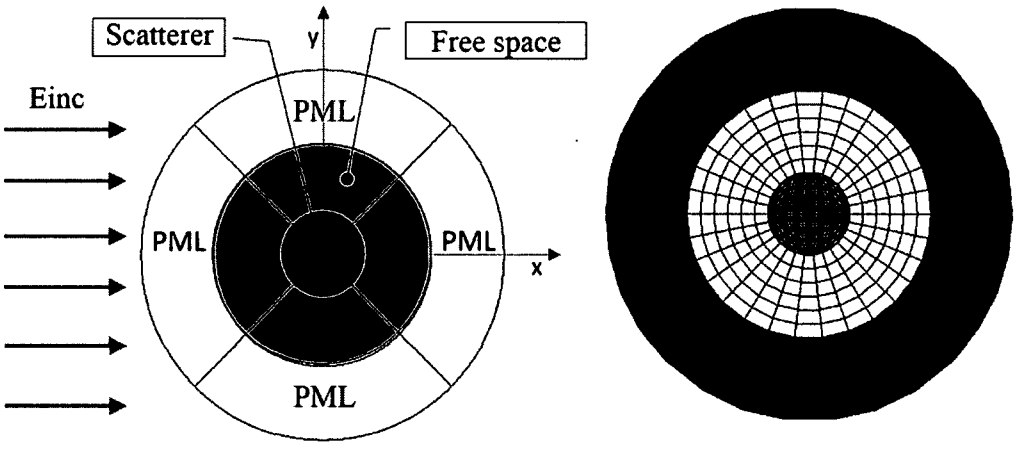
FIGURE. 5.6: PML construction in IGA

reflection. The $\boldsymbol{x}_a$ in this example are defined as the coordinates of the inner curve and $\boldsymbol{x}_b$ are the coordinates of the points on the exterior boundary. Therefore, the complex stretching variable, defined in Eq.5.4.1, is only applied for the inner points between the inner and the outer curves shown with hollow circles in Fig.5.6b.

The homogeneous Dirichlet boundary condition is applied to the boundary points. Note that the propagating wave in the PML layer vanishes regardless of the application of the boundary condition. This is necessary to have a reflection-less boundary layer which mimics the infinite domain. The example illustrated in Fig. 5.6 has a very simple geometry and therefore is not fully representing the advantages of the PML layer in IGA compared to the one in conventional FEM. The boundaries of the PML layer in IGA can represent the boundary of the scatterer exactly. The scattering on a circular domain and the PML layer in IGA are be discussed and evaluated in the next section.

5.5 SCATTERING ON A DIELECTRIC CIRCLE

As the first example the electromagnetic scattering problem is solved on a circular dielectric domain with a diameter $d = 2r = \lambda = 2$ where λ represents the wave length in free space. The model is composed of a scatterer which is modeled as a circle ($r = 1$) surrounded by free space, $1 < r < 3$, and the PML layer, $3 < r < 5$, as shown in Fig.5.7a. The discretized model is shown in Fig.5.7b.



(a) Layout of the patches (b) Initial mesh

FIGURE. 5.7: Dielecteric scattering - model generation

The frequency of the electromagnetic wave can be described as:

$$\text{frequency} = \frac{\omega}{2\pi a\sqrt{\epsilon_0\mu_0}}, \tag{5.5.1}$$

where a is a reference length which is determined based on the scatterer size: for the brain section, $a = 0.1m$. The frequency of the incident wave in this example corresponds to 1.5 GHz. A Matlab code (the Matworks, Natick, MA, USA) has been prepared to estimate the scattered and the total fields in IGA context.

Three sub-domains are considered to model the analysis domain which represent the scatterer, the free space, and the PML layer, respectively. The sub-domains representing the free space and the PML layer are composed of four patches each as shown in Fig.5.7a. The required Coons patches are shown in Fig.5.7b which are refined to satisfy the conforming patch requirements.

The material properties of the scatterer is tabulated in Table 1. This model is hit by the monochromatic plane wave identical to the one described in Eq.5.3.1.

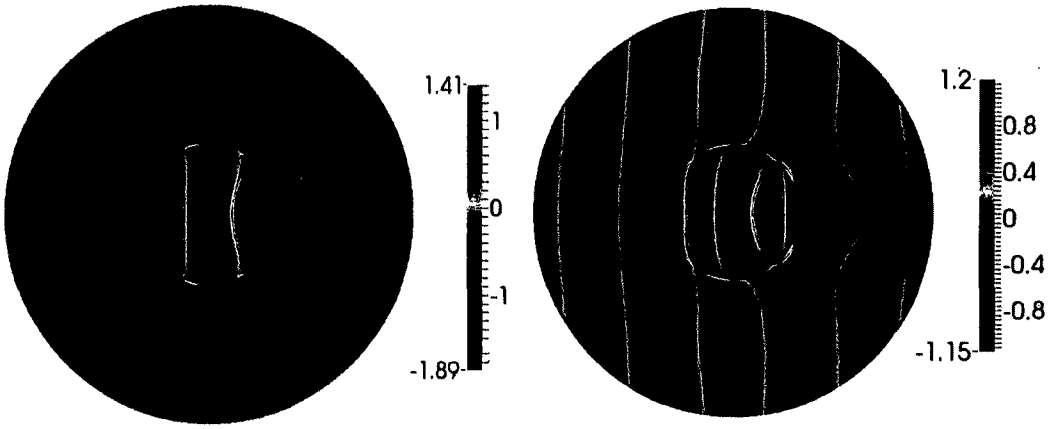
$$\mathbf{E}_{inc} = \mathbf{E}_0 e^{-j\mathbf{kx}}. \tag{5.5.2}$$

The real and imaginary parts of the scattered and total electrical field are depicted in Fig.5.8 and Fig.5.9. As expected, the x component of the scattered field is identical to the x component of the total field.

TABLE 1: Material properties of the dielectric scatterer

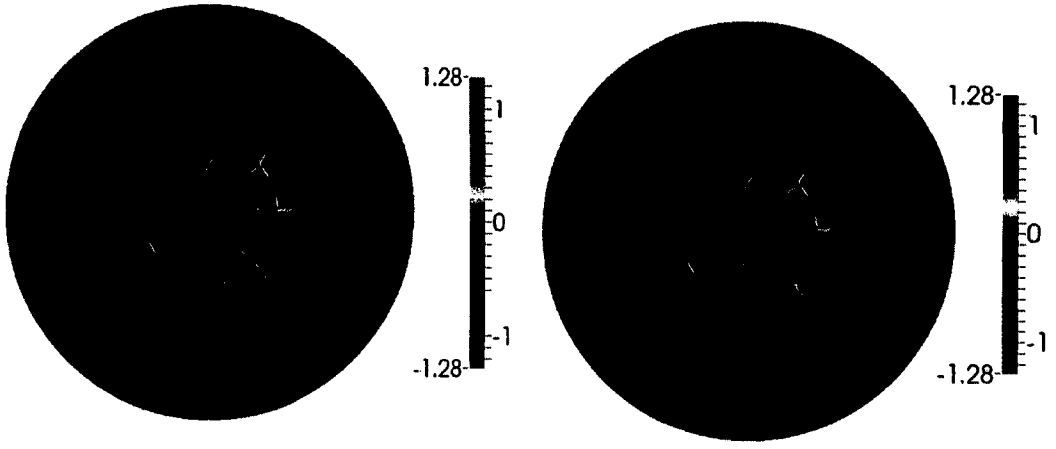
Dielectric circle	ε_r	μ_r	σ_r
Material Properties	4.0	1.0	0.0

To evaluate the numerical results, the computed solution is compared to the analytical solution obtained using the code generated by Schäfer [66]. The absolute value of the scattered field is compared with the analytical solution in Fig.5.10. To further evaluate the numerical results the solution along the x axis is compared with the analytical solution in Fig.5.11.



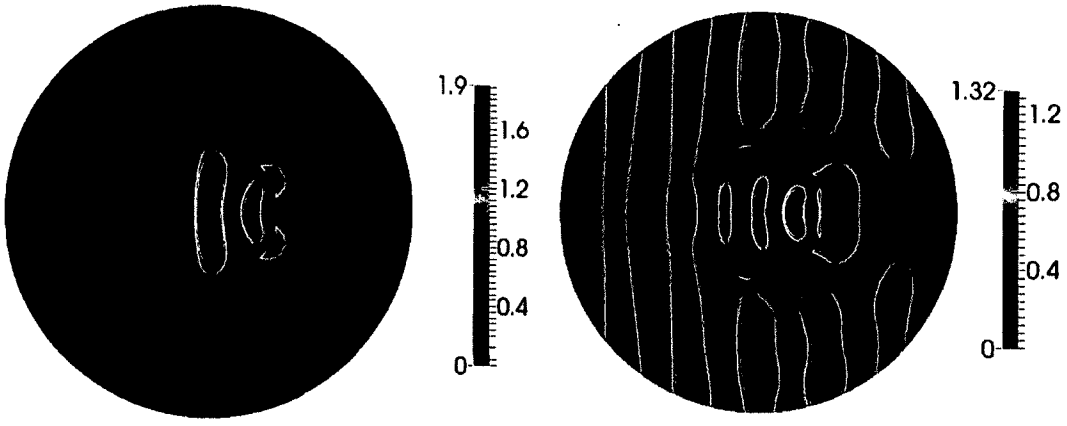
(a) Scattered field - y component

(b) Total field - y component



(c) Scattered field - x component

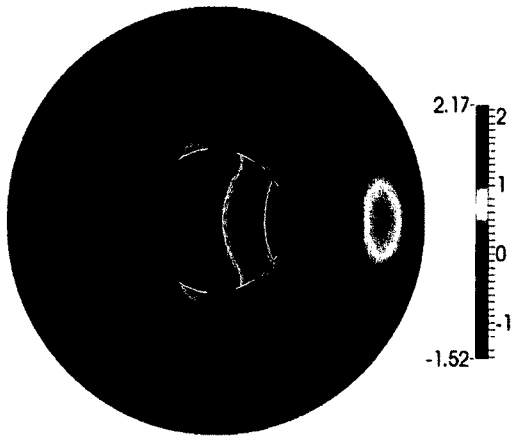
(d) Total field - x component



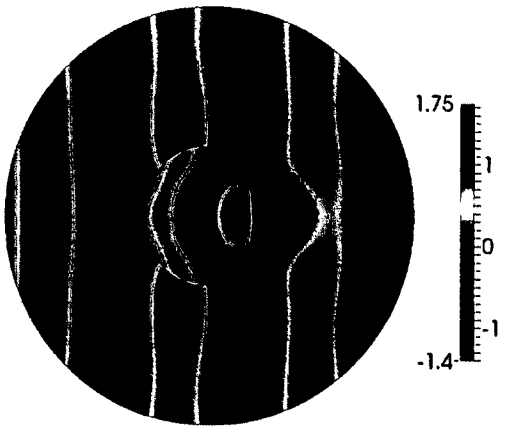
(e) Magnitude of the scattered field - $Abs(E_{scat})$

(f) Magnitude of the total field - $Abs(E_{total})$

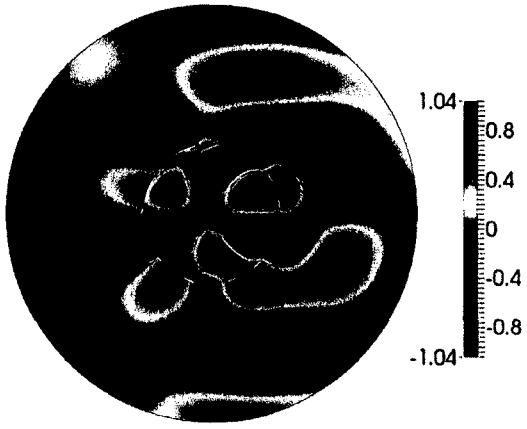
FIGURE. 5.8: Dielectric scattering - real part of the scattered and total electrical fields



