MULTILEVEL FAST MULTIPOLE METHOD FOR MODELING PERMEABLE STRUCTURES USING CONFORMAL FINITE ELEMENTS

by

Kubilay Sertel

A dissertation submitted in partial fulfilment
of the requirements for the degree of
Doctor of Philosophy
(Electrical Engineering)
in the University of Michigan
2003

Doctoral Committee:

Professor John L. Volakis, Chair
Professor Robert Krasny
Research Scientist Valdis V. Liepa
Professor Thomas B. A. Senior

ABSTRACT

MULTILEVEL FAST MULTIPOLE METHOD FOR MODELING PERMEABLE STRUCTURES USING CONFORMAL FINITE ELEMENTS

by

Kubilay Sertel

Chair: John L. Volakis

The analysis of penetrable structures has traditionally been carried out using partial differential equation methods due to the large computation time and memory requirements of integral equation methods. To reduce this computational bottleneck, this thesis focuses on fast integral equation methods for modeling penetrable geometries with both dielectric and magnetic material properties.

Previous works have employed the multilevel fast multipole method for impenetrable targets in the context of flat-triangular geometry approximations. In this thesis, we integrate the multilevel fast multipole method with surface and volume integral equation techniques to accurately analyze arbitrarily curved inhomogeneous targets. It is demonstrated that conformal geometry modeling using curvilinear elements achieve higher accuracy at lower sampling rates. Also, the combined use of curvilinear elements and the multilevel fast multipole method allows for significantly faster and more efficient numerical methods.

The proposed method reduces the traditional $O(N^2)$ computational cost down to $O(N \log N)$ and thus practical size geometries can be analyzed. Several example calculations are given in the thesis along with comparisons with partial differential equation methods.

$$
\begin{aligned}
\nabla \times \mathbf{E}(\mathbf{r}, t) &= -\frac{\partial}{\partial t}\mathbf{B}(\mathbf{r}, t), \\
\nabla \times \mathbf{H}(\mathbf{r}, t) &= -\frac{\partial}{\partial t}\mathbf{D}(\mathbf{r}, t) + \mathbf{J}(\mathbf{r}, t), \\
\nabla \cdot \mathbf{B}(\mathbf{r}, t) &= 0, \\
\nabla \cdot \mathbf{D}(\mathbf{r}, t) &= \rho(\mathbf{r}, t).
\end{aligned}
$$

Dedicated to my parents and my brother.

ACKNOWLEDGMENTS

During my 5.5 years at the University of Michigan, I had the chance to meet a lot great people. Among many them, I would like to name a few that have helped me with their personal and professional support. I feel very happy to have met these great people and value their friendship a lot. Dr. Michael A. Carr, Dr. Dejan S. Filipovic, Mr. Rick W. Kindt, Mr. Dimitris Psychodakis, and Mr. Eng-Swee Siah, I would like to thank all of you guys for making the VCC a great experience.

I would like to thank my committee members, Professor Robert Krasny from Mathematics Department and Professor Thomas B. A. Senior and Professor Valdis V. Liepa from Electrical Engineering and Computer Science Department, for reviewing my work and providing their suggestions.

And finally, I am grateful to my advisor Prof. John L. Volakis for providing all the help and guidance, both professionally and personally, during my Ph.D. study here at Ann Arbor. I consider myself very lucky to have met him and his lovely family, and I am looking forward to continuing to work with in the future.

CONTENTS

LIST OF FIGURES

Figure

LIST OF TABLES

Table

LIST OF APPENDICES

Appendix

CHAPTER 1

Introduction

Evaluation of the electromagnetic properties of complex real-life targets constitutes one of the most demanding computational tasks in mathematics and engineering. Realistic computations using general purpose numerical algorithms quickly lead to millions of degrees of freedom (DOFs). Moreover, the vector nature of the problem, the critical need to treat penetrable complex materials (possibly anisotropic or even non-linear), and the necessity for variable and adaptable gridding coupled with requirements for accuracy to within less than a dB over a large dynamic range serve to exacerbate the situation. As a result, the generated numerical systems are highly heterogeneous and very large. Thus, fast algorithms and highly convergent methods are required for the solution of realistic problems. The need to maintain accuracy down to a fraction of a dB over a large dynamic range of 70 to 100 dB leads to even further challenges.

This thesis is aimed at developing fast algorithms based on the multilevel fast multipole method (MLFMM) for composite structures. Previous developments and applications of MLFMM dealt primarily with surface integral equations (SIEs). The work presented in this thesis is the first to focus on the development and application of MLFMM to volumetric composite structures. The key aspects of the thesis are:

1. Accurate geometry modeling. Accuracy is of most importance in large-scale modeling due to numerical error accumulation as the matrix system size increases. Also, geometry modeling accuracy is crucial as is the characterization of

material properties. This thesis is the first to develop fast solution algorithms for large scale structures utilizing curvilinear surface and volume elements using integral and hybrid integro-differential equation solvers.

2. Volumetric formulations. In the past, volume integral equations were not attractive due to their excessive computational demands. In this thesis, we develop fast algorithms for volumetric composite structures for the first time and demonstrate that such algorithms are very attractive as compared to partial differential equation (PDE) formulations. The use of curvilinear elements is a crucial aspect of the formulation and is rather important when dealing with high contrast dielectric coatings. As reported previously [1], these high contrast dielectrics require precise geometry representation to maintain acceptable levels of accuracy. Also, different approaches to solving volumetric integral equations (VIEs) are considered in terms of efficiency and accuracy. These VIEs are implemented using a single vector unknown within the volume and the MLFMM is applied for a numerically efficient solution of such systems. A substantial contribution of the thesis is concerned with the specific volume formulations and how these lend themselves to accurate, efficient and parallelizable algorithms.

As an alternative to VIE methods, we also implement a finite element-boundary integral (FE-BI) solution using curvilinear elements. For the first time, the MLFMM is implemented for carrying out the matrix-vector products in the boundary integral portion of the matrix system. Thus, the resulting CPU requirements are $O(N \log N)$, where N represents the size of the resulting matrix equation. Our study concludes with a preliminary comparison of the hybrid FE-BI and VIE methods in terms of solution accuracy and utilization of computer resources.

In Chapter 2, surface integral equations (SIEs) for conducting targets are reviewed with particular emphasis on curvilinear elements for surface modeling. Traditionally, solutions of SIEs for arbitrarily shaped conducting targets have mostly been studied

using flat triangular geometry models and Rao-Wilton-Glisson (RWG) [2] basis functions. Higher-order modeling approaches using bilinear quadrilaterals [3], biquadratic quadrilaterals [4], non-uniform rational b-splines (NURBSs) [5], and curvilinear triangular modeling [6] have also been reported for method of moments applications. A comparison of different modeling techniques is given in [6] demonstrating the advantages of using better geometry models. All of the above approaches use suitable conformal basis functions for modeling the unknown surface current [7].

Chapter 2 starts with the pertinent electric and magnetic field integral equations for conducting surfaces and outlines the conventional Method of Moments (MoM) solution algorithm using conformal basis functions and curvilinear surface elements. Formulations for second order conformal surface elements and conformal basis functions defined on these curved elements are given. A summary of the general curvilinear coordinate systems is provided for completeness to clarify various definitions used in the formulations.

An inherent problem in using curvilinear elements is the accurate evaluation of the singular integrals appearing in the self terms of the moment method matrix. This is addressed in Appendix B. Specifically, the annihilation method that transforms the singular integrals into regular functions through a set of parametric transformations is adopted. These methods are also applied to the VIE formulations and are necessary for an accurate numerical evaluation.

In Chapter 3, we outline the MLFMM and its application to SIEs. The Fast Multipole Method (FMM) [8] was introduced to remove the $O(N^2)$ computational bottleneck of traditional MoM solutions of integral equations. Variations of the single level FMM with better than $O(N^{3/2})$ complexities have been reported in [9, 10, 11, 12]. The multilevel version of FMM having a low $O(N \log N)$ complexity was reported in [13]. For completeness, this chapter begins with the single level FMM for the SIE formulations. Following the pertinent FMM equations, key aspects of MLFMM,

namely multilevel grouping, k-space sampling and interpolation methods, memory and computational time savings are discussed. Possible use of the near-field matrix as a preconditioner is also discussed for single level FMM as well as the MLFMM.

To alleviate the $O(N^2)$ CPU bottleneck for modeling dielectric volumes, k–space methods utilizing iterative solution approaches such as the conjugate gradient fast Fourier transform method (CGFFT) were introduced in the late 1980s [14, 15, 16, 17]. These latter approaches employ the FFT to reduce storage and CPU requirements down to $O(N)$ and $O(N \log N)$, respectively. However, by virtue of the FFT, they lack accuracy in modeling curvilinear dielectric structures. This is particularly so for high contrast dielectrics. As demonstrated in [1], extremely high sampling is required for accurate solutions without the use of conformal elements. Straightforward application of the MoM to solving integral equations results in a matrix system of size N, where N is proportional to the electrical size of the geometry. For a numerical solution of the problem, the system matrix has N^2 non-zero entries and hence computer storage requirements are of $O(N^2)$. The solution of this system requires $O(N^3)$ floating point operations (flops) using a direct matrix decomposition method (such as the LU decomposition). For large N, the $O(N^3)$ CPU cost leads to prohibitive CPU and memory requirements. However, an iterative method requires only $O(N^2)$ flops per iteration on the provision that convergence is achieved in a few number of iterations. Preconditioning methods can play a significant role in improving the convergence of the iterative solver. The matrix-vector product in each iteration of the iterative solver can correctly be identified as a field translation operation of the basis (radiating) functions onto the testing (observation) domains. Utilizing multipole expansions for clusters of basis (source) functions, we can reduce the number of operations required to evaluate the radiated fields on the observation domains using fewer number of flops.

A single level grouping strategy results in a matrix-vector product evaluated in

$O(N^{3/2})$ flops. This is the so called Fast Multipole Method (FMM). When the multipole expansions are utilized in a multilevel scheme (MLFMM), the resulting complexity of the method can be reduced down to a remarkable $O(N \log N)$. Moreover, since the whole MoM matrix need not be computed explicitly, the storage cost is also reduced down to $O(N^{3/2})$ for the single level FMM and down to $O(N \log N)$ for the MLFMM. Similar to the FFT, the MLFMM is a fast and numerically exact way of carrying out the matrix-vector product operation (which is essentially a field translation operation) very efficiently.

We also examine the implementation of the MLFMM on distributed memory parallel computers. Inter-processor communication issues are discussed for a general purpose parallel implementation. With such parallel implementations, it is possible to solve very large scale electromagnetics problems on low cost personal computer clusters. This is only achievable with careful utilization of accurate higher-order geometry models and adaptation of fast solvers such as the MLFMM.

In the third chapter of the thesis, we present the application of MLFMM to hybrid FE-BI formulations for inhomogeneous problems. In this chapter, the FE-BI method [18] using curvilinear volume and surface modeling [19] is outlined. In addition to conformal surface elements, conformal volume elements are introduced. These volume elements form the basis for the discretization of the VIEs that will be outlined in the subsequent chapter. The extension of MLFMM to FE-BI systems is presented as a totally new contribution. Strategies to reduce computer time and memory as well as matrix conditioning issues for various choices of surface testing functions are discussed.

Several methods have been used to formulate scattering and radiation problems involving penetrable materials. The finite element method (FEM) [18] along with various mesh truncation schemes is one of the most commonly used approaches. Among these FEM methods, the FE-BI method [18, 19, 20, 21] provides an exact

means of truncating the FEM mesh, hence keeping the FEM domain small. This is crucial in numerical simulations since the FEM method is prone to error propagation. The necessity of using suitable geometry modeling schemes and basis functions in the FE-BI formulation are also demonstrated, both in terms of solution accuracy and convergence. As an alternative to partial differential equation methods, for problems involving only homogeneous material regions, a surface integral equation formulation can be used [22]. However, for regions with varying material properties a VIE must be employed [23, 24]. In [23], cubic elements were used in this context and tetrahedral elements were used in [24]. For curved geometries, tetrahedral elements [25, 26] are more suitable than cubic elements due to their geometrical adaptability. Curved hexahedral elements [19] are capable of modeling arbitrary geometries better than tetrahedra. When dealing with thin layers, curved hexahedral elements are especially more suitable since they avoid elongated tetrahedra which can lead to ill-conditioned matrix systems.

So far, the FE-BI approach has not been exploited due to the excessive CPU and memory requirements for large scale computations. Nevertheless, the MLFMM can readily be incorporated to reduce the computational burden. Also, MLFMM may allow for practical implementations of VIEs.

The last part of this thesis develops VIEs for composite targets. The modeling of inhomogeneous structures has traditionally [27, 28] been carried out using equivalent volume currents. Initial numerical implementations were done by Richmond [29] for two-dimensional scatterers and later by Livesay and Chen [23] and Schaubert et. al. [24] for three dimensions (see also Peterson [30] and Graglia et. al. [31]). Direct use of equivalent currents to formulate integral equations is known to lead to 6 scalar unknowns per location when both the permittivity and permeability are different from those of free-space. As a result, volumetric structures over a wavelength per linear dimension lead to very large numerical systems. Further, excessive sampling

requirements [1] for high contrast materials add to the computational burden. Thus, numerical simulations have so far been focused on purely dielectric bodies ($\mu_r = 1$).

To reduce the number of unknowns per volume location, two approaches can be followed. One is to combine the equivalent electric and magnetic currents into a single equivalent electric (or magnetic) current density $\mathbf{J}_{eq}$ (or $\mathbf{M}_{eq}$) [32]. Another is to reduce the volume integral equation (VIE) into a volume-surface integral equation (VSIE) as done in [1, 32] by invoking several integral identities and the divergence theorem. The resulting integral equation involves differentiation of the permeability within the inhomogeneous region. This differentiation may compromise the accuracy of the formulation for high contrast dielectrics, and is further undesirable for piecewise constant volumes.

The VIE development begins with a rigorous derivation of the appropriate integral equation for efficient modeling of volume regions. In particular, we propose a VIE which employs a single vector field unknown (instead of two vector unknowns as in the past). The application of the MLFMM to this VIE is another important contribution of this thesis. We present validation of the VIE and compare the solutions for various composite targets using various higher-order volumetric basis functions. The performance of the MLFMM solution is also demonstrated for electrically large dielectric targets. Further, the VIE and FE-BI methods are compared both in terms of solution accuracy and computational resources. The low computational cost and superior error performance of the VIE-MLFMM solutions as compared to the FE-BI-MLFMM solutions are demonstrated.

The thesis concludes with a summary of major contributions and solution methods for arbitrarily shaped inhomogeneous geometries.

CHAPTER 2

Surface Integral Equations

In this chapter, we outline the governing surface integral equations (SIEs) used to formulate scattering by perfectly electrically conducting (PEC) bodies. The existence and derivation of such integral equations starting with Maxwell's equations will not be discussed here, as they are quite standard in electromagnetic theory. The reader is referred to referenced classical textbooks on advanced electromagnetic theory [27, 33]. Next, we first proceed with a description of the method of moments (MoM) employed to solve the given integral equations. In doing so, we describe our geometry modeling approach and specific basis functions for the discretization of the unknown induced surface current density in the integral equations.

2.1 Surface Integral Equations for PEC Structures

In the presence of an external excitation, such as an incident electromagnetic field due to an impressed source, the scattered field due to a PEC body (see Fig. 2.1) can be calculated by replacing the object's surface by an induced current source radiating a scattered field in free-space, viz.

$$\mathbf{E}^{scat}(\mathbf{r}) = i\omega\mu_0 \int_s d\mathbf{r}' \left[\mathbf{J}(\mathbf{r}') + \frac{1}{k^2}\nabla' \cdot \mathbf{J}(\mathbf{r}')\nabla\right] g(\mathbf{r}, \mathbf{r}') \tag{2.1}$$

where $\mathbf{E}^{scat}$ denotes the scattered field, ω is the frequency of operation, μ_0 is the free-space permeability, $\mathbf{J}(\mathbf{r})$ is the induced surface current on the PEC surface, $g(\mathbf{r}, \mathbf{r}') = \exp(ik|\mathbf{r} - \mathbf{r}'|)/4\pi|\mathbf{r} - \mathbf{r}'|$ is the free-space scalar Green's function with k being the wavenumber in the background medium. Throughout this thesis, an $e^{i\omega t}$

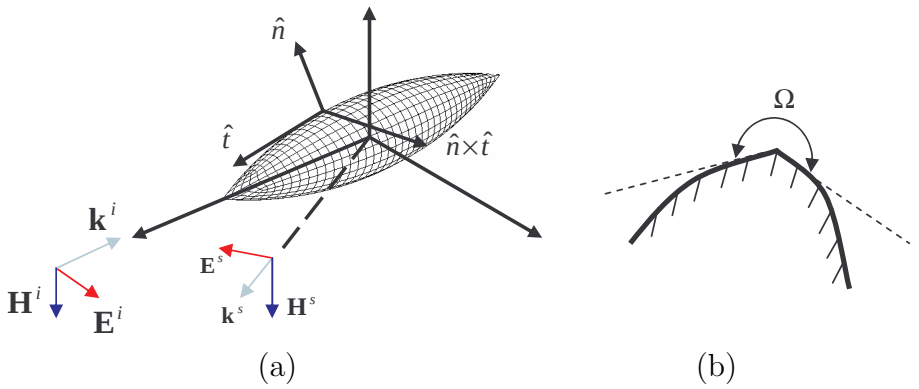


Figure 2.1. PEC scatterer in the presence of an external excitation. (a) Example surface mesh of an ogive using conformal quadrilaterals, (b) Definition of the solid angle Ω.

time convention is assumed and suppressed.

The integral equation for this problem can be constructed by merely enforcing the boundary condition for the tangential electric field component on the PEC surface s. Enforcing the total tangential electric field to vanish on s, i.e. $\hat{t}\cdot(\mathbf{E}^{inc}+\mathbf{E}^{scat})=0$, where $\hat{t}$ is the tangent unit vector to the body's surface s (see Fig. 2.1 (a)), yields the so called electric field integral equation (EFIE)

$$\hat{t}\cdot\int_s d\mathbf{r}'\overline{\mathbf{G}}(\mathbf{r},\mathbf{r}')\cdot\mathbf{J}(\mathbf{r}')=\frac{i}{k\eta}\hat{t}\cdot\mathbf{E}^{inc}(\mathbf{r}), \tag{2.2}$$

where $\mathbf{E}^{inc}(\mathbf{r})$ is the incident electric field and $\eta=\sqrt{\mu/\epsilon}$ is the impedance of the background medium. Here, we have equivalently adopted the more compact dyadic Green's function representation with

$$\overline{\mathbf{G}}(\mathbf{r},\mathbf{r}')=\left[\bar{\mathbf{I}}+\frac{1}{k^2}\nabla\nabla\right]g(\mathbf{r},\mathbf{r}'). \tag{2.3}$$

Similarly, using the boundary condition $\mathbf{J}=\hat{n}\times(\mathbf{H}^{inc}+\mathbf{H}^{scat})$ [1], it is possible to derive the so called magnetic field integral equation (MFIE) valid for closed scatterers and given by

$$-T\hat{n}\times\mathbf{J}(\mathbf{r})+\oint_s d\mathbf{r}'\mathbf{J}(\mathbf{r}')\times\nabla g(\mathbf{r},\mathbf{r}')=\hat{n}\times\mathbf{H}^{inc}(\mathbf{r}). \tag{2.4}$$

[1]The enforced boundary condition is the usual $\mathbf{J}=\hat{n}\times(\mathbf{H}^{+}-\mathbf{H}^{-})$, where $\mathbf{H}^{+}=\mathbf{H}^{inc}+\mathbf{H}^{scat}$ is the exterior field and $\mathbf{H}^{-}=0$ within the PEC body.

In this, $T = 1-\Omega/4\pi$ and Ω is the solid angle subtended by the observation point (see Fig. 2.1 (b)) and the bar through the integral sign represents the principal value excluding the observation point $\mathbf{r}$. For smooth surfaces, $\Omega = 2\pi$ and hence $T = 1/2$. Also, in the above, $\hat{n}$ denotes the normal vector on the closed surface S pointing outward as shown in Fig. 2.1 (a).

For arbitrary geometries, the solution of the above EFIE and MFIE must be carried out numerically. The MoM is the most popular numerical solution approach. It is based in expanding the unknown $\mathbf{J}$ using a suitable set of basis functions. Below, we present the MoM solution of (2.2) and (2.4) starting by introducing the geometry modeling approach and basis functions.

2.2 Geometry Modelling and Basis Functions

Real-life electromagnetics problems almost always involve arbitrary geometries composed of quite complicated components having arbitrary curvatures and intricate detail. A military aircraft, for example, has a streamlined fuselage with very thin wings attached along with attached munition and various antennas. To mathematically represent such complicated structures, one must resort to simpler geometry modeling techniques. Among several popular and powerful modeling methods [34, 35], we use parametric elements to model scatterer geometries. As compared to flat facet modeling, conformal elements lower the solution error by improving geometry representations. Below, we outline the mathematical basis for parametric surface modeling and present the specific modeling technique used throughout this work.

Parametric representation is the mapping of a unit element (be it a triangle or a square or a general multilateral) in a parametric (u, v) space through the transformation

$$\mathbf{r} = \mathbf{r}(u, v). \tag{2.5}$$

Usually such a parametric representation is constructed as a Cartesian product of

one-dimensional transformations. Fig. 2.2 depicts several one-dimensional parametric mappings. Basically, the parametric curve in the (x, y, z) space is mapped on a unit

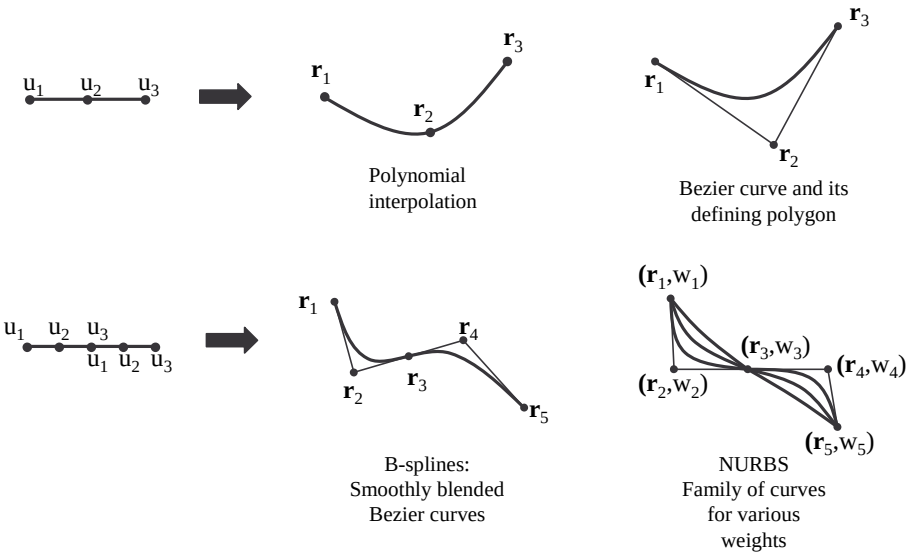


Figure 2.2. Examples of parametric mappings.

straight line in the parametric (u) space through $\mathbf{r} = \mathbf{r}(u)$. Likewise, a curvilinear quadrilateral element in the (x, y, z) space is the image of the unit square in the (u, v) space through a different form of (2.5). Needless to say, (2.5) is the mathematical representation of the curvilinear element in the (x, y, z) coordinates. Several common elements are depicted in Fig. 2.3.