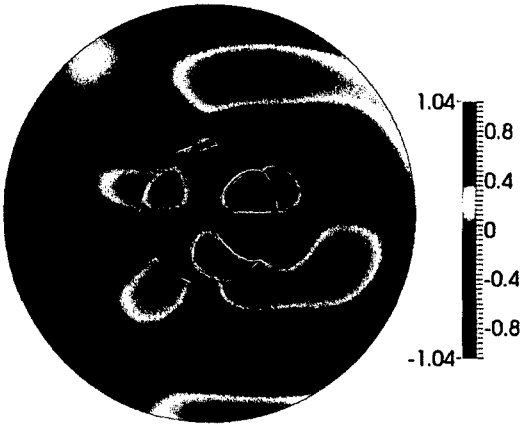
(a) Scattered field - y component



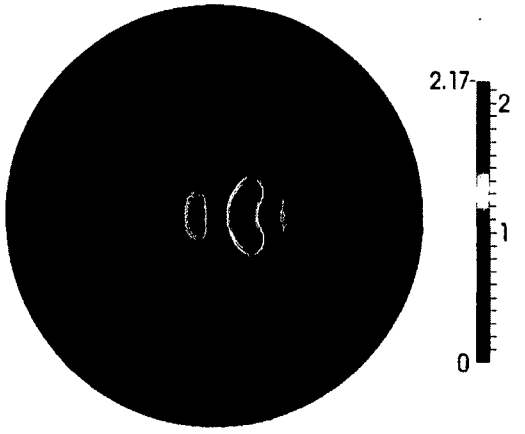
(b) Total field - y component



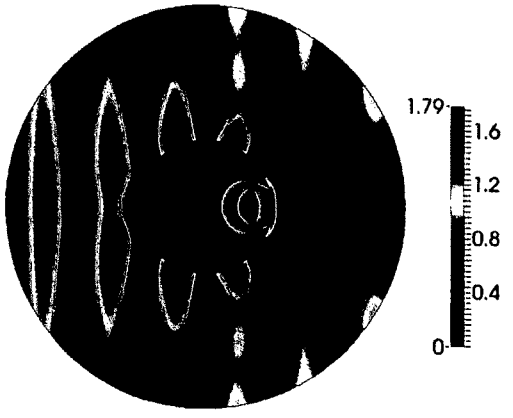
(c) Scattered field - x component



(d) Total field - x component

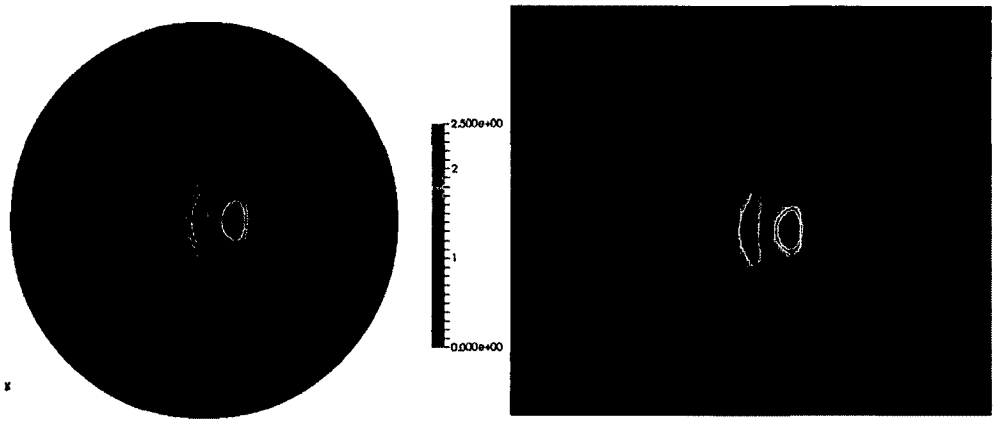


(e) Magnitude of the scattered field



(f) Magnitude of the total field

FIGURE. 5.9: Dielectric scattering - imaginary part of the scattered and total electrical fields



(a) $Abs(E_{scat})$ - Computed solution (b) $Abs(E_{scat})$ - Analytical solution

FIGURE. 5.10: Dielectric scattering - comparison of the computed $Abs(E_{scat})$ with the analytical solution

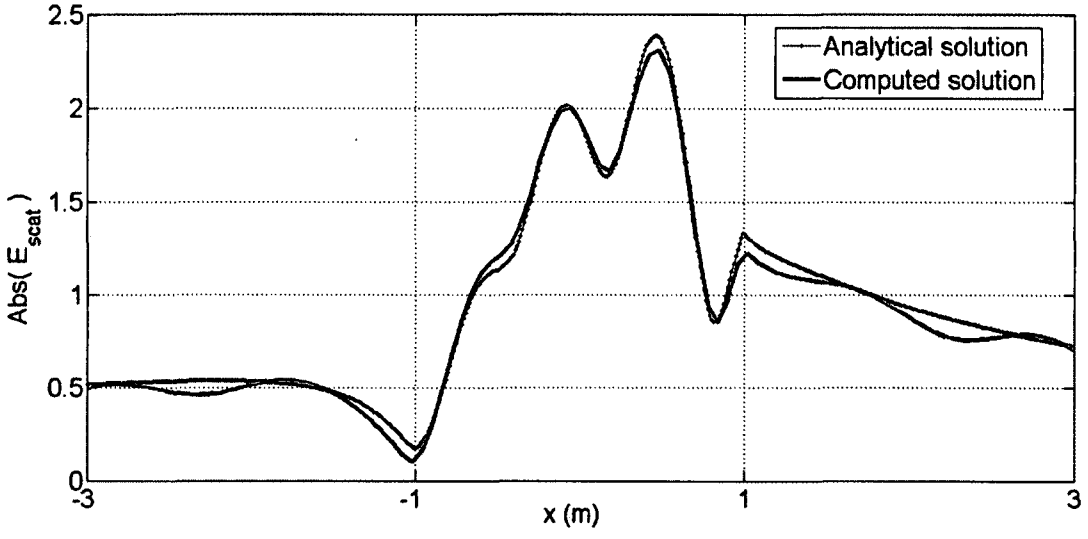
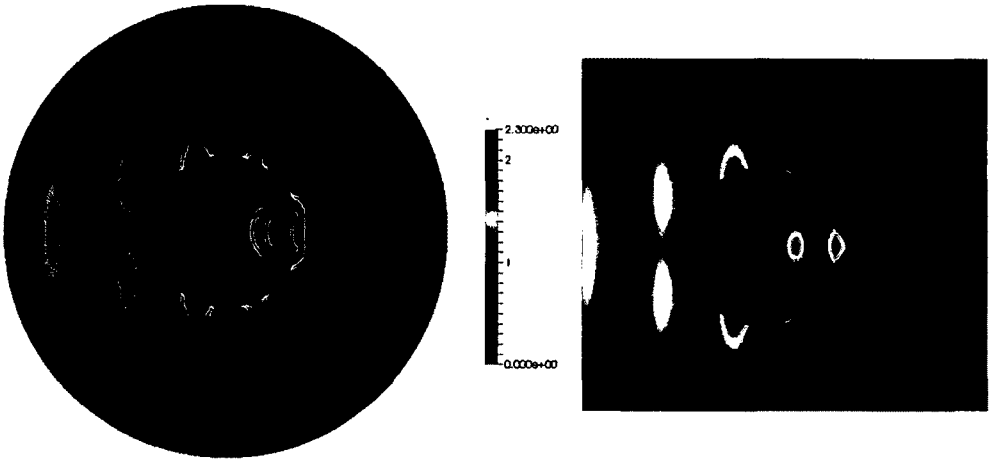


FIGURE. 5.11: Dielectric scattering - comparison the computed $Abs(E_{scat})$ with analytical solution along x axis



(a) $Abs(E_{total})$ - Computed solution

(b) $Abs(E_{total})$ - Analytical solution

FIGURE. 5.12: Dielectric scattering - Comparison of the computed $Abs(E_{total})$ with the analytical solution

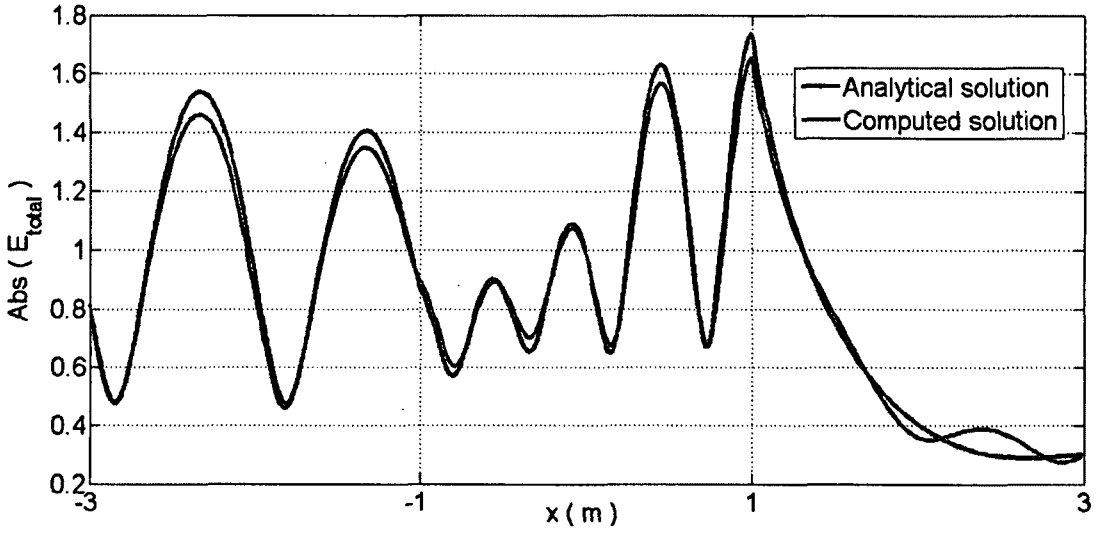


FIGURE. 5.13: Comparison the computed $Abs(E_{total})$ analytical solution along x axis

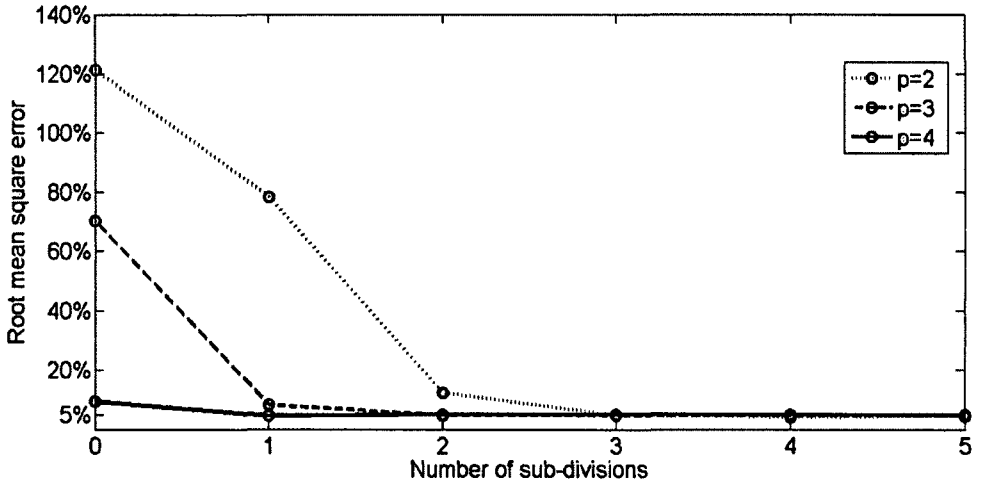


FIGURE. 5.14: Dielectric scatterer - root mean square error along x axis

Similarly, the numerical solution of the total electrical field is compared with the analytical solution in Fig.5.12 and Fig.5.13. Note that the computed field inside the scatterer has a better agreement with the analytical solution. The error seen in the free space of the computed electrical field is related to the definition and tuning of the PML layer. Better results may be achieved through tuning and optimization of the PML layer. However, PML optimization is not performed in this study.

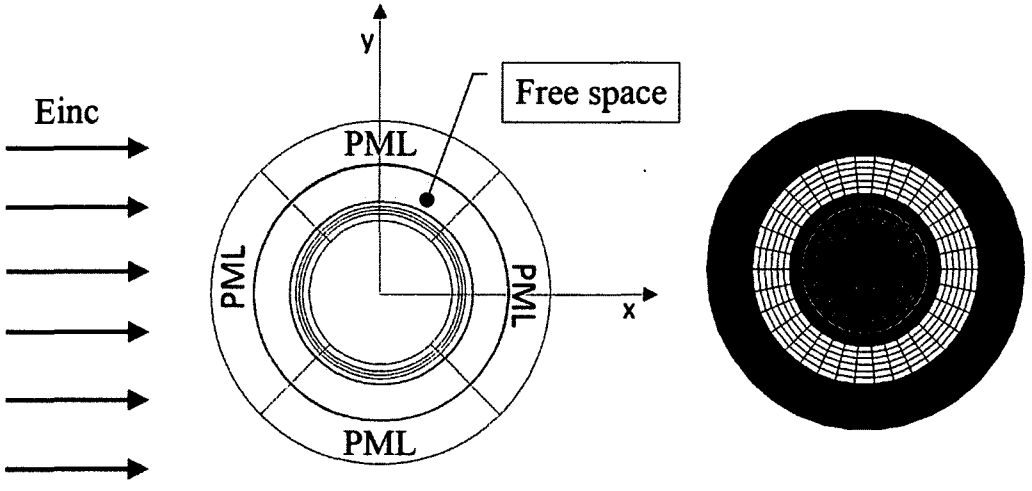
To quantify the error of the numerical results obtained for the dielectric scatterer, the root mean square error of the electrical field magnitude was computed along the x axis crossing the scatterer and the free space. As a result, the computed error captures the variations of the computed solution from the analytical one in the free space. The solution is not acceptable for the order $p = 2$ with the initial discretization. However, the error is reduced the to $\sim 7\%$ by increasing the order to $p = 4$ without increasing the number of sub-divisions. As shown in Fig.5.14, further refinement reduced the error to less than 5%.

5.6 FOUR LAYER BRAIN MODEL

In this section a four layered circular head model is considered to analytically validate the numerical results. The multipatch model and the corresponding mesh are shown in Fig.5.15a and Fig.5.15b. The four layers are representing the brain, the Cerebro-Spinal-Fluid (CSF), the skull, and the skin. Since the geometry of the inner layers of the brain cannot be represented with circles, the average permittivity and conductivity is used as presented in Table 2 [63].