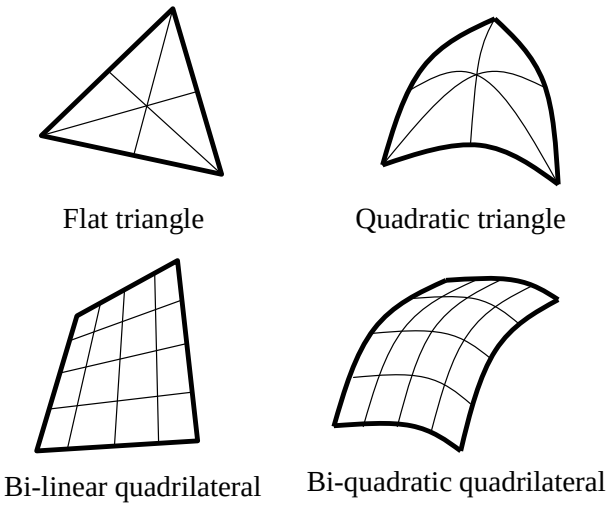


Figure 2.3. Examples of surface elements.

In our work, we will use biquadratic quadrilateral surface elements, each defined by a set of 9 points in space $\{\mathbf{r}_{ij}, i, j = 0, 1, 2\}$ on a topologically rectangular grid as shown in Fig. 2.4. Given this set of 9 defining points for each element, we can form

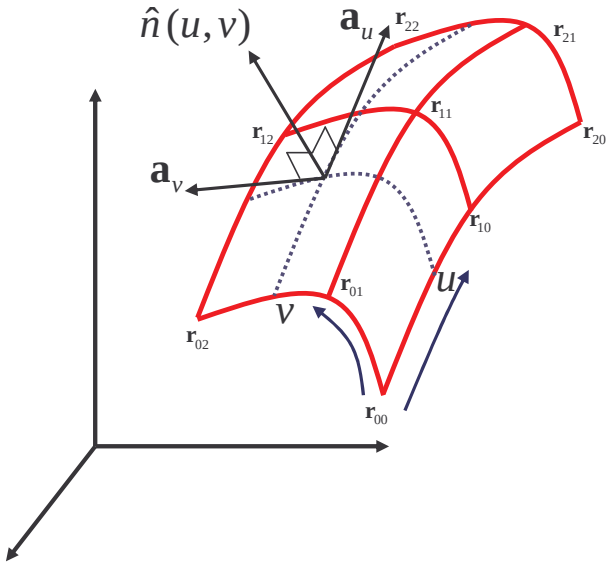


Figure 2.4. Curvilinear parametric surface element defined by 9 points.

the transformation of a unit square in the (u, v) parametric space through a simple two-dimensional interpolation, viz.

$$\mathbf{r}(u, v) = \sum_{i=0}^{2} \sum_{j=0}^{2} L_{ij}(u, v)\, \mathbf{r}_{ij} \tag{2.6}$$

where $L_{ij}(u, v)$ are the Cartesian products of the usual Lagrange interpolation functions (see Appendix A).

The whole geometry of the problem is constructed by a connected mesh of such curvilinear elements (see Fig.2.5 for an example). Use of conformal elements results in better modeling of curved components of the problem geometry including the finer details which would otherwise require a much finer mesh density to resolve. Furthermore, curvilinear elements reduces the overall unknown count and since the problem's computational requirement is dependent on the size of the matrix, curvilinear elements lead to less memory and CPU. These are key advantages of curved surface modeling as have been reported many times in the past [1, 4, 6, 19, 36].

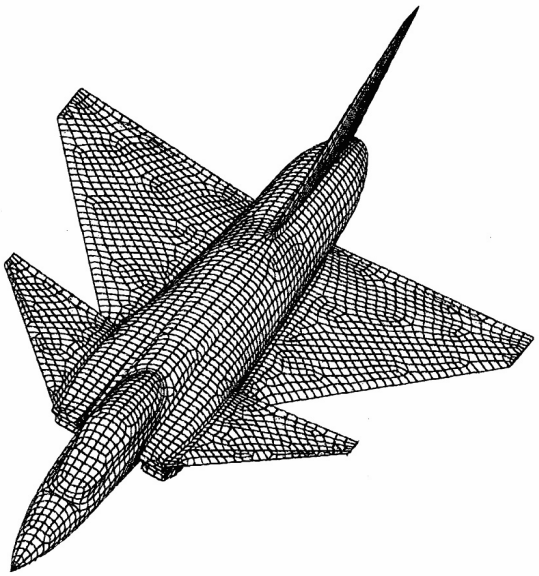


Figure 2.5. Generic VFY218 aircraft modelled by quadrilateral patches.

Once the geometry is mathematically represented with a mesh of connected elements, the next step is to define basis functions conformal to these elements for representing the unknown induced surface current density. This allows us to convert the SIE into a matrix system through the MoM procedure. The intricate relationships between the mesh density, problem size, and solution time will be discussed in detail later in the context of fast solution algorithms.

2.3 MoM Procedure for EFIE

To discretize and solve (2.2) for the unknown surface current, we introduce a linear representation of the unknown $\mathbf{J}$ in terms of known basis functions with unknown coefficients. Mathematically, this corresponds to projecting the vector function $\mathbf{J}$ onto a finite dimensional sub-space spanned by the set of basis $\{\mathbf{j}_1, \mathbf{j}_2, ..., \mathbf{j}_N\}$. We will not repeat the details of the general MoM procedure here. Instead, we will only outline the specific MoM solution for the governing integral equation.

The basis functions for the induced surface electric currents are constructed as a generalization of rooftop basis functions defined on a pair of flat rectangular do-

mains [37]. Such basis functions are often referred to as sub-domain basis functions (since they have finite support). For each biquadratic surface element, four half basis functions are defined in terms of the parametric surface derivatives, each associated with one of the quadrilateral's edges (see Fig. 2.6 (a) for local edge numbering). These half basis functions on each element are defined as

$$\begin{aligned} \mathbf{j}_1 &= \frac{u}{\sqrt{G_s}}\mathbf{a}_u, \quad \mathbf{j}_2 = \frac{1}{\sqrt{G_s}}(1-u)\mathbf{a}_u, \\ \mathbf{j}_3 &= \frac{v}{\sqrt{G_s}}\mathbf{a}_v, \quad \mathbf{j}_4 = \frac{1}{\sqrt{G_s}}(1-v)\mathbf{a}_v, \end{aligned} \tag{2.7}$$

where $\mathbf{a}_u = \partial\mathbf{r}/\partial u$ and $\mathbf{a}_v = \partial\mathbf{r}/\partial v$ and G_s is the determinant of the metric tensor of the transformation (2.5) given by

$$G_s = \begin{vmatrix} g_{uu} & g_{uv} \\ g_{vu} & g_{vv} \end{vmatrix} \tag{2.8}$$

where $g_{ij} = \mathbf{a}_i \cdot \mathbf{a}_j$, with $i = (u, v)$ and $j = (u, v)$. This metric completely characterizes the geometrical properties of the parametric surface. We remark that the vector associated with each half basis function $\mathbf{j}_i$ is tangent to the surface. Also, the amplitude of the basis function is maximum at the representing edge and vanishes linearly (in parametric space) to zero at the opposite edge of the element. This is displayed in Fig. 2.6 (b).

These definitions ensure the conformality of the basis functions and retain the same linear variation of the rooftop basis functions in the parametric space. It can be easily verified that for the above definitions, the divergence of the basis functions is constant in parametric space. Properly pairing these half rooftop basis functions on neighboring patches allows for a continuous current representation on the entire body. It can be verified that the normal component of the current across neighboring patches is continuous, i.e. it is guaranteed that there is no artificial charge accumulation on the common edge joining a pair of elements on which the basis function is defined. We further remark that more general higher order basis functions can be constructed

if the linear variations in (2.7) are replaced by appropriate functions. However, the accuracy of such a representation should be correlated to the accuracy of the actual geometry modeling. That is, one should not pursue a higher-order representation of the current density aiming to improve solution accuracy if the surface representation is not equally accurate. Nevertheless, with accurate geometry representations, higher-order basis functions have been reported to generate smaller MoM systems hence require less computational resources [38].

For completeness, we note here the basic differential operations associated with parametric surfaces. The surface gradient of a scalar function $\phi(u,v)$ is given by

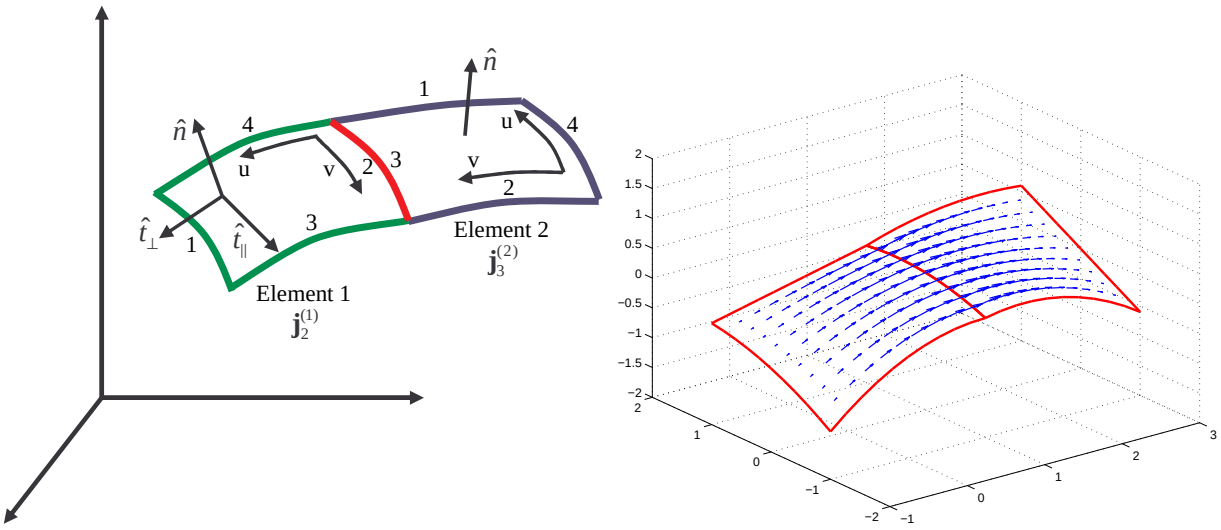
$$\nabla_s \phi(u,v) = g^{uu}\frac{\partial \phi}{\partial u}\mathbf{a}_u + g^{uv}\frac{\partial \phi}{\partial u}\mathbf{a}_v + g^{vu}\frac{\partial \phi}{\partial v}\mathbf{a}_u + g^{vv}\frac{\partial \phi}{\partial v}\mathbf{a}_v \tag{2.9}$$

where $g^{ij}, i=(u,v)$ and $j=(u,v)$ are the elements of the inverse metric tensor. Also, the surface divergence of a vector function $\mathbf{f}(u,v) = f_u\mathbf{a}_u + f_v\mathbf{a}_v$ is given by

$$\begin{aligned} \nabla_s \cdot \mathbf{f}(u,v) &= g^{uu}\frac{\partial \mathbf{f}}{\partial u}\cdot\mathbf{a}_u + g^{uv}\frac{\partial \mathbf{f}}{\partial u}\cdot\mathbf{a}_v + g^{vu}\frac{\partial \mathbf{f}}{\partial v}\cdot\mathbf{a}_u + g^{vv}\frac{\partial \mathbf{f}}{\partial v}\cdot\mathbf{a}_v \\ &= \frac{1}{\sqrt{G_s}}\left(\frac{\partial(f_u\sqrt{G_s})}{\partial u} + \frac{\partial(f_v\sqrt{G_s})}{\partial v}\right). \end{aligned} \tag{2.10}$$