TABLE 2: Material properties of the circular brain section

Layer	thickness	ε_r	σ	μ_r
Brain	0.086 m	46.509064	0.712895	1.0
CSF	0.008 m	69.005669	2.355479	1.0
Skull	0.004 m	16.872126	0.215958	1.0
Skin	0.004 m	42.316116	0.816917	1.0



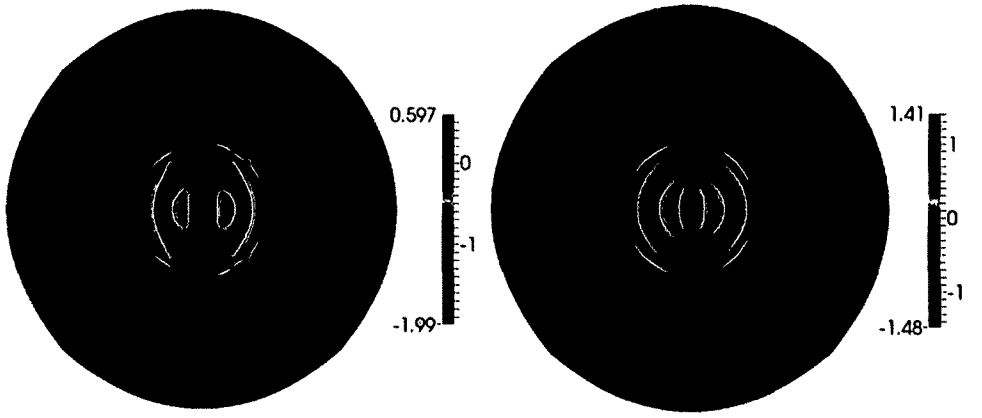
(a) Layout of the patches

(b) Initial mesh

FIGURE. 5.15: Circular brain model generation

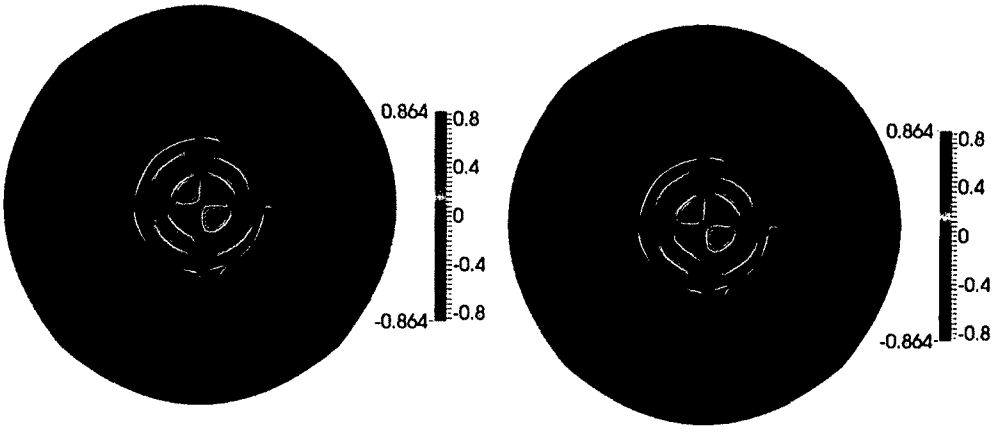
This simplified head model is surrounded by a circle representing the free space and by a PML layer which mimics the infinite space. The incident wave is a monochromatic plane wave as described by Eq.5.3.1. The frequency of the incident wave is 750 MHz. For the sake of comparison in the first set of the results the materials are incorrectly assumed to be lossless. The real and imaginary parts of the estimated scattered and the total fields for lossless materials are shown in Fig.5.16 and Fig.5.17.

The computed absolute value of the scattered field is compared with the analytical solution of the model with lossless materials in Fig.5.18. The numerical and the analytical solutions are also compared along the x axis in Fig. 5.19. Similarly, the computed total field is compared with the analytical solution in Fig.5.20 and Fig.5.21. Note that the assumption of the lossless properties is no applicable to the real brain problem. It is presented for the purpose of comparison with the results obtained for lossy materials.



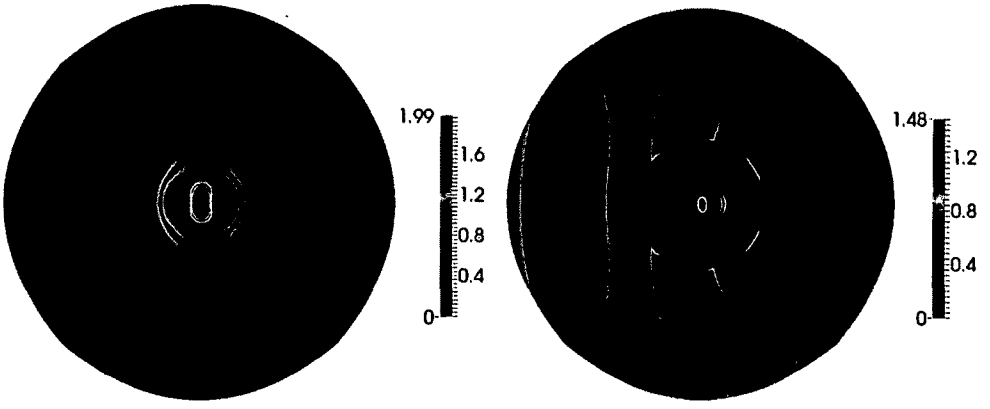
(a) Scattered field - y component

(b) Total field - y component



(c) Scattered field - x component

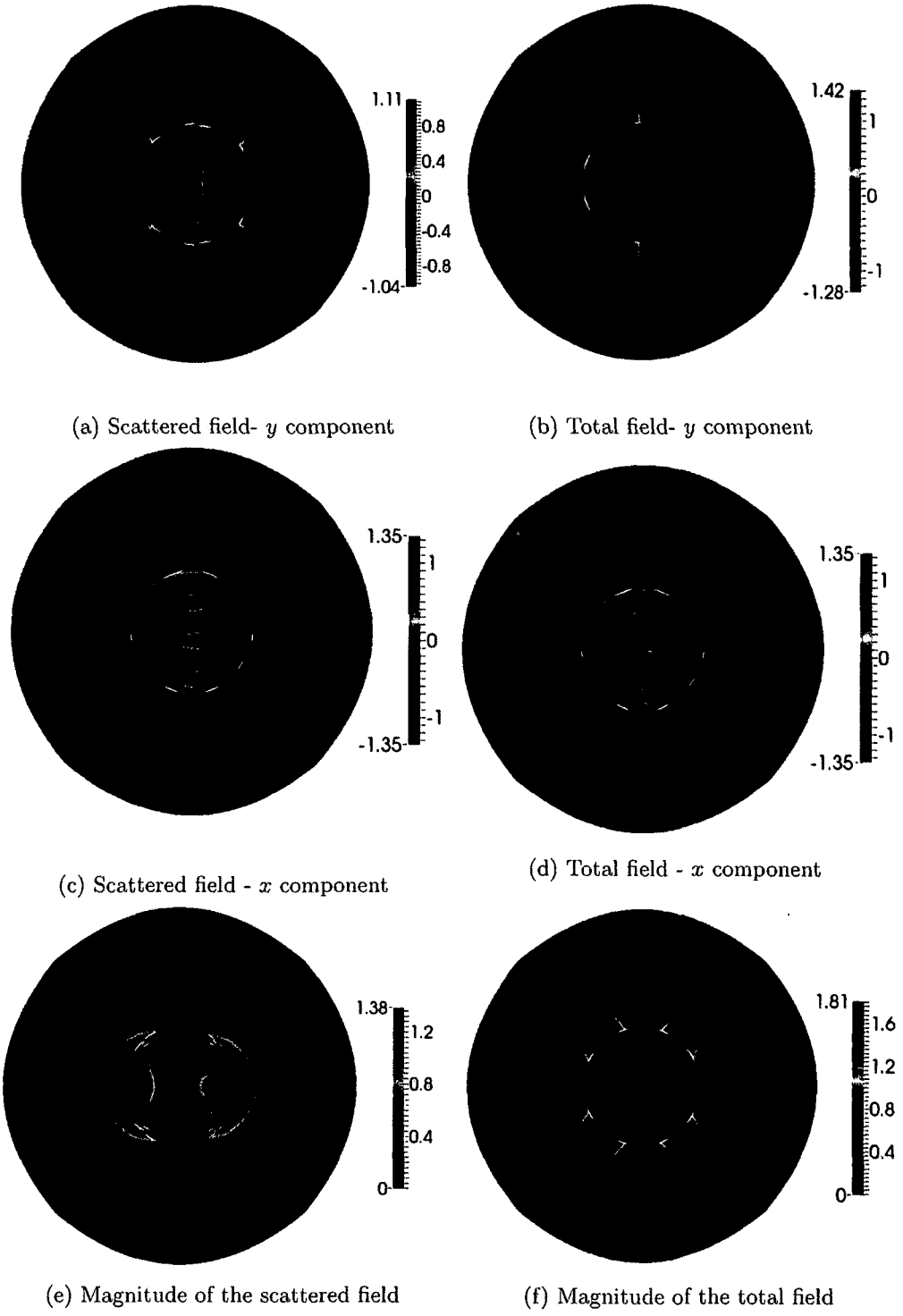
(d) Total field - x component



(e) Magnitude of the scattered field

(f) Magnitude of the total field

FIGURE. 5.16: Scattering of the lossless circular brain model - real part



(a) Scattered field- y component

(b) Total field- y component

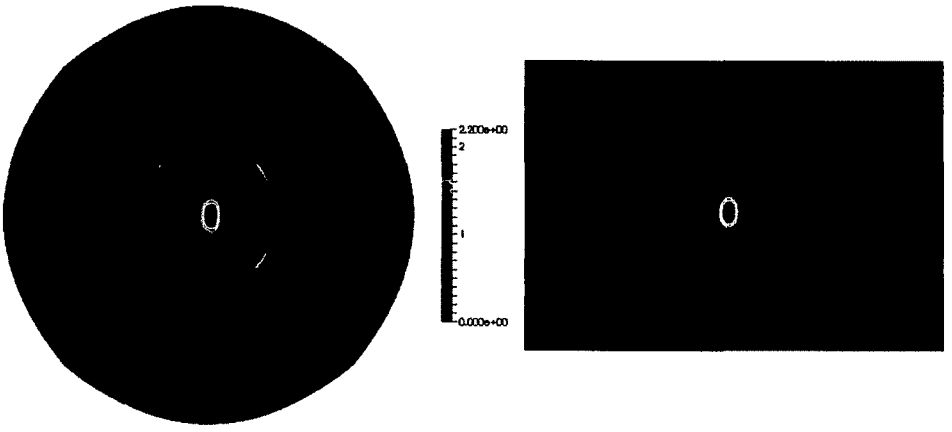
(c) Scattered field - x component

(d) Total field - x component

(e) Magnitude of the scattered field

(f) Magnitude of the total field

FIGURE. 5.17: Scattering of the lossless circular brain model - imaginary part



(a) $Abs(E_{scat})$ - Computed solution

(b) $Abs(E_{scat})$ - Analytical solution

FIGURE. 5.18: Circular lossless brain model - Comparison of the real part of the computed $Abs(E_{scat})$ with the analytical solution

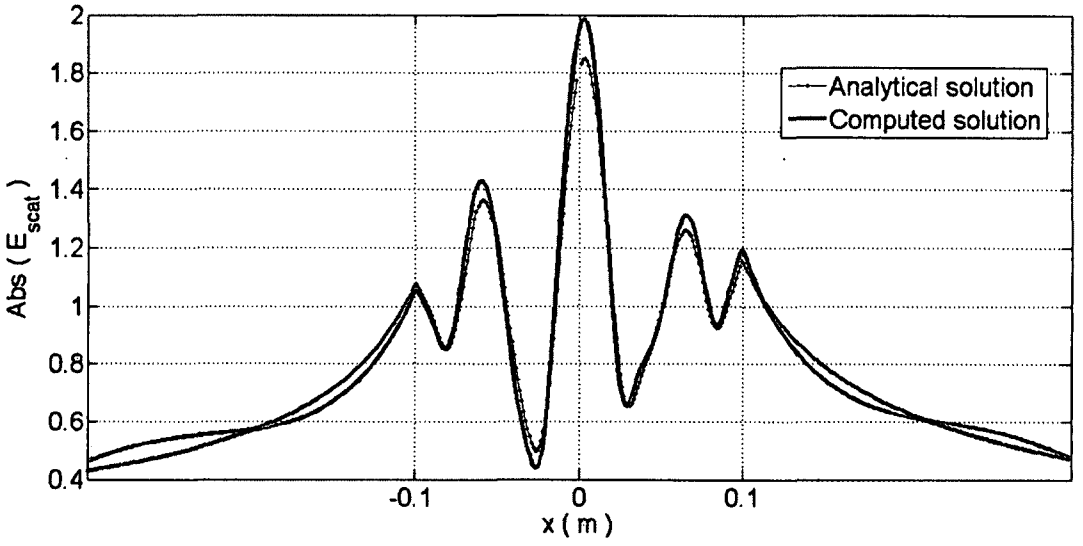
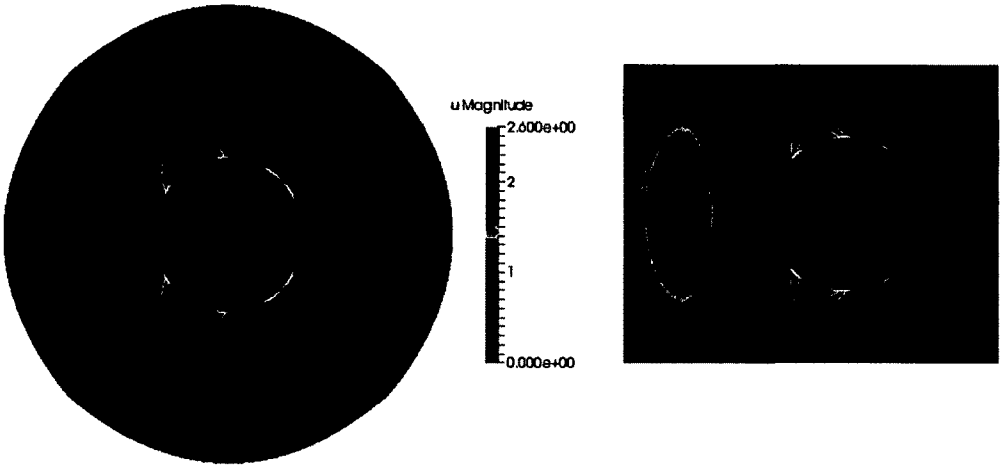


FIGURE. 5.19: Circular lossless brain model - Comparison of the real part of the computed $Abs(E_{scat})$ with the analytical solution along x axis



(a) $Abs(E_{scat})$ - Computed solution

(b) $Abs(E_{scat})$ - Analytical solution.

FIGURE. 5.20: Circular lossless brain model - Comparison the computed $Abs(E_{total})$ with analytical solution.

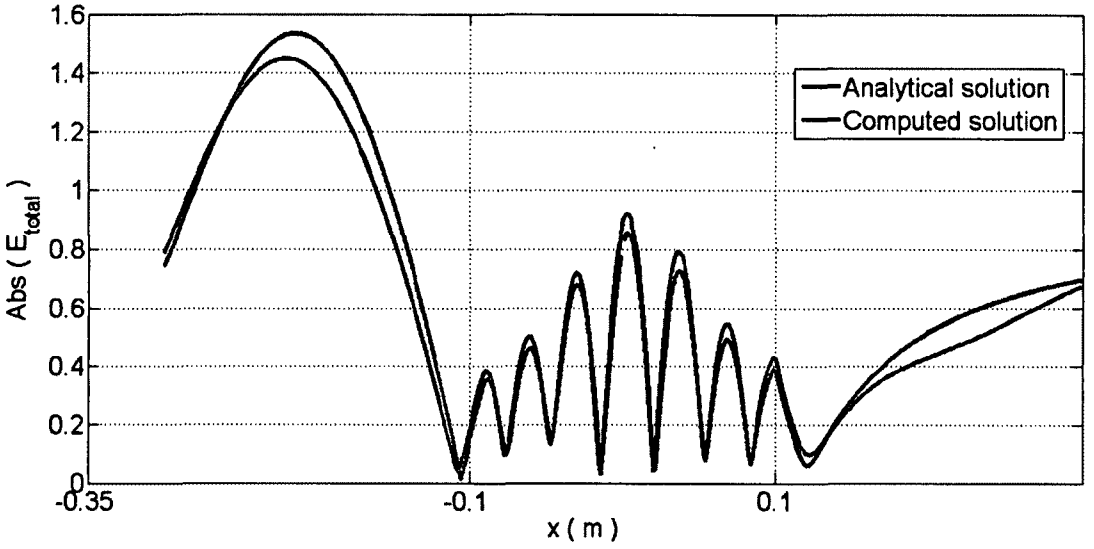


FIGURE. 5.21: Circular lossless brain model - Comparison the computed $Abs(E_{total})$ with analytical solution along x axis

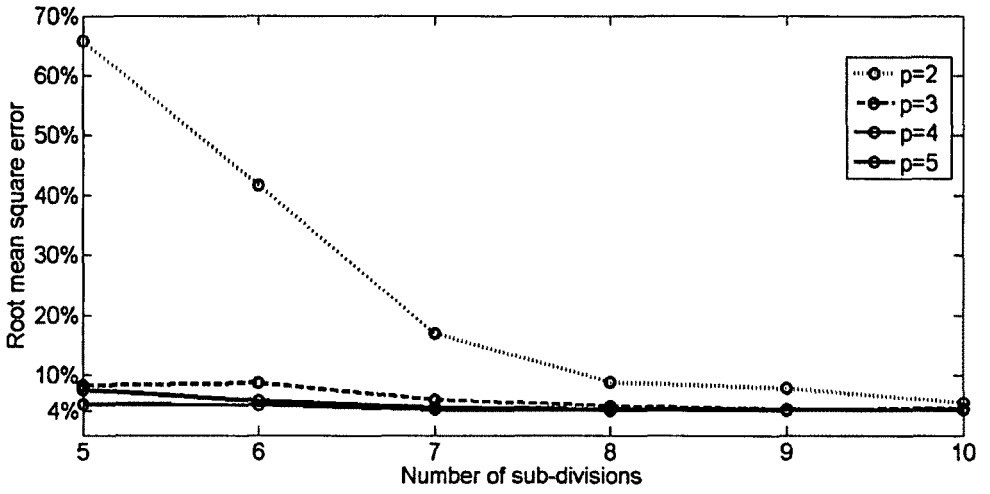
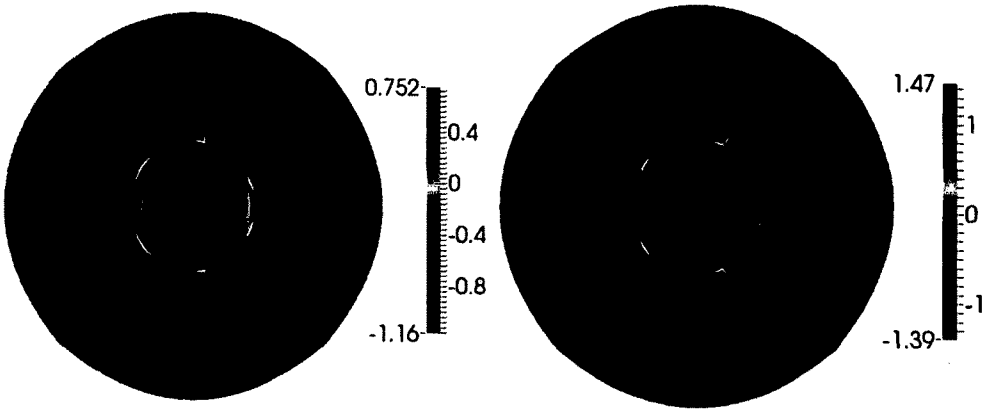


FIGURE. 5.22: Lossless circular brain model - root mean square error along x axis

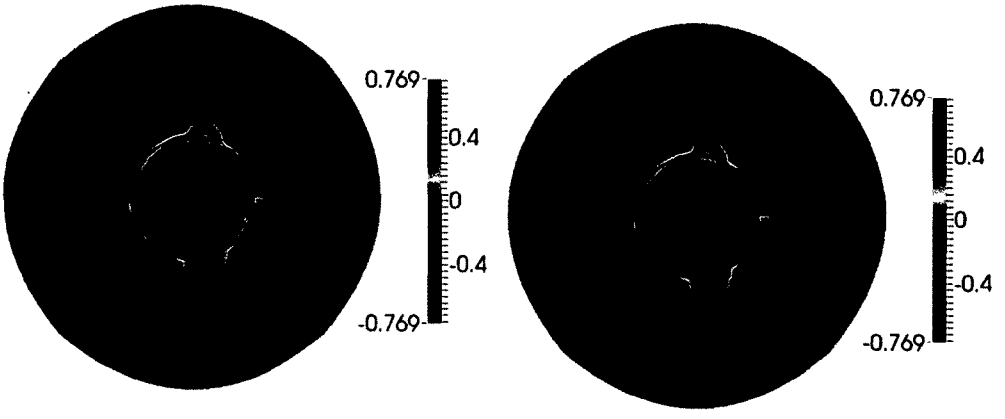
The solution is not acceptable for the order $p = 2$ with the initial discretization. However, the error is reduced the to $\sim 5\%$ by increasing the order to $p = 4$ without increasing the number of sub-divisions. As shown in Fig.5.22, further refinement reduced the error to $\sim 4\%$. Note that, the reported error is calculated for the solution in the scatterer as well as the free space. The error in the free space is considerably larger than the error inside the circular brain model.

The computed scattered and total fields considering realistic lossy properties. Figure 5.25, and Fig.5.26 are comparing the computed $Abs(E_{scat})$ and the $Abs(E_{total})$ along x axis with the analytical solution inside the brain. The comparison of Fig.5.25 and Fig.5.19 shows the effect of considering lossy material in analysis. The effects of including realistic material properties on the total electerical field is also evident when comparing Fig.5.26 with Fig.5.21.



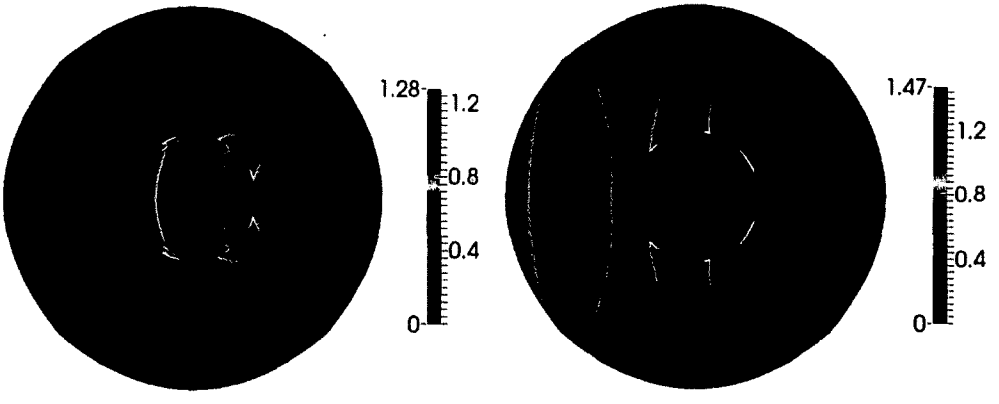
(a) Scattered field along y.

(b) Total field along y



(c) Scattered field along x

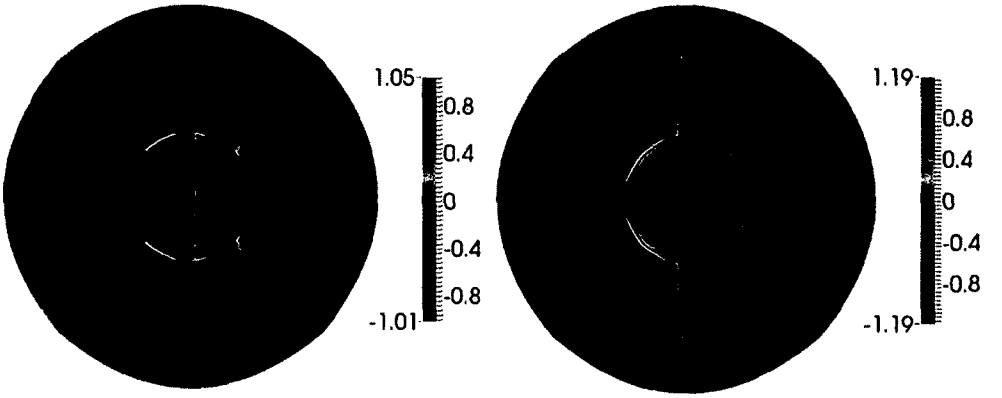
(d) Total field along x



(e) Magnitude of the scattered field

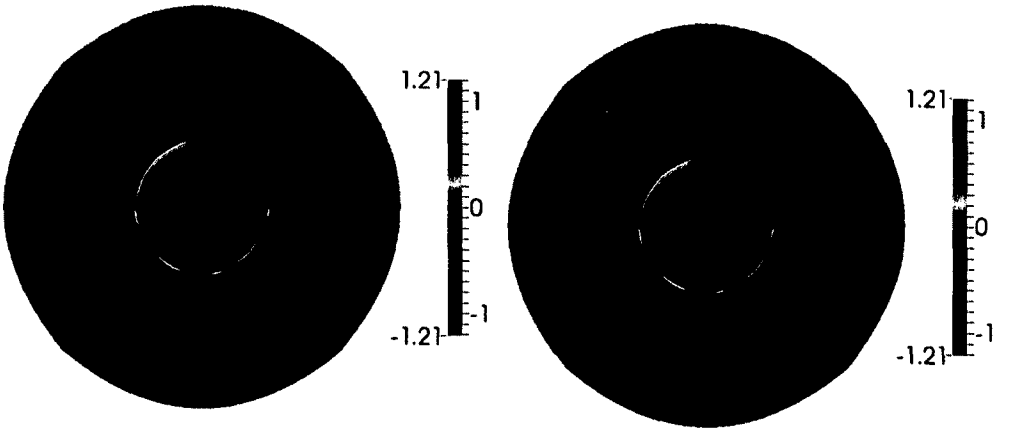
(f) Magnitude of the total field

FIGURE. 5.23: Scattering of the circular brain model with realistic material properties - real part



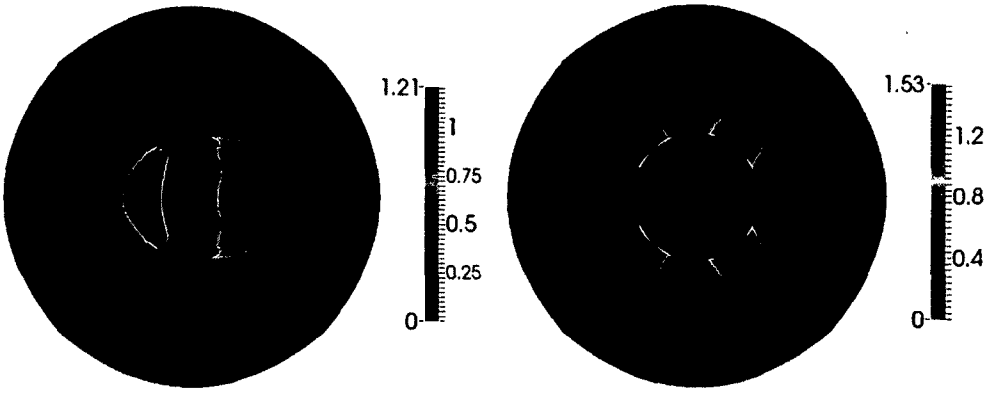
(a) Scattered field along y

(b) Total field along y



(c) Scattered field along x

(d) Total field along x



(e) Magnitude of the scattered field

(f) Magnitude of the total field

FIGURE. 5.24: Scattering of the circular brain model with realistic material properties - imaginary part