Figure 2.6 depicts two quadrilaterals, each arbitrarily oriented, sharing a common edge. As mentioned, each basis function is defined on a pair of adjoining elements, and the definition (2.7) ensures the continuity of the normal current density component across the common edge. For the depicted situation, the half basis function defined on element 1 will be $\mathbf{j}_2^{(1)} = -\frac{1}{\sqrt{G_s}}(1-u)\mathbf{a}_u$ where the superscript denotes the element number and the subscript refers to the local edge number. Similarly, the half basis function on element 2 is $\mathbf{j}_3^{(2)} = -\frac{v}{\sqrt{G_s}}\mathbf{a}_v$, where the minus signs are introduced to correct for the direction of current flow. The normal component of $\mathbf{j}_2^{(1)}$ across the common edge can be found using the tangential and normal unit vectors to that edge, i.e.

$$\hat{t}_{\parallel} = \frac{1}{\sqrt{g_{vv}}}\mathbf{a}_v, \quad \hat{t}_{\perp} = \hat{t}_{\parallel}\times\hat{n} = \frac{1}{\sqrt{G_s g_{vv}}}\left[g_{vv}\mathbf{a}_u - g_{uv}\mathbf{a}_v\right]. \tag{2.11}$$



(a) (b)

Figure 2.6. Constructing basis functions: (a) Two patches forming the support of basis function associated with the common edge, (b) Quiver plot of the conformal basis (length of arrows is associated with the amplitude of the basis function).

Using (2.11), the normal component of $\mathbf{j}_2^{(1)}$ and $\mathbf{j}_3^{(2)}$ are given by

$$\hat{t}_\perp \cdot \mathbf{j}_2^{(1)} = \frac{1}{\sqrt{g_{vv}(u=1,v)}} \tag{2.12}$$

$$\hat{t}_\perp \cdot \mathbf{j}_3^{(2)} = \frac{1}{\sqrt{g_{uu}(u,v=0)}}. \tag{2.13}$$

Since both values only depend on the differential length over the common edge, the normal components of both halves of the basis function will be equal, indicating continuity of that component. We can also verify that the surface divergence on both halves are $\nabla_s \cdot \mathbf{j}_2^{(1)} = 1/\sqrt{G_s}$ and $\nabla_s \cdot \mathbf{j}_3^{(2)} = -1/\sqrt{G_s}$, implying constant surface charge $\sigma = \nabla_s \cdot \mathbf{j}$ in the parametric space (i.e. $\sigma ds/dudv = \pm 1$)[2].

To proceed with the MoM implementation, we assume that the unknown function $\mathbf{J}$ is a linear combination of the defined basis functions as $\mathbf{J}(\mathbf{r}) = \sum_{i=1}^{N} x_i \mathbf{j}_i(\mathbf{r})$, where x_i are the unknown coefficients of the expansion. When this expansion is substituted in (2.2), we get

$$\hat{t} \cdot \int_s d\mathbf{r}' \overline{\mathbf{G}}(\mathbf{r},\mathbf{r}') \cdot \sum_{i=1}^{N} x_i \mathbf{j}_i(\mathbf{r}') = \frac{i}{k\eta} \hat{t} \cdot \mathbf{E}^{inc}(\mathbf{r}) \tag{2.14}$$

[2] $ds = \sqrt{G_s} dudv$

or equivalently,

$$\sum_{i=1}^{N} x_i \hat{t} \cdot \int_s d\mathbf{r}' \overline{\mathbf{G}}(\mathbf{r}, \mathbf{r}') \cdot \mathbf{j}_i(\mathbf{r}') = \frac{i}{k\eta} \hat{t} \cdot \mathbf{E}^{inc}(\mathbf{r}). \tag{2.15}$$

To find the unknown coefficients x_i, we employ Galerkin's testing (for Galerkin's testing $\mathbf{t}_j = \mathbf{b}_j$) to obtain

$$\sum_{i=1}^{N} x_i \int_s d\mathbf{r} \mathbf{t}_j(\mathbf{r}) \cdot \int_s d\mathbf{r}' \overline{\mathbf{G}}(\mathbf{r}, \mathbf{r}') \cdot \mathbf{j}_i(\mathbf{r}') = \frac{i}{k\eta} \int_s d\mathbf{r} \mathbf{t}_j(\mathbf{r}) \cdot \mathbf{E}^{inc}(\mathbf{r}). \tag{2.16}$$

For $j = 1, ..., N$, this leads to N linear equations for the solution of the N unknowns. Equation (2.16) can thus be cast into a matrix system as $[Z]\{x\} = \{b\}$, where

$$Z_{ji} = \int_s d\mathbf{r} \mathbf{t}_j(\mathbf{r}) \cdot \int_s d\mathbf{r}' \overline{\mathbf{G}}(\mathbf{r}, \mathbf{r}') \cdot \mathbf{j}_i(\mathbf{r}') \tag{2.17}$$

and

$$b_j = \frac{i}{k\eta} \int_s d\mathbf{r} \mathbf{t}_j(\mathbf{r}) \cdot \mathbf{E}^{inc}(\mathbf{r}) \tag{2.18}$$

being the excitation vector due to the incident wave $\mathbf{E}^{inc}(\mathbf{r})$. Solution of this system provides the unknown coefficients of the expansion $\mathbf{J}(\mathbf{r}) = \sum_{i=1}^{N} x_i \mathbf{j}_i(\mathbf{r})$ for the induced surface current density.

The double surface integrals in the matrix element in (2.17) need to be numerically evaluated using a suitable quadrature rule. Since each basis and testing function is defined on pairs of elements it is advantageous to compute the contributions to the matrix entries from pairs of basis and testing domains (i.e. compute element sub-matrices). Subsequently, these contributions are added into the actual matrix using the connectivity of the mesh, much like it is done in standard finite element method. That is, the 4×4 element matrix $\widetilde{Z}_{kl}$ between the l^{th} source element and the k^{th} testing element (see Fig. 2.7) is computed in the parametric (u, v) and (u', v') unit squares as

$$\widetilde{Z}_{kl} = \int_0^1 \int_0^1 d\mathbf{r} \mathbf{t}'_j(\mathbf{r}) \cdot \int_0^1 \int_0^1 d\mathbf{r}' \overline{\mathbf{G}}(\mathbf{r}, \mathbf{r}') \cdot \mathbf{j}'_i(\mathbf{r}') \tag{2.19}$$

where $\mathbf{t}'_j$ denotes each of the 4 half-testing functions on element k and $\mathbf{j}'_i$ denotes each of the 4 half-basis functions on element l. Using $d\mathbf{r} = \sqrt{G_s} du dv$ and $d\mathbf{r}' = \sqrt{G'_s} du' dv'$,

$$Z_{el} = \begin{bmatrix} Z_{11} & Z_{12} & Z_{13} & Z_{14} \\ Z_{21} & Z_{22} & Z_{23} & Z_{24} \\ Z_{31} & Z_{32} & Z_{33} & Z_{34} \\ Z_{41} & Z_{42} & Z_{43} & Z_{44} \end{bmatrix}$$

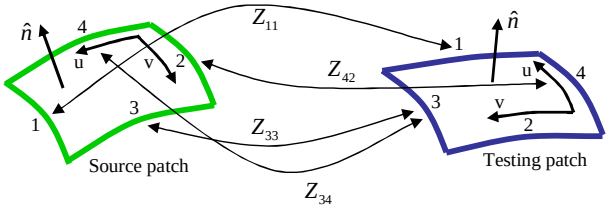


Figure 2.7. Illustration for the computation of the element matrices.

we can rewrite (2.19) as

$$\widetilde{Z}_{kl} = \int_{s_k} \sqrt{G_s} du dv \mathbf{t}'_j(\mathbf{r}) \cdot \int_{s_l} \sqrt{G'_s} du' dv' \overline{\mathbf{G}}(\mathbf{r}, \mathbf{r}') \cdot \mathbf{j}'_i(\mathbf{r}'). \tag{2.20}$$

Recalling the definitions of basis functions in (2.7), we realize that the normalization factor $1/\sqrt{G_s}$ need never actually be computed since it is cancelled by the Jacobian (i.e. the differential area factor) in (2.20). The hyper-singularity in (2.20) can be relaxed through the use of the divergence theorem. When we explicitly state the dyadic Green's function in (2.17), we obtain

$$Z_{ji} = \int_s d\mathbf{r} \mathbf{t}_j(\mathbf{r}) \cdot \left[\int_s d\mathbf{r}' g(\mathbf{r}, \mathbf{r}') \mathbf{j}_i(\mathbf{r}') + \frac{1}{k^2} \int_s d\mathbf{r}' \nabla \nabla g(\mathbf{r}, \mathbf{r}') \cdot \mathbf{j}_i(\mathbf{r}') \right] \tag{2.21}$$

or equivalently,

$$Z_{ji} = \int_s d\mathbf{r} \mathbf{t}_j(\mathbf{r}) \cdot \left[\int_s d\mathbf{r}' g(\mathbf{r}, \mathbf{r}') \mathbf{j}_i(\mathbf{r}') - \frac{1}{k^2} \nabla \int_s d\mathbf{r}' \nabla' g(\mathbf{r}, \mathbf{r}') \cdot \mathbf{j}_i(\mathbf{r}') \right]. \tag{2.22}$$

We can transfer the ∇' operator to the basis function using the vector identity $\nabla' \cdot (g\mathbf{j}_i) = \nabla' g \cdot \mathbf{j}_i + g \nabla' \cdot \mathbf{j}_i$ and the divergence theorem. These manipulations result in

$$\begin{aligned} Z_{ji} &= \int_s d\mathbf{r} \mathbf{t}_j(\mathbf{r}) \cdot \left[\int_s d\mathbf{r}' g(\mathbf{r}, \mathbf{r}') \mathbf{j}_i(\mathbf{r}') \right. \\ &- \left. \frac{1}{k^2} \nabla \int_s d\mathbf{r}' \left\{ \nabla' \cdot [g(\mathbf{r}, \mathbf{r}') \mathbf{j}_i(\mathbf{r}')] - g(\mathbf{r}, \mathbf{r}') \nabla' \cdot \mathbf{j}_i(\mathbf{r}') \right\} \right] \\ &= \int_s d\mathbf{r} \mathbf{t}_j(\mathbf{r}) \cdot \left[\int_s d\mathbf{r}' g(\mathbf{r}, \mathbf{r}') \mathbf{j}_i(\mathbf{r}') + \frac{1}{k^2} \nabla \int_s d\mathbf{r}' g(\mathbf{r}, \mathbf{r}') \nabla' \cdot \mathbf{j}_i(\mathbf{r}') \right] \\ &- \int_s d\mathbf{r} \mathbf{t}_j(\mathbf{r}) \cdot \left[\frac{1}{k^2} \nabla \oint_c d\mathbf{r}' \cdot g(\mathbf{r}, \mathbf{r}') \mathbf{j}_i(\mathbf{r}') \right] \end{aligned} \tag{2.23}$$

where $d\mathbf{r}'$ of the closed line integral is the normal differential vector to the boundary c of the basis domain s (see Fig. 2.8). Since $d\mathbf{r}'$ is always perpendicular to the basis function $\mathbf{j}_i(\mathbf{r}')$, this integral vanishes, hence

$$Z_{ji} = \int_s d\mathbf{r}\mathbf{t}_j(\mathbf{r}) \cdot \left[\int_s d\mathbf{r}' g(\mathbf{r},\mathbf{r}')\mathbf{j}_i(\mathbf{r}') + \frac{1}{k^2}\nabla \int_s d\mathbf{r}' g(\mathbf{r},\mathbf{r}')\nabla' \cdot \mathbf{j}_i(\mathbf{r}') \right]. \tag{2.24}$$