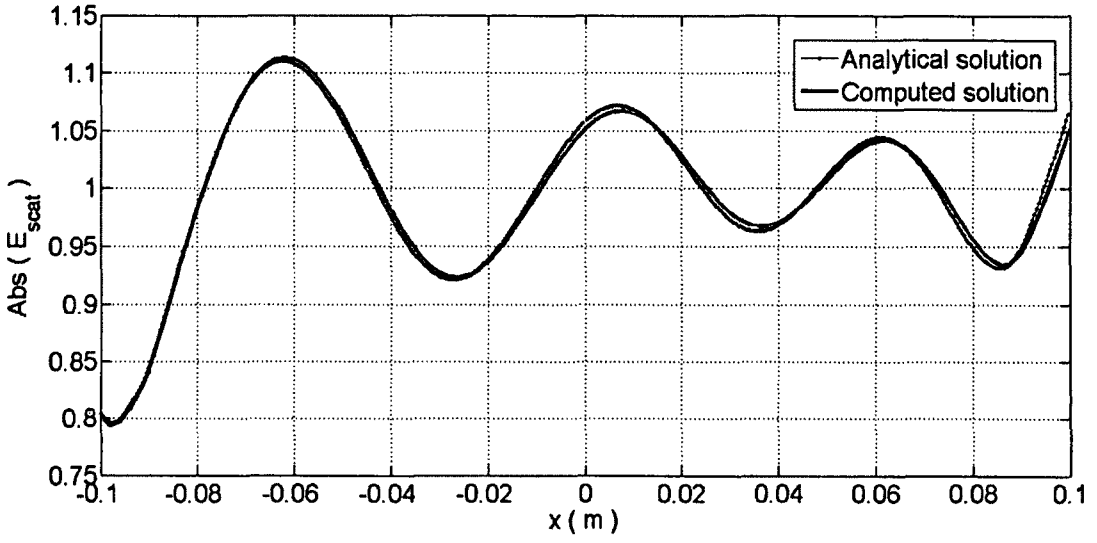


FIGURE. 5.25: Circular lossy brain model - Comparison the computed $Abs(E_{scat})$ with analytical solution along x axis

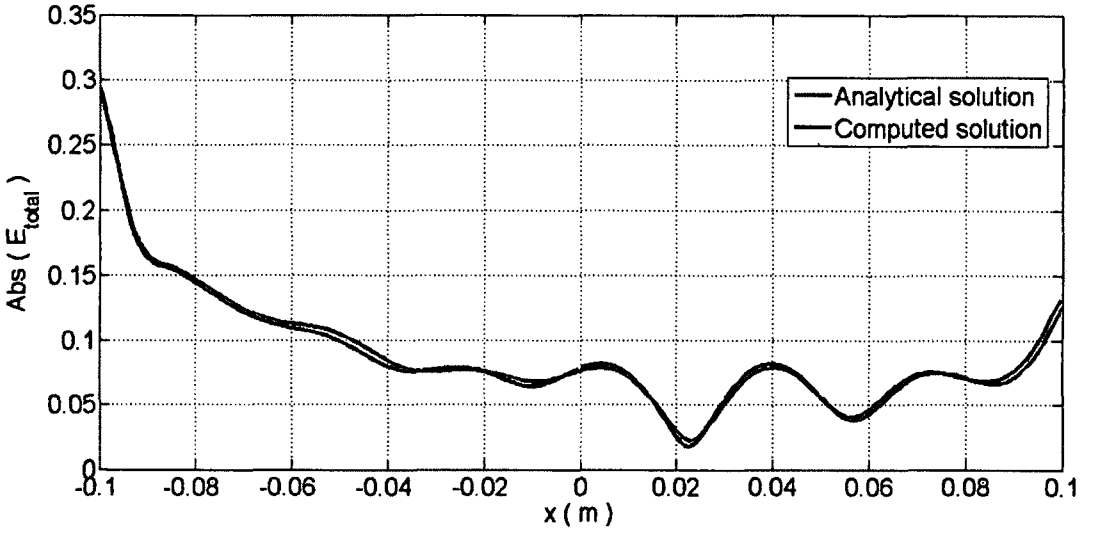


FIGURE. 5.26: Circular lossy brain model - Comparison the computed $Abs(E_{total})$ with analytical solution along x axis

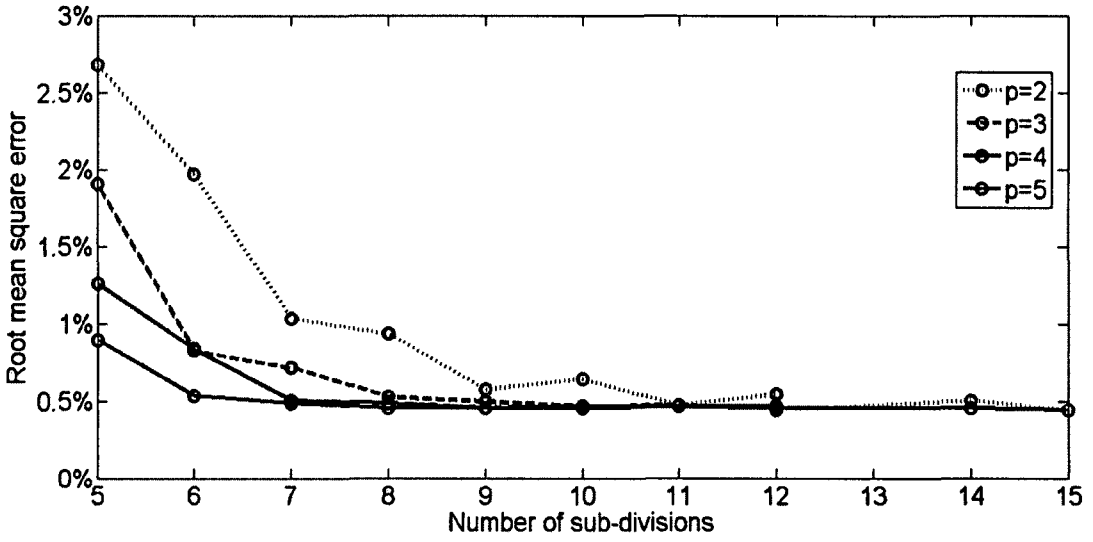


FIGURE. 5.27: Lossy circular brain model - root mean square error along x axis inside the brain

Since the focus of this study is estimating the electrical field inside the brain, the accuracy of the numerical solution inside the scatterer has particular importance. The root mean square error was calculated along the x axis inside the circular brain model with lossy material properties. The effects of refinement in reducing the error is depicted in Fig.5.27. The root mean square error along the x axis ranges from $\sim$ 0.8 - 2.7% for the orders of $p = 5$ and $p = 2$ and five sub-divisions. However, this error is decreased to $\sim$ 0.5% after increasing the number of sub-divisions as shown in Fig.5.27. The magnitude of the solution error is considerably smaller inside the brain.

5.7 ELECTROMAGNETIC RADIATION ON HUMAN BRAIN

In this section the numerical results of electromagnetic scattering is obtained on the brain section model. The incident wave is a monochromatic plane wave as

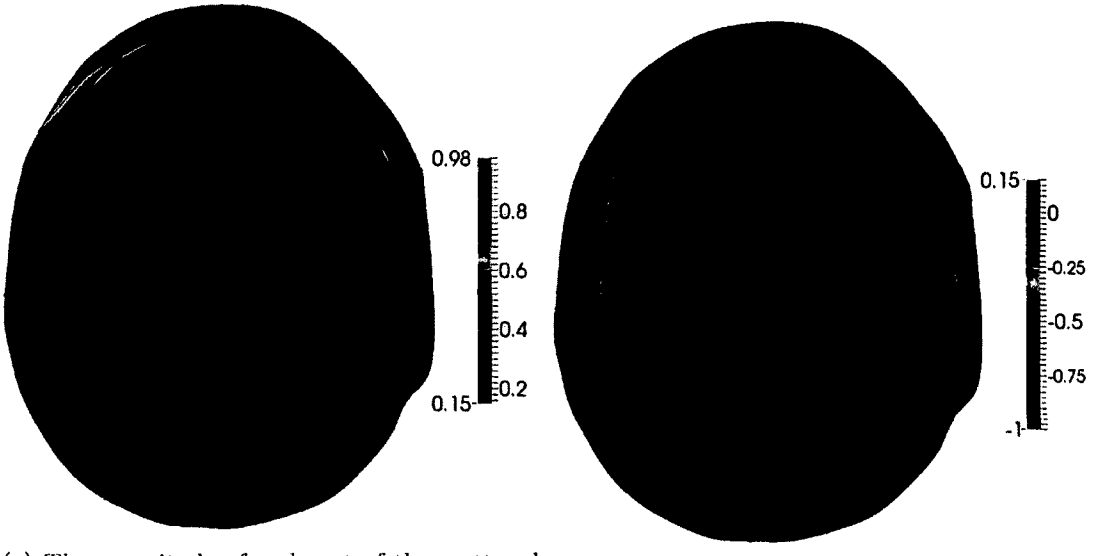
described by Eq.5.3.1. The brain section is composed of four different materials: the skull, the CSF, the grey matter, and the white matter. The frequency of the incident wave is 750 MHz. The material properties for this frequency are tabulated in Table 3 [63].

TABLE 3: Material properties of the brain section

Layer	ε_r	σ	μ_r
White matter	39.461304	0.545943	1.0
Grey Matter	53.556828	0.879847	1.0
CSF	69.005669	2.355479	1.0
Skull	16.872126	0.215958	1.0

The thickness of the surrounding patches of the brain section shown in Fig.4.7 were extended to λ, the wave length in the free space. The model was refined to create compatible patches as described in Chapter 4. No extra refinement is performed on the brain section model. The magnitude of the real and the imaginary part of the scattered field are depicted in Figure 5.28a and Fig.5.29a respectively. The real and imaginary parts of the scattered field along y direction are shown in Fig.5.28b and Fig.5.29b respectively.

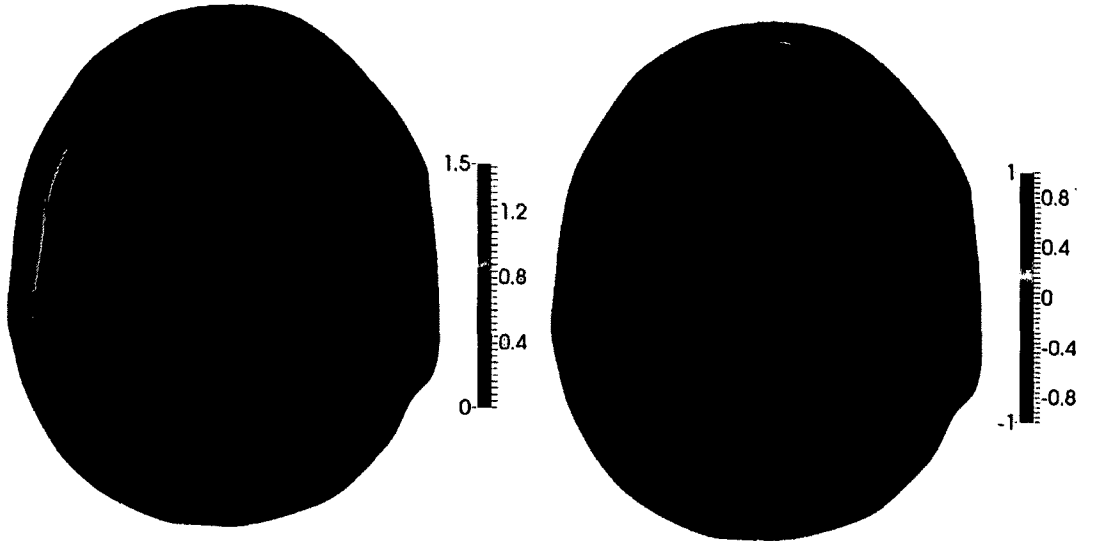
The magnitudes of the scattered and the total fields on the brain section are shown in Fig.5.30a and Fig.5.30 respectively. Moreover Fig.5.30 shows the difficulties of deep brain radiation as most of the electromagnetic wave energy is attenuated at the outer brain layers. To better visualize the scattered and total electrical fields inside the brain, the magnitudes of the scattered and the total fields for the white matter of the brain is depicted in Fig. 5.30a and Fig.5.31b respectively.



(a) The magnitude of real part of the scattered field

(b) Real part of the scattered field along y

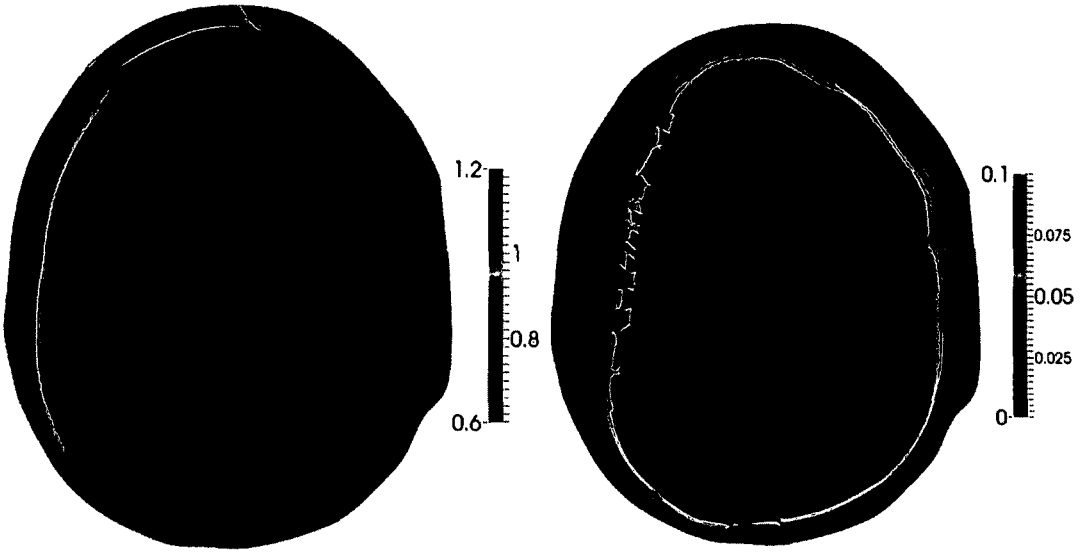
FIGURE. 5.28: Real part of the scattered field in the brain section with realistic material properties



(a) The magnitude of imaginary part of the scattered field

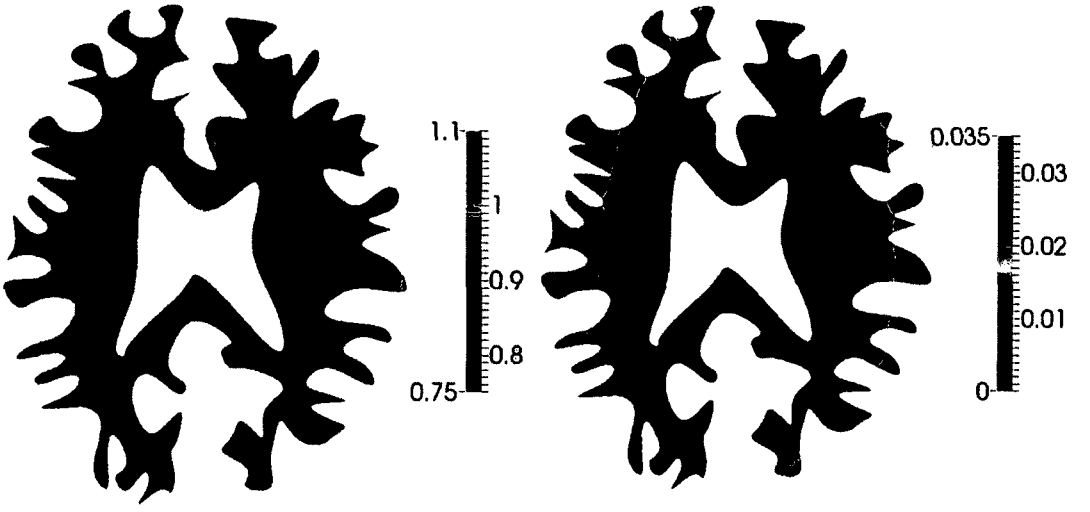
(b) Imaginary part of the total field along y

FIGURE. 5.29: Imaginary part of the scattered field in the brain section with realistic material properties



(a) $Abs(E_{scat})$ (b) $Abs(E_{total})$

FIGURE. 5.30: The magnitudes of the scattered and total field in the brain section



(a) $Abs(E_{scat})$ (b) $Abs(E_{total})$

FIGURE. 5.31: The magnitudes of the scattered and total fields in the white matter of the brain

CHAPTER 6

DISCUSSION AND CONCLUSION

The purpose of this study was to develop a framework for computational electromagnetics in IGA context. Specifically, the focus was on generating an imaged-based analysis model from geometrical data and estimating the electrical field within the resulted hi-fidelity mode. The developed method can be used to evaluate the safety of exposure to radiating devices used in cell-phones, LANs, and medical imaging devices. Another application of the developed framework is designing/optimizing antennas used for electromagnetic radiation. The summery of the present work, its limitations, and suggestions for future work are briefly discussed in this chapter.

6.1 RESULTS

Taking the advantage of the CAD modeling techniques, the IGA method can be used to produce hi-fidelity mesh models for analysis. The B-spline global curve interpretation and the Coons patches are used to generate analysis ready two dimensional geometries. The particular example problems studied here include the steady and transient Maxwell equations as well as electromagnetic scattering.

The IGA codes generated in this study were first validated by solving a steady and a time-dependent Maxwell equation defined over a square domain with Dirichlet boundary conditions. It is shown that the accuracy of the solution can be increased conveniently by superior refinement possibilities offered by IGA.

The developed IGA platform was then extended to solve the electromagnetic scattering problem. The FEM construction of the Perfectly Matched Layer (PML), which is required to mimic the infinite domain, was extended and incorporated in the developed IGA framework to produce an absorbing layer without wave reflection. Three examples were studied here for validation. In the first example a circular scatterer subjected to a plane incident wave is modeled with lossless material properties. The numerical solutions for the scattered and the total fields are obtained and validated against analytical solutions.

Then, a circular brain section with four layers subjected to a given incident wave was studied for both cases; with or without lossy materials. Again, the numerical solutions were compared with those obtained by analytical solutions. The same codes were then extended to the human brain section generated from the MRI image. The brain section and its surrounding was modeled with 399 patches. A four layer spherical model was studied in [41]. However the numerical error was not quantified. In both studies increasing the order of basis functions proved to be very effective in increasing the accuracy of the model as well as the convergence rate.

The IGA and its related theories and algorithms have been developed and validated here for two dimensional electromagnetic problems in this study. It will be an implementation issue to extend the developed IGA code to three dimensional problems. The IGA provided an effective and unified way to perform the analysis and to achieve the desired accuracy over an hi-fidelity model generated from geometrical data. Thus, the developed IGA package can be easily applied to solve

biomechanical problems with complicated geometries. Possible future application is the cancer treatment using the electromagnetic field generated by the specially designed antenna. Moreover, the developed IGA tool can help to optimize the shape of the antenna to serve the specific need of the patient.

6.2 LIMITATIONS

In IGA, The meshing within each patch is obtained automatically. However, complex geometries can not be meshed with a single patch. In multi-patch IGA it is necessary to define each patch manually which reduces the effectiveness of this method.The material properties considered here were all time-independent and the corresponding equations considered were all linear.

Reducing the error of the electromagnetic scattering problem can be done through PML tunning or optimization which is not performed in this study. The refinement performed on the four layer brain model with realistic materials reduced the solution error along x axis to less than 0.5% inside the circular brain model. However, further refinement did not reduce the error. This might be related to the PML layer definition or the accuracy of the numerical evaluation of the analytical solution.

The conforming mesh requirement considered in this study resulted in increased number of the meshes. Considering non-conforming patches in IGA may also be computationally expensive because of the added variables to the problem. A comparative study should be performed to select the best method for a particular

application.

6.3 FUTURE WORK

The models generated in this study were two-dimensional models. Expanding the current capability of imaged based 2D model generation to a 3D one is necessary for obtaining reliable results suitable for biomedical applications. To generate image-based 3D models additional algorithms should be added to the current codes.

The possibility of representing complex geometries in IGA combined with automatic meshing makes it the ideal framework for shape optimization. The combination of shape optimization with analysis in a single package has numerous applications in aerospace, defense, and biomedical fields. particularly, by modeling the shape of the antenna in the analysis domain one an evaluate/optimize the antenna design.

The study of electromagnetic scattering presented in this study represents an initial attempt to solve the electromagnetic radiation problem in biological media. Since electromagnetic radiation generates heat inside the biological tissue, it is desirable to couple bio-heat transfer equations with the electromagnetic equations to fully understand the impact of radiation on patients. Such a combination, for example, can be used to design the radiation devices required for hyperthermia treatment.

BIBLIOGRAPHY

[1] S. Adjerid, M. Aiffa, and J.E. Flaherty. Hierarchical finite element bases for triangular and tetrahedral elements. *Computer Methods in Applied Mechanics and Engineering*, 190, 1968.

[2] F. Auricchio, F. Calabr, T.J.R. Hughes, A. Reali, and G. Sangalli. A simple algorithm for obtaining nearly optimal quadrature rules for NURBS-based isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering*, 249252:15 – 27, 2012.

[3] I. Babuska, B. A Szabo, and I. N. Katz. The p-version of the finite element method. *SIAM Journal on Numerical Analysis*, 1981.

[4] Y. Bazilevs, V.M. Calo, J.A. Cottrell, J.A. Evans, T.J.R. Hughes, S. Lipton, M.A. Scott, and T.W. Sederberg. Isogeometric analysis using T-splines. *Computer Methods in Applied Mechanics and Engineering*, 2010.

[5] L. Beilina and M.J. Grote. Adaptive hybrid finite element/difference method for maxwells equations. *Technical Report September, University of Basel, Switzerland*, 2004.

[6] T. Belytschko, T. Rabczuk, A. Huerta, and S. Fernndez-Méndez. *Encyclopedia of Computational Mechanics*. Wiley.

[7] J.P. Berenger. A perfectly matched layer for the absorption of electromagnetic waves. *Journal of Computational Physics*, 114(2):185 – 200, 1994.

[8] M.J. Berger and R.J. LeVeque. Adaptive mesh refinement using wave-propagation algorithms for hyperbolic systems. *SIAM J. Numer. Anal.*, 35(6):2298–2316, October 1998.

[9] C.F. Bohren and D.R. Huffman. *Absorption and scattering of light by small particles.* New York: Wiley-Interscience.

[10] P.B. Bornemann and F. Cirak. A subdivision-based implementation of the hierarchical b-spline finite element method. *Computer Methods in Applied Mechanics and Engineering*, 253:584 – 598, 2013.

[11] A. Buffa, D. Cho, and G. Sangalli. Linear independence of the T-spline blending functions associated with some particular T-meshes. *Computer Methods in Applied Mechanics and Engineering*, 199(2324):1437 – 1445, 2010.

[12] A. Buffa, G. Sangalli, and R. Vázquez. Isogeometric analysis in electromagnetics: B-splines approximation. *Computer Methods in Applied Mechanics and Engineering*, 199(1720):1143 – 1152, 2010.

[13] Carl de Boor. Quasi interpolants and approximation power of multivariate splines. In: W. Dahmen, M. Gasca, C.A. Micchelli. (Eds.). *Computation of Curves and Surfaces.*

[14] P. Carnevali, R. B. Morris, Y. Tsuji, and G. Taylor. New basis functions and computational procedures for p-version finite element analysis. *International Journal for Numerical Methods in Engineering*, 36(22):3759–3779, 1993.

[15] W. Cecot, W. Rachowicz, and L. Demkowicz. An hp-adaptive finite element method for electromagnetics. part 3: A three-dimensional infinite element for maxwell's equations. *International Journal for Numerical Methods in Engineering*, 57, June. 2003.

[16] J. S. Chen, C. T. Wu, S Yoon, and Y. You. A stabilized conforming nodal integration for galerkin mesh-free methods. *International Journal for Numerical Methods in ETWMS Journal of Pure and Applied Mathematicsngineering*, 50, 1999.

[17] R.J. Cheng and K.M. Liew. A meshless analysis of three-dimensional transient heat conduction problems. *Engineering Analysis with Boundary Elements*, 36, 2012.

[18] W.C. Chew and W.H. Weedon. A 3D Perfectly Matched Medium from modified Maxwell's equations with stretched coordinates. *Microwave and Optical Technology Letters*, 7(13):599–604, 1994.

[19] S. A. Coons. Surfaces for Computer-Aided Design of Space Forms. Technical report, Cambridge, MA, USA, 1967.

[20] P. Costantini, C. Manni, F. Pelosi, and M. L. Sampoli. Quasi-interpolation in Isogeometric Analysis Based on Generalized B-splines. *Comput. Aided Geom. Des.*, 27(8):656–668, November 2010.

[21] J.A. Cottrell, T.J. R. Hughes, and Y. Bazilevs. *Isogeometric Analysis: Toward Integration of CAD and FEA*. Wiley Publishing, 1st edition, 2009.

[22] L. Cueto-Felgueroso, I. Colominas, G. Mosqueira, F. Navarrina, and M. Casteleiro. On the galerkin formulation of the smoothed particle hydrodynamics method. *International Journal for Numerical Methods in Engineering*, 60(9):1475–1512, 2004.

[23] C. de Falco, A. Reali, and R. Vázquez. GeoPDEs: A research tool for isogeometric analysis of PDEs. *Advances in Engineering Software*, 42(12):1020 – 1034, 2011.

[24] J. Deng, F. Chen, X. Li, C. Hu, W. Tong, Z. Yang, and Y. Feng. Polynomial splines over hierarchical T-meshes. *Graphical Models*, 70(4):76 – 86, 2008.

[25] T. Dokken and V. Skytt. Locally refined splines. In *proceeding of IV European Conference On Computational Mechanics. Solids, Structures and Coupled Problems in Engineering.*

[26] J. Dolbow and T. Belytschko. Numerical integration of the Galerkin weak form in meshfree methods. *Computational Mechanics*, 23(3):219–230, 1999.

[27] M.R. Dörfel, B. Jüttler, and B. Simeon. Adaptive isogeometric analysis by local h-refinement with T-splines. *Computer Methods in Applied Mechanics and Engineering*, 199(58):264 – 275, 2010. Computational Geometry and Analysis.

[28] A. Düster and E. Rank. The p-version of the finite element method compared to an adaptive h-version for the deformation theory of plasticity. *Computer Methods in Applied Mechanics and Engineering*, 190(1517):1925 – 1935, 2001.

[29] B. Engquist and A. Majda. Absorbing boundary conditions for the numerical simulation of waves. *Mathematics of Computation*, 31(139):pp. 629–651, 1977.

[30] G. Farin. *Curves and Surfaces for CAGD: A Practical Guide*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 5th edition, 2002.