We can again use the divergence theorem $\nabla \cdot (\phi\mathbf{A}) = \nabla\phi \cdot \mathbf{A} + \phi\nabla \cdot \mathbf{A}$ on the second term in the above integral with $\mathbf{A} = \mathbf{t}_j(\mathbf{r})$ and $\phi = \int_s d\mathbf{r}' g(\mathbf{r},\mathbf{r}')\nabla' \cdot \mathbf{j}_i(\mathbf{r}')$ and rewrite (2.24) as

$$\begin{aligned} Z_{ji} &= \int_s d\mathbf{r}\mathbf{t}_j(\mathbf{r}) \cdot \int_s d\mathbf{r}' g(\mathbf{r},\mathbf{r}')\mathbf{j}_i(\mathbf{r}') \\ &+ \frac{1}{k^2} \oint_c d\mathbf{r} \cdot \left[\mathbf{t}_j(\mathbf{r}) \int_s d\mathbf{r}' g(\mathbf{r},\mathbf{r}')\nabla' \cdot \mathbf{j}_i(\mathbf{r}') \right] \\ &- \frac{1}{k^2} \int_s d\mathbf{r}\nabla \cdot \mathbf{t}_j(\mathbf{r}) \int_s d\mathbf{r}' g(\mathbf{r},\mathbf{r}')\nabla' \cdot \mathbf{j}_i(\mathbf{r}') \end{aligned} \tag{2.25}$$

where $d\mathbf{r}$ is the normal differential vector to the boundary c of the testing domain s (as depicted in Fig. 2.8). Since $d\mathbf{r}$ and $\mathbf{t}_j(\mathbf{r})$ are always perpendicular to each other

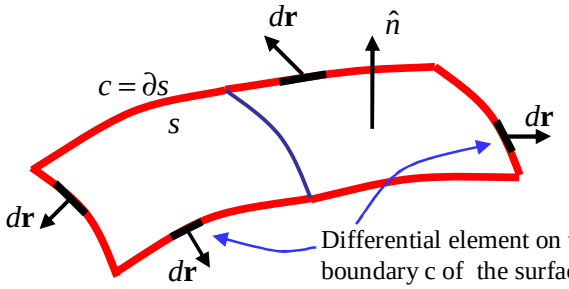


Figure 2.8. Application of the divergence theorem on the testing function.

on the outer boundary c, the line integral in (2.25) is identically zero, yielding the desired equation

$$Z_{ji} = \int_s d\mathbf{r}\mathbf{t}_j(\mathbf{r}) \cdot \int_s d\mathbf{r}' g(\mathbf{r},\mathbf{r}')\mathbf{j}_i(\mathbf{r}') - \frac{1}{k^2} \int_s d\mathbf{r}\nabla \cdot \mathbf{t}_j(\mathbf{r}) \int_s d\mathbf{r}' g(\mathbf{r},\mathbf{r}')\nabla' \cdot \mathbf{j}_i(\mathbf{r}'). \tag{2.26}$$

After this manipulation, both of the integrals in (2.25) have a first order singularity when the source and testing domains overlap. This is an integrable singularity, nevertheless care must be taken in numerically evaluating the singular kernels. We have

used an annihilation technique for integrating the singularity in the source integral for the self term (i.e. when testing and basis domains overlap). This technique is outlined in the Appendix B. In the numerical computation of (2.26), we have used Gaussian quadrature. Several orders of Gaussian quadrature points on the unit square in the (u, v) parametric space are depicted in Fig. 2.9.

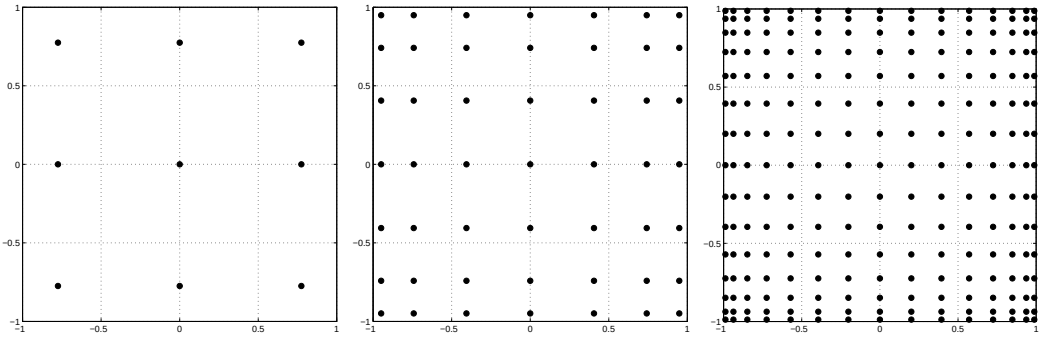


Figure 2.9. Several orders of Gaussian quadrature points for $-1 < u < 1$ and $-1 < v < 1$.

Once the elements of the matrix equation are accurately computed, the matrix system must be solved to get the unknown coefficients x_i for $\{i = 1, ..., N\}$. To do so, one can use a direct solution method, such as Gaussian elimination (or LU decomposition), which however is associated with $O(N^3)$ computational cost. Alternatively, an iterative solution method [39] with $O(N^2)$ complexity per iteration can be used. For the latter case, the hope is to achieve convergence in a few iterations. Several preconditioning methods can also be utilized to achieve quick convergence [40]. In the next section, we present examples and validations of the outlined methodology. Since the MoM is a well-established method for solving integral as well as partial differential equations [41], and since an error analysis for the above procedure is well beyond the scope of this work (and is probably impossible for an arbitrary geometry), here we only demonstrate the convergence of the numerical results as the mesh density is increased. However, as a rule of thumb, a $\lambda/10$ element size will be adopted. This rule aims at accurately resolving variations in the induced surface current using rooftop (linear) basis functions. This is demonstrated below as the results are accu-

rate to within less than 1 % for $\lambda/10$ discretization sizes for all test geometries. We also demonstrate below the superior geometry modeling capability of the outlined curvilinear elements.

2.4 Examples and Validations

To validate the presented method, we consider several PEC scatterers each having unique characteristics. Surface meshes of these test geometries using curvilinear quadrilaterals are given in Fig. 2.10. For all given results, a fixed mesh density for

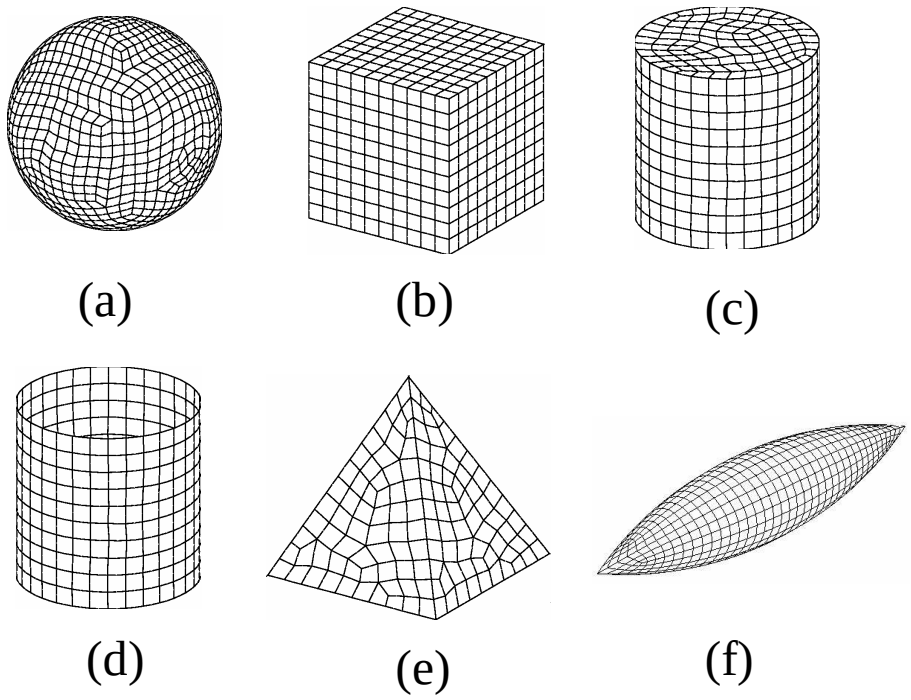


Figure 2.10. Test geometries: (a) Sphere, (b) Cube, (c) Cylinder, (d) Open-ended cylinder, (e) Pyramid, (f) Ogive.

which the average edge length is specified to be $\lambda/10$ (λ being the free-space wavelength at the simulation frequency) is used. For generating the surface meshes, a commercially available meshing package (MSC-PATRAN) was used. This mesher is capable of generating two different surface meshes as depicted in Fig. 2.11 for a circle. The isoparametric mesh shown in Fig. 2.11 (a), generates fairly smooth, structured meshes. However, depending on how curved the surface is, some of the generated

elements may be severely distorted, making the mesh unsuitable for numerical solutions due to matrix ill-conditioning. On the other hand, the paver mesh (shown in Fig. 2.11 (b)) generates an unstructured mesh and is rid of the limitations of the isoparametric mesh. Below, we provide some numerical solutions using both kinds of meshes for the same geometry, and demonstrate that both meshes provide comparable accuracy. The first test geometry is a PEC sphere of radius 1 m, as depicted

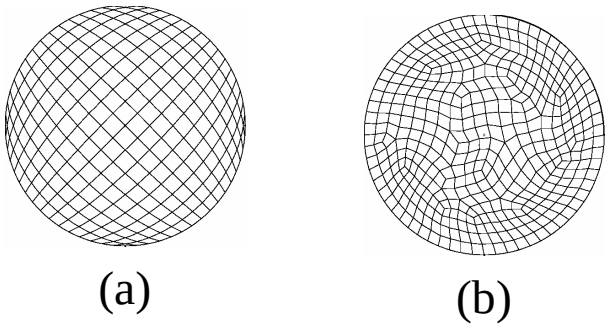


Figure 2.11. Two different meshing methods for a circle: (a) Isoparametric mesh, (b) Paver mesh.

in Fig. 2.10 (a). This is a unique 3-dimensional geometry for which an analytical solution (Mie series) exists. Hence, it provides a reference solution to be used for evaluating the numerical solution error. The frequency of the incident electromagnetic field is 300 MHz. The bistatic RCS results for both polarizations of the incident

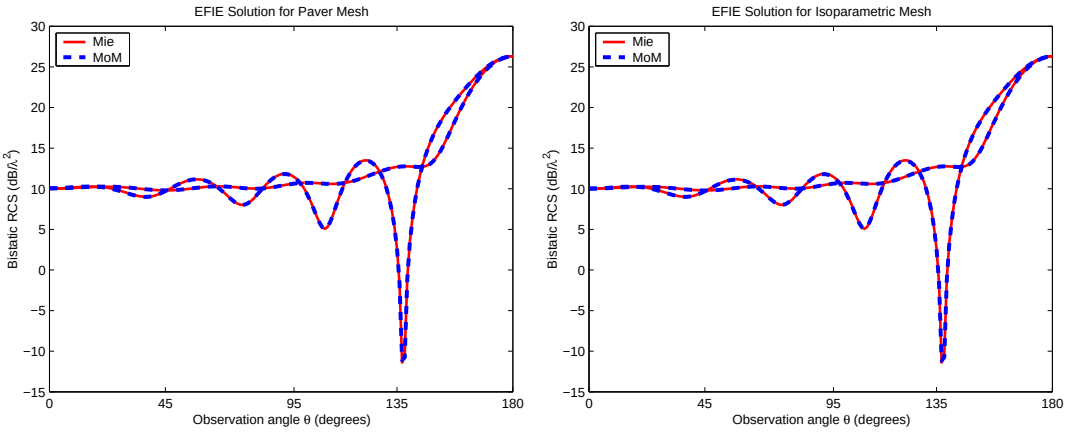


Figure 2.12. Bistatic scattering results for the sphere using the EFIE.

field using the paver and isoparametric meshes are shown in Fig. 2.12. Owing to the

high fidelity in geometry modeling, the computed results are in very good agreement with the analytical Mie solution. This level of accuracy is unattainable using lower order geometry modeling such as flat triangular elements as given in [2]. A comparison of several different modeling techniques was given in [6]. To prove the convergence of the method with respect to increasing mesh density, the root-mean-square (RMS) error in the RCS solution is given in Table 2.1. As can be observed, there is very

	paver	paver	paver	iso	iso	iso
Discretization	0.3 λ	0.2 λ	0.1 λ	0.3 λ	0.2 λ	0.1 λ
Matrix size	192	486	2118	192	423	1728
θ-polarization	16.05 %	4.01 %	0.39 %	16.58 %	5.26 %	0.56 %
ϕ-polarization	18.00 %	4.73 %	0.43 %	18.31 %	5.77 %	0.65 %

Table 2.1. RMS percent error in RCS results for the sphere for different mesh densities (EFIE solution).

little difference between using a paver mesh or an isoparametric mesh as long as the nominal sampling rate of $\lambda/10$ is used. At this sampling rate, the RCS results for both polarizations agree with the Mie solution to within less than 1%.

In the solution process, a conjugate gradient squared (CGS) [18, 39] iterative solver was used. The CGS requires two matrix-vector products per iteration and was observed to provide the best convergence behavior among other iterative solvers. For the paver mesh (2118 unknowns), the EFIE convergence was achieved in 82 iterations for the θ-polarization (81 for ϕ-polarization) to reach a pre-specified error of 10^{-3}. To further speed-up the convergence and avoid ill-conditioning of the EFIE due to spurious solutions around internal resonances of the geometry, the combined field integral equation (CFIE) must be used. The CFIE is simply a linear combination of the EFIE and MFIE using a combination parameter α as CFIE = α EFIE + $(1-\alpha)\frac{i}{k\eta}$ MFIE. In Fig. 2.13, we present the RCS results for the same setup using CFIE with $\alpha = 0.5$ as the combination parameter. Also, the RMS percent error data for the CFIE solution is given in Table 2.2. The CFIE achieved convergence within 15 iterations, significantly less than those required for the EFIE, for both polarizations

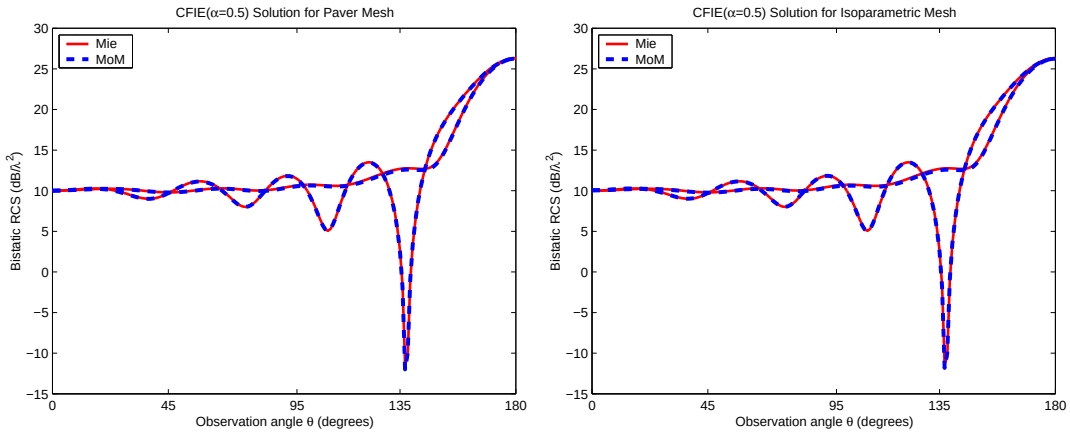


Figure 2.13. Bistatic scattering results for the sphere using the CFIE with ($\alpha = 0.5$).

	paver	paver	paver	iso	iso	iso
Discretization	0.3 λ	0.2 λ	0.1 λ	0.3 λ	0.2 λ	0.1 λ
Matrix size	192	486	2118	192	423	1728
θ-polarization	16.04 %	5.08 %	0.82 %	16.01 %	5.70 %	1.00 %
ϕ-polarization	16.26 %	4.23 %	0.67 %	16.46 %	5.05 %	0.82 %

Table 2.2. RMS percent error in RCS results for the sphere for different mesh densities (CFIE solution).

and for an error of less than 10^{-3}. In Fig. 2.14, we depict the RCS results using the MFIE. The MFIE is also prone to spurious solutions and unlike the EFIE, the MFIE fails even to generate correct far-field results. The results given in Fig. 2.14 are provided for completeness and the MFIE will not further be considered for the other test geometries.

The next test geometry is a cube of side length 1 m. For the cube, there is no advantage in using curvilinear elements. Nevertheless, since flat facets can be modeled equally well using curvilinear elements, they will be used for the cube as well. This geometry was chosen as a test case for representing targets with sharp edges. The bistatic RCS of the cube for both incident field polarizations using the EFIE and the CFIE are given in Fig. 2.15. Since these is no analytical solution for this problem, we depict two solutions for two different sampling rates. The agreement for both integral equations is excellent and the RMS error between the two solutions

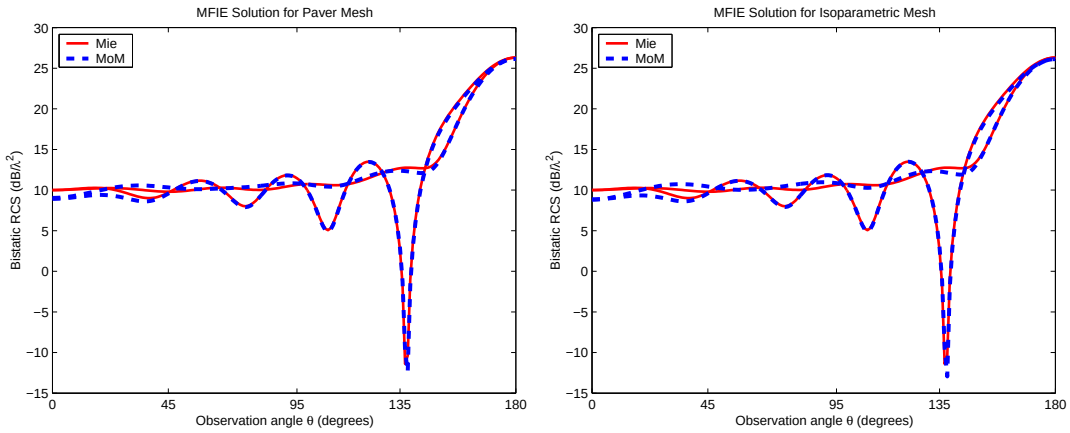


Figure 2.14. Bistatic scattering results for the sphere using the MFIE

are $0.38\% - 0.37\%$ for the EFIE and $0.48\% - 0.53\%$ for the CFIE. Nominal sampling

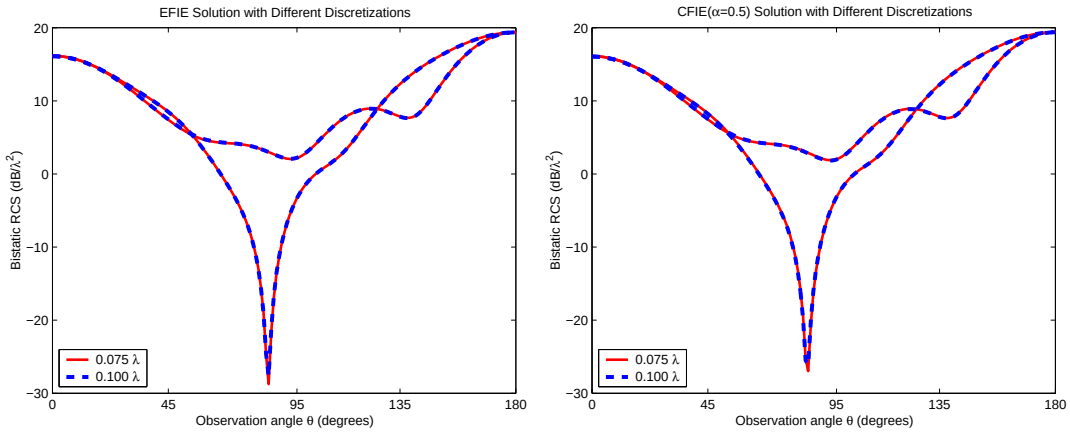


Figure 2.15. Bistatic scattering results for the cube using the EFIE and the CFIE ($\alpha = 0.5$).

resulted in 1200 unknowns and convergence was achieved within $137 - 102$ iterations for the EFIE and 13 iterations for the CFIE. The over-sampled mesh (at 0.075λ) generated 2028 unknowns and the EFIE converged in 116 iterations as compared to 14 CFIE iterations.

The cylinder is the third test geometry to be considered here. This geometry requires both flat and curved elements for modeling. The RCS results for the cylinder are given in Fig. 2.16. Again, the convergence of the RCS curves with increasing mesh density is excellent. The difference between the two solutions with a paver mesh are $0.31\% - 0.34\%$ and $0.47\% - 0.50\%$ for the EFIE and CFIE, respectively.

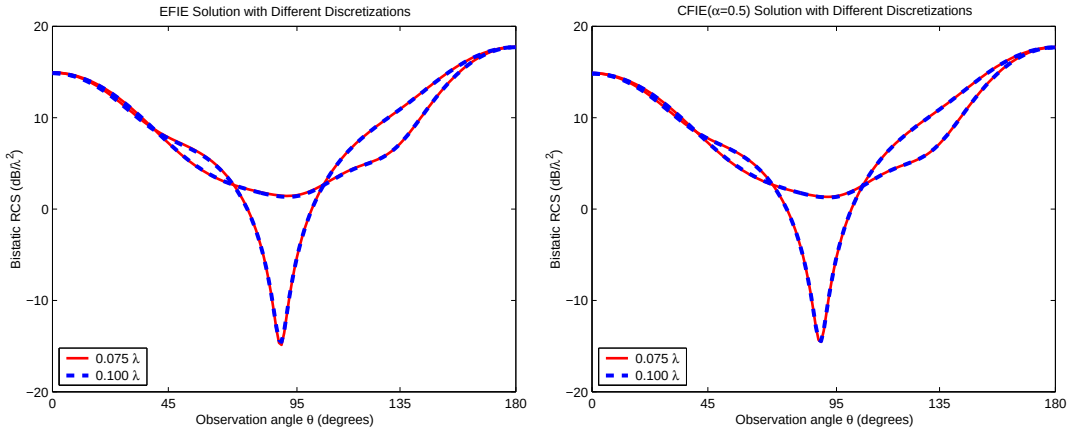


Figure 2.16. Bistatic scattering results for the cylinder using the EFIE and the CFIE ($\alpha = 0.5$).