[31] G. Gouesbet, J.A. Lock, and G. Gréhan. Generalized Lorenz-Mie theories and description of electromagnetic arbitrary shaped beams: Localized approximations and localized beam models, a review. *Journal of Quantitative Spectroscopy and Radiative Transfer*, 112(1):1 – 27, 2011.

[32] S. Groppa, A. Oliviero, A. Eisen, A. Quartarone, L.G. Cohen, V. Mall, A. Kaelin-Lang, T. Mima, S. Rossi, G.W. Thickbroom, P.M. Rossini, U. Ziemann, J. Valls-Sol, and H.R. Siebner. A practical guide to diagnostic transcranial magnetic stimulation: Report of an IFCN committee. *Clinical Neurophysiology*, 123(5):858 – 882, 2012.

[33] M.J. Grote and J.B. Keller. Nonreflecting Boundary Conditions for Maxwell's Equations. *Journal of Computational Physics*, 139(2):327 – 342, 1998.

[34] R.S. Gurjar, V. Backman, L.T. Perelman, I. Georgakoudi, K. Badizadegan, I. Itzkan, and et. al. Imaging human epithelial properties with polarized light-scattering spectroscopy. *Nat Med*, 2001.

[35] A.W. Guy. Electromagnetic Fields and Relative Heating Patterns Due to a Rectangular Aperture Source in Direct Contact with Bilayered Biological Tissue. *Microwave Theory and Techniques, IEEE Transactions*, 19(2):214–223,

February 1971.

[36] T.J.R. Hughes. *The finite element method : linear static and dynamic finite element analysis.* Englewood Cliffs, N.J. Prentice-Hall International, 1987.

[37] T.J.R. Hughes, J.A. Cottrell, and Y. Bazilevs. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. *Computer Methods in Applied Mechanics and Engineering*, 194(3941):4135 – 4195, 2005.

[38] T.J.R. Hughes, A. Reali, and G. Sangalli. Efficient quadrature for nurbs-based isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering*, 199, 2010.

[39] G. J. Johnson. Notes on Perfectly Matched Layers (PMLs). *MIT courses 18.369 and 18.336*, March 2010.

[40] M. Kerker. The scattering of light and other electromagnetic radiation. *New York: Academic Press*, 1969.

[41] K. Kim. *Finite Element Modeling of Electromagnetic Radiation and Induced Heat Transfer in the Human Body.* PhD thesis, The University of Texas at Austin, 2013.

[42] C. Kottke, A. Farjadpour, and S.G. Johnson. Perturbation theory for anisotropic dielectric interfaces, and application to subpixel smoothing of discretized numerical methods. *Phys. Rev. E*, 77:036611, Mar 2008.

[43] B.G Lee, T. Lyche, and T. Morken. Some Examples of Quasi-Interpolants Constructed from Local Spline Projectors. In *Mathematical Methods in CAGD: Oslo 2000, Vanderbilt*, pages 243–252. University Press, 2000.

[44] E.T.Y. Lee. Choosing nodes in parametric curve interpolation. *Computer-Aided Design*, 21(6):363 – 370, 1989.

[45] J.P. Lefaucheur et.al. Evidence-based guidelines on the therapeutic use of repetitive Transcranial Magnetic Stimulation (rTMS). *Clinical Neurophysiology*, 125(11):2150 – 2206, 2014.

[46] Li Liu, M.I Mishchenko, and W.P. Arnott. A study of radiative properties of fractal soot aggregates using the superposition T-matrix method. *Journal of Quantitative Spectroscopy and Radiative Transfer*, 109(15):2656 – 2663, 2008.

[47] T. Lyche, C. Manni, and P. Sablonniere. Quasi-interpolation projectors for box splines. *Journal of Computational and Applied Mathematics*, 221(2):416 – 429, 2008.

[48] T. Lyche and L.L. Schumaker. Local spline approximation methods. *Journal of Approximation Theory*, 15(4):294 – 325, 1975.

[49] S.G. Mark and M.P. Robert. Daily left Prefrontal Repetitive Transcranial Magnetic Stimulation for Acute Treatment of Medication-Resistant Depression. *American Journal of Psychiatry*, 168(4):356–364, 2011. PMID: 21474597.

[50] P.J. Matuszyk and L.F. Demkowicz. Parametric finite elements, exact sequences and perfectly matched layers. *Computational Mechanics*, 51(1):35–45, 2013.

[51] Mätzle. Matlab functions for mie scattering and absorption, version 2. Technical report, Institut für angewandte Physik, Universität Bern, 2002.

[52] J.M. Melenk and I. Babuska. The partition of unity finite element method: Basic theory and applications. *Computer Methods in Applied Mechanics and Engineering*, 139(14):289 – 314, 1996.

[53] G. Mie. Beiträge zur optik trüber medien, speziell kolloidaler metallösungen. *Ann Phys*, 1908.

[54] P. Monk. Analysis of a Finite Element Method for Maxwell's Equations. *SIAM Journal on Numerical Analysis*, 29(3):pp. 714–729, 1992.

[55] J.C. Nedelec. Mixed finite elements in *R*3. *Numerische Mathematik*, 35(3):315–341, 1980.

[56] J.C. Nedelec. A new family of mixed finite elements in *R*3. *Numerische Mathematik*, 50(1):57–81, 1986.

[57] V.P. Nguyen, T. Rabczuk, S. Bordas, and M. Duflot. Meshless methods: A review and computer implementation aspects. *Mathematics and Computers in Simulation*, 79(3):763 – 813, 2008.

[58] N. Nguyen-Thanh, J. Kiendl, H. Nguyen-Xuan, R. Wchner, K.U. Bletzinger, Y. Bazilevs, and T. Rabczuk. Rotation free isogeometric thin shell analysis

using PHT-splines. *Computer Methods in Applied Mechanics and Engineering*, 200(4748):3410 – 3424, 2011.

[59] N. Nguyen-Thanh, H. Nguyen-Xuan, S.P.A. Bordas, and T. Rabczuk. Isogeometric analysis using polynomial splines over hierarchical T-meshes for two-dimensional elastic solids. *Computer Methods in Applied Mechanics and Engineering*, 200(2122):1892 – 1908, 2011.

[60] M. Okoniewski, E. Okoniewska, and M.A. Stuchly. Three-dimensional subgridding algorithm for FDTD. *Antennas and Propagation, IEEE Transactions on*, 1997.

[61] L. Piegl and W. Tiller. *The NURBS book*. Springer-Verlag, London, UK, 1995.

[62] X. Qian and O. Sigmund. Isogeometric shape optimization of photonic crystals via Coons patches. *Computer Methods in Applied Mechanics and Engineering*, 200(25-28):2237–2255, 2011.

[63] Radio Communication Commission - Radio frequency safety. http://transition.fcc.gov/oet/rfsafety/dielectric.html.

[64] C.M. Rappaport. Perfectly matched absorbing boundary conditions based on anisotropic lossy mapping of space. *Microwave and Guided Wave Letters, IEEE*, 5(3):90–92, Mar 1995.

[65] J. Schäfer. *Implementierung und Anwendung analytischer und numerischer Verfahren zur L¨osung der Maxwellgleichungen für die Untersuchung der Lichtausbreitung in biologischem Gewebe.* PhD thesis, Univerität Ulm, 2011.

[66] J. Schäfer, S.-C. Lee, and A. Kienle. Calculation of the near fields for the scattering of electromagnetic waves by multiple infinite cylinders at perpendicular incidence. *Journal of Quantitative Spectroscopy and Radiative Transfer*, 113(16):2113 – 2123, 2012.

[67] D. Schillinger, L. Dede', M.A. Scott, J.A. Evans, M.J. Borden, E. Rank, and T.J.R. Hughes. An Isogeometric design-through-analysis methodology based on adaptive hierarchical refinement of NURBS, immersed boundary methods, and T-spline CAD surfaces. *Computer Methods in Applied Mechanics and Engineering*, 249-252:116–150, 2012.

[68] H.P. Schwan and G.M. Piersol. The absorption of electromagnetic energy in body tissues; a review and critical analysis. *Am J Phys Med*, 34(3):425–448, 1955.

[69] M.A. Scott, X. Li, T.W. Sederberg, and T.J.R. Hughes. Local refinement of analysis-suitable t-splines. *Computer Methods in Applied Mechanics and Engineering*, 213216(0):206 – 222, 2012.

[70] M.A. Scott, R.N. Simpson, J.A. Evans, S. Lipton, S.P.A. Bordas, T.J.R. Hughes, and T.W. Sederberg. Isogeometric boundary element analysis using unstructured T-splines. *Computer Methods in Applied Mechanics and Engineering*,

254:197 – 221, 2013.

[71] T.W. Sederberg, J. Zheng, A. Bakenov, and A. Nasri. T-splines and T-NURCCs. *ACM Trans. Graph.*, 22(3):477–484, July 2003.

[72] D. Shepard. A Two-dimensional Interpolation Function for Irregularly-spaced Data. In *Proceedings of the 1968 23rd ACM National Conference*, ACM '68, pages 517–524, New York, NY, USA, 1968. ACM.

[73] P Silvester. Finite element solution of homogeneous waveguide problems. *Alta Frequenza*, 1969.

[74] R.N. Simpson, S.P.A. Bordas, J. Trevelyan, and T. Rabczuk. A two-dimensional isogeometric Boundary Element Method for elastostatic analysis. *Computer Methods in Applied Mechanics and Engineering*, 209-212:87–100, 2012.

[75] H. Speleers, C. Manni, F. Pelosi, and M.L. Sampoli. Isogeometric analysis with Powell-sabin splines for advection-diffusion-reaction problems. *Computer Methods in Applied Mechanics and Engineering*, 221-222:132 – 148, 2012.

[76] B.A. Szabo and I. Babuska. *Introduction to finite element analysis : formulation, verification and validation.* Wiley.

[77] F.L. Teixeira and W.C. Chew. General closed-form PML constitutive tensors to match arbitrary bianisotropic and dispersive linear media. *Microwave and Guided Wave Letters, IEEE*, 8(6):223–225, Jun 1998.

[78] G. Ventura. An augmented Lagrangian approach to essential boundary conditions in meshless methods. *International Journal for Numerical Methods in Engineering*, 53(4):825–842, 2002.

[79] A.V. Vuong, C. Giannelli, B. Jüttler, and B. Simeon. A hierarchical approach to adaptive local refinement in isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering*, 200(4952):3554 – 3567, 2011.

[80] A.V. Vuong, Ch. Heinrich, and B. Simeon. Isogat: A 2D tutorial MATLAB code for Isogeometric Analysis. *Computer Aided Geometric Design*, 27(8):644 – 655, 2010.

[81] D. Wang and J. Xuan. An improved NURBS-based isogeometric analysis with enhanced treatment of essential boundary conditions. *Computer Methods in Applied Mechanics and Engineering*, 199(3740):2425 – 2436, 2010.

[82] Y. Wang, Y. He, X. Li, X. Gu, and H. Qin. Computer-Aided Design. *International Journal for Numerical Methods in Engineering*, 2008.

[83] A. J. Ward and J. B. Pendry. Refraction and geometry in Maxwell's equations. *Journal of Modern Optics*, 43(4):773–793, 1996.

[84] G. Xu, E. Atroshchenko, and S. Bordas. Geometry independent field approximation for spline-based finite element methods. *Computer Methods in Applied Mechanics and Engineering*, Sep 2014.

[85] Y.L. Xu. Electromagnetic scattering by an aggregate of spheres. *Appl. Opt.*, 34(21):4573–4588, Jul 1995.

[86] K. Yee. Numerical solution of initial boundary value problems involving Maxwell's equations in isotropic media. *Antennas and Propagation, IEEE Transactions on*, 14(3):302–307, May 1966.

[87] A.R. Zakharian, M. Brio, and J.V. Moloney. FDTD based second-order accurate local mesh refinement method for Maxwells equations in two space dimensions. *Communications in Mathematical Sciences*, 2004.

APPENDIX A

DERIVATION OF THE WEAK FORM

The governing differential equation for electromagnetic field is given by Eq.3.1.11 as

$$\varepsilon \boldsymbol{E}_{,tt} + \sigma \boldsymbol{E}_{,t} + \nabla \times (1/\mu \nabla \times \boldsymbol{E}) = \boldsymbol{f} \qquad \text{in} \quad \Omega \times (0,t). \tag{A.0.1}$$

where $\boldsymbol{f} = -\boldsymbol{J}^{imp}_{,t}$. Its weak form can be derived using the following index notations to define the inner and the inner and the cross products will be written for vector functions $\boldsymbol{a}$ and $\boldsymbol{b}$ as:

$$\begin{aligned} \boldsymbol{a} \cdot \boldsymbol{b} &= a_i b_i, \\ \boldsymbol{a} \times \boldsymbol{b} &= \varepsilon_{ijk} a_j b_k. \end{aligned} \tag{A.0.2}$$

Note that ε_{ijk} is called permutation symbol, defined as follows: $\varepsilon_{ijk} = 0$ if any index is repeated.

$\varepsilon_{ijk} = 1$ if the indices follow the position permutation, 1, 2, 3.

$\varepsilon_{ijk} = -1$ if the index doesn't follow the position permutation, 1, 2, 3.