For an isoparametric mesh, the pertinent data is $0.30\% - 0.29\%$ for the EFIE and $0.36\% - 0.42\%$ for the CFIE. We must note here that for the over-sampled mesh, the difference between the EFIE and the CFIE solutions are $0.71\% - 0.67\%$.

As a fourth test geometry, we consider an open-ended cylinder (cylinder with top cap open). Since it does not form a closed surface, we can only use the EFIE to solve this problem. The RCS results are given in Fig. 2.17. The agreement between the

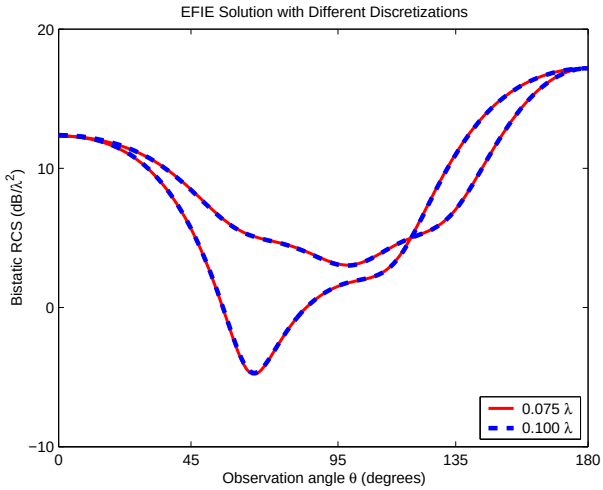


Figure 2.17. Bistatic scattering results for the open cylinder.

two mesh densities is excellent, indicating the convergence of the RCS curves. It is also interesting to note the effect of opening the top face of the cylinder on the RCS signature. The back-scattering (at $\theta = 0$) is reduced by 3 dB on removing the top

cover of the cylinder.

Fig. 2.18 depicts the bistatic RCS of the pyramid shown in Fig. 2.10 (e), our fifth geometry. Two different results are given to ensure mesh convergence. As compared

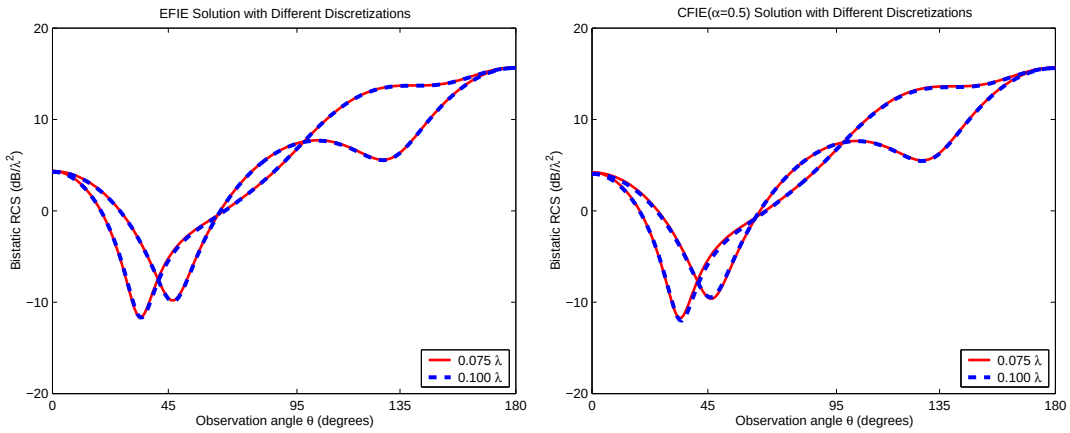


Figure 2.18. Bistatic scattering results for the pyramid using the EFIE and the CFIE ($\alpha = 0.5$).

to the closed cylinder, there is more than 10 dB reduction in the back-scatter return. The paver and isoparametric meshes resulted in 704 and 960 unknowns at $\lambda/10$ sampling and 1222 and 1766 unknowns at $\lambda/10$ sampling, respectively. The EFIE converged in 69 CGS iterations and the CFIE converged in 15 CGS iterations for the 704 unknown problem. For the over-sampled case (1766 unknowns), $145 - 142$ EFIE iterations were executed as compared to 24 CFIE iterations.

Our sixth and final test geometry is from the Electromagnetic Code Consortium (EMCC). The ogive geometry has a curved elongated body and two sharp tips. The sharp tips present a modeling challenge and consequently the geometry is well-suited for validating electromagnetic computer codes being developed by research centers and government agencies using different solution approaches. The computed bistatic RCS results are shown in Fig. 2.19. The dynamic RCS range of the curves is about 50 dB and the back-scattering return is clearly very low. In spite of this large dynamic range, the predicted results for both polarizations agree very well for the two mesh densities. The EFIE convergence was achieved in $313 - 363$ iterations

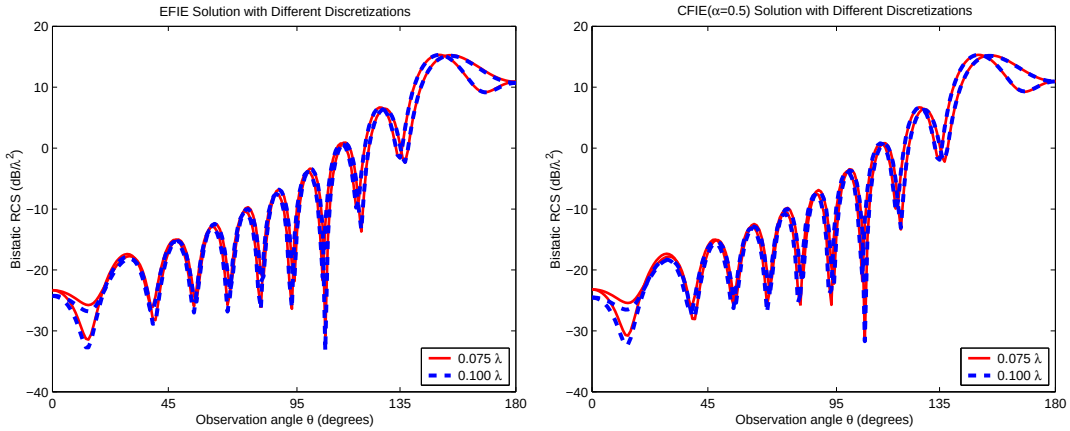


Figure 2.19. Bistatic scattering results for the ogive using the EFIE and the CFIE ($\alpha = 0.5$).

compared to $31-30$ iterations when the CFIE is used (a total of 8872 unknowns were used).

We conclude this chapter with color plots of induced surface current densities given in Fig.2.20 for all of the above test geometries. Usually, induced surface currents are

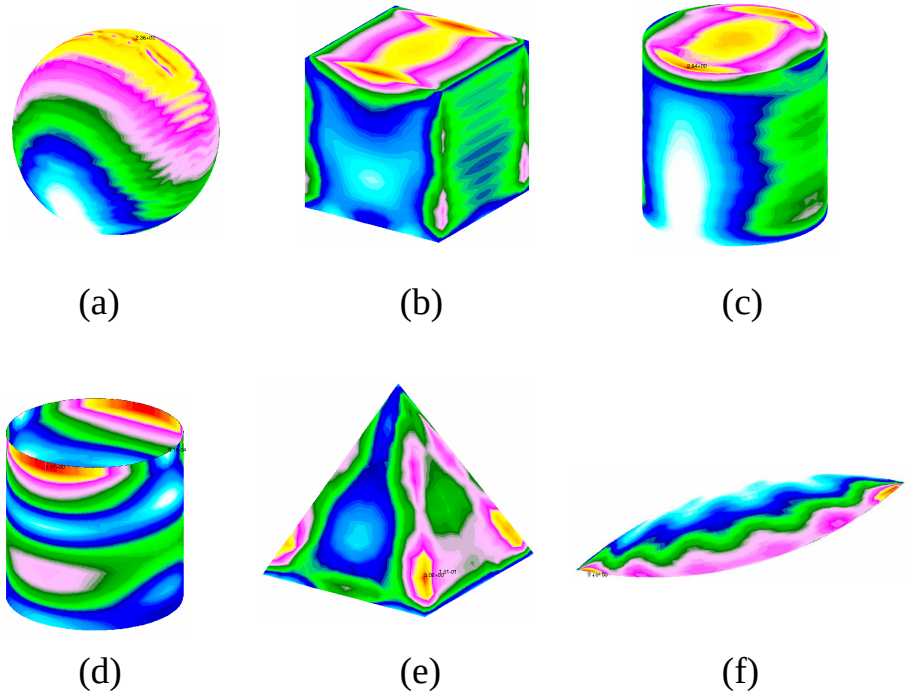


Figure 2.20. Induced surface currents on test targets.

not an observable in RCS measurements. However, computed distributions such as those given in Fig. 2.20 can facilitate the understanding of scattering mechanisms

and is necessary in designing low RCS targets.

For realistic structures with sharp edges and tips as well as curved surfaces, the problem size quickly reaches the limits of a given computing platform. As an example, the EFIE ogive system required 630 MBytes of computer memory and 45 minutes on a Pentium III processor. To remove the $O(N^2)$ memory and CPU bottlenecks of the conventional MoM procedure for electrically large problems, we outline the fast multipole and its multilevel implementation in the next chapter.

CHAPTER 3

Fast Multipole Method and Its Multilevel Implementation

As noted in the previous Chapter, standard MoM implementation quickly reaches its limits in terms of computer time and memory requirements for modern computing platforms. Even though better geometry modeling methods enables the solution of the same problem using less computer resources, fast and low memory solution methods must be employed in connection with iterative solvers to tackle electrically large real-life problems. The most time consuming step in any iterative algorithm, such as the conjugate gradient (CG), biconjugate gradient (BiCG), quasi-minimal residual (QMR), and generalized minimal residual (GMRES) routines [39], is the matrix-vector product and this is where the fast methods are focused on. Fast methods are often referred to as matrix compression algorithms, and k-space methods [15] were among the first such approaches to be employed with iterative solvers. Although k-space methods lead to $O(N \log N)$ memory and computational complexity, their application is restricted to systems/geometries which can be approximated with circulant matrices. Originally, this requirement could only be fulfilled using uniform discretizations of the integral equation. However, recently introduced fast integral methods such as the fast multipole method (FMM) [8, 13, 42, 43, 44, 45] and the adaptive integral method (AIM) [21, 46, 47, 48] are rid of restrictions for uniform discretization of the original geometries. These methods have been shown to deliver memory and CPU reduction down to $O(N^{3/2})$ or better. Windowed FMM [10], ray propagation fast multipole algorithm [9], fast far-field approximation [11], and mul-

tilevel FMM [13] can reduce the CPU time down to $O(N^{4/3})$ or even $O(N \log N)$. Early electromagnetic applications of FMM concentrated on pure integral equation approaches, but uses of FMM in the context of hybrid FE-BI formulations have also been reported [43, 45]. We will address the application of FMM within the context of FE-BI method using curvilinear volumetric elements in the next chapter.

AIM can be considered as the natural extension of the k–space methods and was introduced for arbitrary surface and volumetric scattering problems [46] and is especially the method of choice for planar BI surfaces. In this case, only two-dimensional FFT algorithms need to be employed and the method results in a low $O(N \log N)$ complexity. Thus, the speed-up of AIM is considerably better than that of FMM or multilevel FMM for planar boundaries, even for relatively small numbers of unknowns. However, for arbitrary three-dimensional geometries, AIM requires a three-dimensional grid and a consequent three-dimensional FFT using all the grid points. For surface geometries, most of the grid points are not used since they lie far from the actual surface. This redundancy is a severe limiting factor in the range of AIM applications. The remedy is to utilize FMM which is applicable to planar and non-planar surfaces as well as volumetric integral equations.