Furthermore, the general Gauss's Theorem is used:

$$\int_{\Omega} (T_{jk\ldots}) d\Omega = \int_{\partial\Omega} (T_{jk\ldots}) n_i d\Omega \tag{A.0.3}$$

where n_i is the i^{th} component of the outward unit vector normal to the surface, $\partial\Omega$,

of the domain Ω. Let $\boldsymbol{v}$ be arbitrary. One then obtains the following:

$$
\begin{aligned}
0 &= \int_\Omega \Big[\Big(\varepsilon\ddot{\boldsymbol{E}} + \sigma\dot{\boldsymbol{E}} + \nabla\times(\frac{1}{\mu}\nabla\times\boldsymbol{E}) - \boldsymbol{f}\Big)\cdot\boldsymbol{v}\Big]d\Omega \\
&= \int_\Omega \Big[\Big(\varepsilon\ddot{E}_p + \sigma\dot{E}_p + \varepsilon_{pqi}(\frac{1}{\mu}\varepsilon_{ijk}E_{k,j})_{,q} - f_p\Big)v_p\Big]d\Omega \\
&= \int_\Omega \Big[\Big(\varepsilon\ddot{E}_p + \sigma\dot{E}_p + \frac{1}{\mu}\varepsilon_{pqi}\varepsilon_{ijk}E_{k,jq} - f_p\Big)v_p\Big]d\Omega \\
&= \int_\Omega \Big[\Big(\varepsilon\ddot{E}_p v_p + \sigma\dot{E}_p v_p + \frac{1}{\mu}\varepsilon_{pqi}\varepsilon_{ijk}E_{k,j}v_{p,q} - f_p v_p\Big)\Big]d\Omega + \int_{\partial\Omega}\Big(\frac{1}{\mu}\varepsilon_{pqi}\varepsilon_{ijk}E_{k,j}v_p n_q\Big)dA \\
&= \int_\Omega \Big[\Big(\varepsilon\ddot{E}_p v_p + \sigma\dot{E}_p v_p - \frac{1}{\mu}(\varepsilon_{ijk}E_{k,j})(\varepsilon_{ipq}v_{p,q}) - f_p v_p\Big)\Big]d\Omega + \int_{\partial\Omega}\Big(\frac{1}{\mu}(\varepsilon_{ijk}(E_{k,j})\varepsilon_{ipq}v_p n_q)\Big)dA \\
&= \int_\Omega \Big[\Big(\varepsilon\ddot{E}_p v_p + \sigma\dot{E}_p v_p + \frac{1}{\mu}(\varepsilon_{ijk}E_{k,j})(\varepsilon_{iqp}v_{p,q}) - f_p v_p\Big)\Big]d\Omega + \int_{\partial\Omega}\Big(\frac{1}{\mu}(\varepsilon_{ijk}(E_{k,j})\varepsilon_{ipq}v_p n_q)\Big)dA \\
&= \int_\Omega \Big[\varepsilon\ddot{\boldsymbol{E}}\cdot\boldsymbol{v} + \sigma\dot{\boldsymbol{E}}\cdot\boldsymbol{v} - \frac{1}{\mu}(\nabla\times\boldsymbol{E})\cdot(\nabla\cdot\boldsymbol{v}) - \boldsymbol{f}\cdot\boldsymbol{v}\Big]d\Omega + \int_{\partial\Omega}\Big[\frac{1}{\mu}(\nabla\times\boldsymbol{E})\cdot(\boldsymbol{v}\times\boldsymbol{n})\Big]dA \\
&= \int_\Omega \Big[\varepsilon\ddot{\boldsymbol{E}}\cdot\boldsymbol{v} + \sigma\dot{\boldsymbol{E}}\cdot\boldsymbol{v} - \frac{1}{\mu}(\nabla\times\boldsymbol{E})\cdot(\nabla\cdot\boldsymbol{v}) - \boldsymbol{f}\cdot\boldsymbol{v}\Big]d\Omega + \int_{\Gamma_D}\Big[\frac{1}{\mu}(\nabla\times\boldsymbol{E})\cdot(\boldsymbol{v}\times\boldsymbol{n})\Big]d\Gamma_D \\
&+ \int_{\Gamma_N}\Big[\frac{1}{\mu}(\nabla\times\boldsymbol{E})\cdot(\boldsymbol{v}\times\boldsymbol{n})\Big]d\Gamma_N \\
&= \int_\Omega \Big[\varepsilon\ddot{\boldsymbol{E}}\cdot\boldsymbol{v} + \sigma\dot{\boldsymbol{E}}\cdot\boldsymbol{v} - \frac{1}{\mu}(\nabla\times\boldsymbol{E})\cdot(\nabla\cdot\boldsymbol{v}) - \boldsymbol{f}\cdot\boldsymbol{v}\Big]d\Omega + \int_{\Gamma_D}\Big[\frac{1}{\mu}(\nabla\times\boldsymbol{E})\cdot(\boldsymbol{v}\times\boldsymbol{n})\Big]d\Gamma_D \\
&- \int_{\Gamma_N}\Big[\frac{1}{\mu}(\nabla\times\boldsymbol{E})\times\boldsymbol{n}\Big]\cdot\boldsymbol{v}d\Gamma_N
\end{aligned}
\tag{A.0.4}
$$

The relation, $(\nabla\times\boldsymbol{E})\cdot(\boldsymbol{v}\times\boldsymbol{n}) = -[(\nabla\times\boldsymbol{E})\times\boldsymbol{n}]\cdot\boldsymbol{v}$, is used in the last step of the derivation. The above equation reveals several ways to prescribe the boundary conditions.

- Case I: homogeneous boundary conditions

  The required boundary conditions can be either the Neumann boundary conditions:

$$
(\nabla\times\boldsymbol{E}) = 0 \quad \text{on} \quad \Gamma_N, \tag{A.0.5}
$$

or the Dirichlet boundary condition:

$$\boldsymbol{E} \times \boldsymbol{n} = 0 \quad \text{on} \quad \Gamma_D. \tag{A.0.6}$$

Then, the variational equation becomes:

$$\int_\Omega \left[\varepsilon \ddot{\boldsymbol{E}} \cdot \boldsymbol{v} + \sigma \dot{\boldsymbol{E}} \cdot \boldsymbol{v} + \frac{1}{\mu}(\nabla \times \boldsymbol{E}) \cdot (\nabla \times \boldsymbol{v}) - \boldsymbol{f} \cdot \boldsymbol{v}\right] d\Omega = 0. \tag{A.0.7}$$

- Case II: non-homogeneous boundary conditions

The required boundary conditions can be either the Neumann boundary conditions:

$$(\nabla \times \boldsymbol{E}) = \boldsymbol{h}_1 \quad \text{on} \quad \Gamma_N, \tag{A.0.8}$$

or the or the Dirichlet boundary condition:

$$\boldsymbol{E} \times \boldsymbol{n} = \boldsymbol{g} \quad \text{on} \quad \Gamma_D. \tag{A.0.9}$$

Now, again set $\boldsymbol{v} \times \boldsymbol{n}$ on Γ_D as the requirement for the selection of the arbitrary function and $\boldsymbol{E} \times \boldsymbol{n} = \boldsymbol{g}$ on Γ_D. Now the variational equation becomes:

$$\int_\Omega \left[\varepsilon \ddot{\boldsymbol{E}} \cdot \boldsymbol{v} + \sigma \dot{\boldsymbol{E}} \cdot \boldsymbol{v} + \frac{1}{\mu}(\nabla \times \boldsymbol{E}) \cdot (\nabla \times \boldsymbol{v}) - \boldsymbol{f} \cdot \boldsymbol{v}\right] d\Omega + \int_{\Gamma_N} \frac{1}{\mu}[\boldsymbol{h}_1 \cdot (\boldsymbol{v} \times \boldsymbol{n})] d\Gamma_N = 0. \tag{A.0.10}$$

Also, with Neumann boundary condition stated as:

$$(\nabla \times \boldsymbol{E}) \times \boldsymbol{n} = \boldsymbol{h}_2 \quad \text{on} \quad \Gamma_N, \tag{A.0.11}$$

the variational equation can be re-casted as follows:

$$\int_\Omega \left[\varepsilon \ddot{\boldsymbol{E}} \cdot \boldsymbol{v} + \sigma \dot{\boldsymbol{E}} \cdot \boldsymbol{v} + \frac{1}{\mu}(\nabla \times \boldsymbol{E}) \cdot (\nabla \times \boldsymbol{v}) - \boldsymbol{f} \cdot \boldsymbol{v}\right] d\Omega - \int_{\Gamma_N} \frac{1}{\mu}[\boldsymbol{h}_2 \cdot \boldsymbol{v}] d\Gamma_N = 0. \tag{A.0.12}$$

The steady state and the time-harmonic form of Eq.3.1.13 can be extended from the above derivation.

VITA

Tahsin Khajah
Department of Department of Mechanical and Aerospace Enigneering
Old Dominion University
Norfolk, VA 23529

EDUCATION Old Dominion University, Norfolk, VA
Doctor of Philosophy in Mechanical Engineering, GPA: 4.0 May 2015
Dissertation topic: Isogeometric Analysis for Electromagnetism

Sharif University of Technology, Tehran, Iran
Master of Science in Mechanical Engineering March 2006
Thesis topic: Dynamic Response of an Axial Loaded Beam on Viscoelastic Foundation to Moving Mass

Razi University, Kermanshah, Iran
Bachelor of Science in Mechanical Engineering May 2001
Thesis topic: Study of Dynamic Vibration Absorbers and Designing a Steady Table using Dynamic Absorbers

EXPERIENCE
ACADEMIC EXPERIENCE
Old Dominion University, Norfolk, VA
Engineering Technology Department

- **Visiting Lecturer** August 2010- May 2012
  Courses taught:
  - Statics
  - Engineering Graphics/AutoCAD
  - Manufacturing Processes

Mechanical and Aerospace Engineering Department

- **Instructor** August 2012 - May 2015
  Courses taught:
  - Mechanical Design II
  - Statics recitation class

INDUSTRIAL EXPERIENCE

Creare Company, Hanover, NH

Off-site Mechanical Engineer Aug 2011 - June 2012

Participated in design and documentation of compact shipboard swaging machine

Payvand Golestan Cement Company (PGCC), Tehran, Iran

Technical Manager March 2009 - May 2010

- Prepared technical specifications for machinery
- Assisted in preparation of detailed work break down structure
- Provided installation schedule and cost estimates
- Conducted failure investigations and installation problems of machinery and structures by performing onsite inspections, and compliance analysis to categorize the causes of accidents and failures

Iran Industrial design Company (IID), Tehran, Iran

Senior Design Engineer September 2003 - March 2009

Provided Mechanical design and consulting services for the following cement factories:

1. *Payvand Golestan Cement Co. (PGCC-3450 tons per day)*
   - Assisted in contract preparation and equipment selection
   - Designed factory Layout and flow-sheet
   - Crated General Arrangement drawings for all departments
   - Reviewed detail design of machinery
   - Approved detailed design drawings for material handling machinery, hot gas ducting, bins and stacks

- Approved structural design in accordance to machinery installation and maintenance requirements
- Participated in foreign and domestic technical meetings with clients and contractors
- Conducted onsite evaluations regarding creation, implementation, and quality assurance of design practices, order and fabrication procedures

2. *Gilan Sabz Cement Co. (GSCC-3400 tons per day)*

   - Assisted in flow-sheet design and equipment selection
   - Designed Layout and General Arrangement drawings for all departments
   - Participated in weekly technical meetings with client, domestic, and foreign designers and contractors
   - Reviewed detailed designs from sub-contractors to assure cost effectiveness, reliability and operability

3. *Faraz Firoozkuh Cement Co. (FFCC-3400 tons per day)*

   - Designed Layout and General Arrangement of all departments
   - Assisted in detailed machinery design
   - Created technical guidelines and manuals for assembly and installations
   - Provided technical problem solving strategies during machinery installations

4. *Bohrook Yazd Cement Co. (BYCC-2000 tons/day)*

   - Modified the flow-sheet in renovation phase
   - Created technical documentation of renovation
   - Prepared work breakdown structure
   - Reviewed and modified General Arrangement and detailed machinery design
   - Revised detailed designs from sub-contractors

5. *Tehran Cement Co. (TCC- 3400 tons per day)*

    - Designed General Arrangement of cements mills, raw mill feed, and cement mill feed departments

6. *Kavir Kashan Cement Co. (KKCC-2000 tons per day)*

    - Designed the Kiln shell and tertiary air duct
    - Provided General Arrangement for cement mill feed department

PUBLICATIONS

Khajah T, Dynamic Response of an axially loaded beam on viscoelastic foundation to moving load, *International Forum on Systems and Mechatronics*, 2012, p 232-240.

Khajah T, Hou G, A Direct Path from Geometrical Data to Electromagnetic Field Estimation using Isogeometric Analysis, *ACES 2015.*

Khajah T, Hou G, Parameter Identification for Vertical Ground Reaction Forces on Feet while Running. *(Accepted: journal of sports engineering)*

RESEARCH AND MANUSCRIPTS IN PROGRESS

Khajah T, Audette M A, Bordas S, Hou G, Alleviating the mesh burden in medical simulation and guidance.

PROFESSIONAL CERTIFICATIONS

- Optimization techniques in industrial psychology interaction, SN: IPM8509-07 (2006)
- Welding Process and Welding Inspection (2006)
- Transactional Analysis and Industrial Management (2007)
- National Council of Examiners for Engineers and Surveyors (NCEES, 2011)

MEMBERSHIP IN PROFESSIONAL SOCIETIES

- Life member of Golden Key honor society
- Life Member of the Sharif University of Technology Association (SUTA)
- Life Member of Iranian Society of Consulting Engineers (Member of FIDIC)

PROFICIENCIES

Finite Element Analysis, Isogeometric Analysis, Optimization, Computational Intelligence, Computational Mechanics, Modeling, Simulation

COMPUTER SKILLS

ANSYS, ABAQUS, MATLAB, MAPLE, Inventor, Solid Works

LANGUAGES

English, Persian, Kurdish, Hindi, Urdu

Typeset using LaTeX.