For an accurate numerical solution of the governing integral equations, the geometrical model of the problem must be meshed at around 10 basis functions per wavelength. The resulting square matrix system has a dimension N which is equal to the number of internal edges in the mesh, where the latter is proportional to the number of elements in the mesh.

As mentioned in the previous chapter, the storage of this matrix system requires $O(N^2)$ computer memory. A direct solution would then require $O(N^3)$ flops, as compared to an iterative solution which requires $O(N^2)$ flops per iteration. The cost of the iterative solution process comes from the matrix-vector multiplications in each iteration step. A closer examination of the matrix elements leads to faster and

more efficient algorithms that evaluate the matrix-vector product indirectly. In this chapter, we outline the fast multipole method (FMM) and its multilevel version in the context of curvilinear elements and conformal basis functions.

3.1 Fast Multipole Method

When the MoM matrix system is being solved iteratively, a search vector is generated at each iteration of the specific solver using the error from the previous iteration. Given a search vector representing the magnitudes of each basis function (i.e. a current distribution over the scatterer geometry), the product of that search vector with each row of the matrix is equivalent to computing the reaction between the field generated by that specific current distribution and each testing function. Each entry of the MoM matrix defined by the inner product $Z_{ji} = \langle \mathbf{t}_j, \mathcal{L}(\mathbf{j}_i) \rangle$ (where $\mathcal{L}$ represents either of the linear EFIE or MFIE operators) corresponds to the reaction between the field generated by the basis function $\mathbf{j}_i$ and the testing function $\mathbf{t}_j$.

The FMM relies on a mathematical manipulation of the free-space Green's function so that the reaction between the collective field of a group of basis functions and a group of testing functions can be evaluated at a lower computational cost by reusing the information already computed. Let's assume that the basis functions $\mathbf{j}_i, \mathbf{j}_{i+1}, ..., \mathbf{j}_{i+M}$ are in the near vicinity of each other and similarly the testing functions $\mathbf{t}_j, \mathbf{t}_{j+1}, ..., \mathbf{t}_{j+M}$ are in the near vicinity of each other (see Fig. 3.1). Furthermore, let's assume that these two groups are separated by a distance larger than the physical sizes of both groups. Computation of all the testing coordinates $\langle \mathbf{t}_j, \mathcal{L}(\mathbf{j}_i) \rangle$ of the radiated fields due to all basis function in the source group requires as many operations as the number of possible connections between each basis and testing pair, i.e. $O(M^2)$ flops. By introducing the spherical multipole expansion to represent the fields radiated by the basis group over the testing group, it is possible to reduce this operation count.

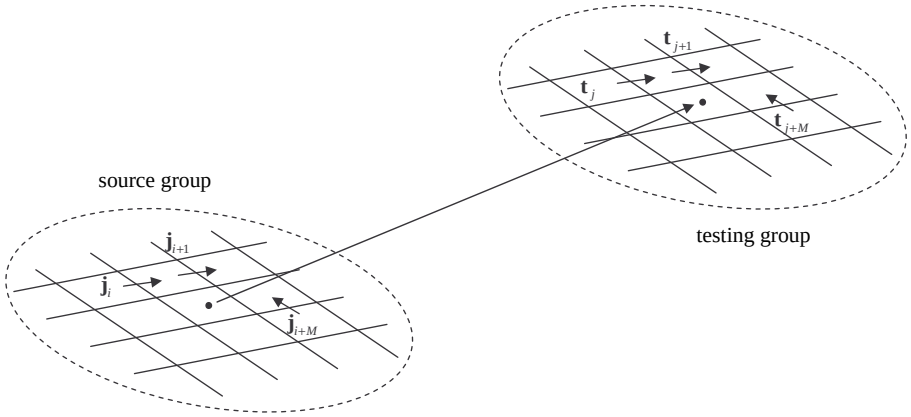


Figure 3.1. Groups of basis and testing functions.

The spherical multipole expansion is based on Gegenbauer's addition theorem [49]

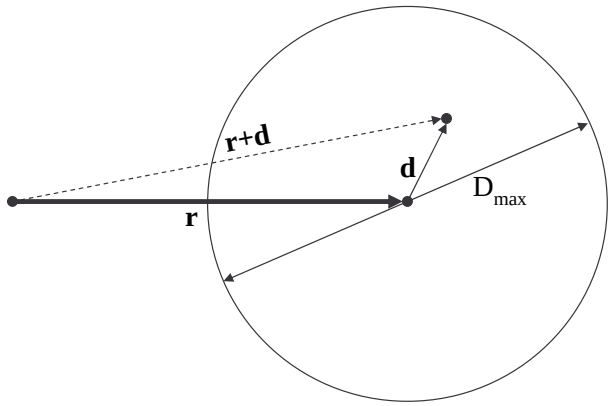


Figure 3.2. Vector definitions for Gegenbauer's addition theorem.

$$\frac{e^{ik|\mathbf{r}+\mathbf{d}|}}{|\mathbf{r}+\mathbf{d}|} = ik\sum_{l=0}^{\infty}(-1)^l(2l+1)j_l(kd)h_l^{(1)}(kr)P_l(\hat{d}\cdot\hat{r}) \tag{3.1}$$

in which j_l is the spherical Bessel function of order l, $h_l^{(1)}$ is the spherical Hankel function of the first kind and of order l, P_l is the Legendre polynomial of order l (see Fig. 3.2). This expansion is the backbone of the FMM algorithm. Using (3.1) and the plane wave expansion of the product

$$4\pi i^l j_l(kd)P_l(\hat{d}\cdot\hat{r}) = \int d^2\hat{k}e^{i\mathbf{k}\cdot\mathbf{d}}P_l(\hat{k}\cdot\hat{r}), \tag{3.2}$$

it can be shown that

$$\frac{e^{ik|\mathbf{r}+\mathbf{d}|}}{|\mathbf{r}+\mathbf{d}|} \approx \frac{ik}{4\pi}\int d^2\hat{k}e^{i\mathbf{k}\cdot\mathbf{d}}T_L(kr,\hat{k}\cdot\hat{r}), \tag{3.3}$$

where from (3.1)

$$T_L(kr, \hat{k} \cdot \hat{r}) = \sum_{l=0}^{L} i^l (2l+1) h_l^{(1)}(kr) P_l(\hat{k} \cdot \hat{r}) \tag{3.4}$$

in which the infinite upper limit of the summation was replaced by a finite integer L, i.e. the infinite sum was truncated up to the first $(L+1)$ multipole terms. The validity region of (3.1) is $|\mathbf{r}| > |\mathbf{d}|$. However, (3.1) would require many terms for $|\mathbf{r}| \approx |\mathbf{d}|$. In practice $|\mathbf{d}|$ is chosen small in order to use fewer terms in the multipole sum for a fast execution. Once the translation sum $T_L(kr, \hat{k} \cdot \hat{r})$ is computed for a given $\mathbf{r}$, (3.3) can be used to compute the field anywhere inside a sphere of radius $D_{max} < 2|\mathbf{r}|$ very quickly. Hence, the field generated by a source outside of the testing cluster, can be evaluated over all testing functions inside the testing cluster merely by employing (3.3) using the pre-computed translation operator [8].

For the more relevant vector construct in Fig. 3.3 it can be shown that (3.3) allows

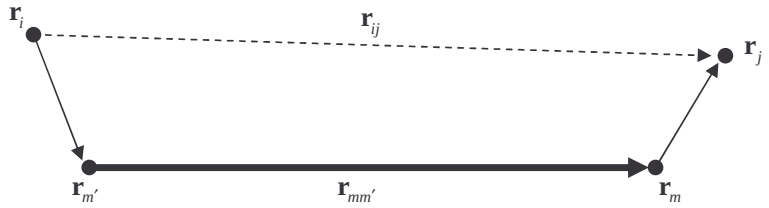


Figure 3.3. Vector definitions for FMM expansion.

the computation of the scalar Green's function using

$$\frac{e^{ikr_{ji}}}{r_{ji}} \approx \frac{ik}{4\pi} \int d^2\hat{k} e^{i\mathbf{k}\cdot\mathbf{r}_{jm}} T_L(kr_{mm'}, \hat{k} \cdot \hat{r}_{mm'}) e^{-i\mathbf{k}\cdot\mathbf{r}_{im'}}. \tag{3.5}$$

We can compute the fields of all basis functions in a source group over all testing functions in the observation group via (3.5) by reusing the pre-computed $T_L(kr_{mm'}, \hat{k} \cdot \hat{r}_{mm'})$ provided $r_{mm'} > (r_{im'} + r_{jm})$.

Using (3.5), it is straightforward to write the MoM matrix elements (for the EFIE)

$$Z_{ji} = \int_s d\mathbf{r} \mathbf{t}_j(\mathbf{r}) \cdot \int_s d\mathbf{r}' \overline{\mathbf{G}}(\mathbf{r}, \mathbf{r}') \cdot \mathbf{j}_i(\mathbf{r}') \tag{3.6}$$

as follows:

$$Z_{ji} \approx \frac{ik}{4\pi} \int d^2\hat{k} \mathbf{V}_{jm}(\hat{k}) \cdot T_L(kr_{mm'}, \hat{k} \cdot \hat{r}_{mm'}) \mathbf{V}^*_{im'}(\hat{k}), \tag{3.7}$$

where

$$\begin{aligned} \mathbf{V}_{im'}(\hat{k}) &= \int_S d\mathbf{r}' e^{i\mathbf{k}\cdot\mathbf{r}_{im'}} \left[\bar{\mathbf{I}} - \hat{k}\hat{k}\right] \cdot \mathbf{j}_i(\mathbf{r}_i), \\ \mathbf{V}_{jm}(\hat{k}) &= \int_S d\mathbf{r} e^{i\mathbf{k}\cdot\mathbf{r}_{jm}} \left[\bar{\mathbf{I}} - \hat{k}\hat{k}\right] \cdot \mathbf{t}_j(\mathbf{r}_j) \end{aligned} \tag{3.8}$$

and

$$T_L(kr_{mm'}, \hat{k} \cdot \hat{r}_{mm'}) = \sum_{l=0}^{L} i^l (2l+1) h_l^{(1)}(kr_{mm'}) P_l(\hat{k} \cdot \hat{r_{mm'}}) \,. \tag{3.9}$$

We remark that in FMM implementation, Z_{ji} are computed using (3.6) for nearby testing and source basis and (3.7) is used otherwise. The fast implementation of (3.7) between far groups of source and testing basis functions is the crux of the FMM.

In (3.7) we used the fact that $\left[\bar{\mathbf{I}} - \hat{k}\hat{k}\right] \cdot \left[\bar{\mathbf{I}} - \hat{k}\hat{k}\right] = \left[\bar{\mathbf{I}} - \hat{k}\hat{k}\right]$ to get a symmetric equation. The terms in (3.8) are known in FMM nomenclature as the signature functions. These are pre-computed and symmetry provides for storage savings since we only need to compute (3.8) for the basis functions (since in Galerkin's method, the testing functions are chosen to be equal to the testing functions). However, this is only true for the symmetric EFIE operator. When the MFIE operator is considered, then

$$Z_{ji} = -\int_s d\mathbf{r} \mathbf{t}_j \cdot \left[\frac{1}{2}\mathbf{j}_i(\mathbf{r}) + \oint_s d\mathbf{r}' \hat{n}' \times \mathbf{j}_i(\mathbf{r}') \times \nabla g(\mathbf{r}, \mathbf{r}')\right] \tag{3.10}$$

and the signatures for the basis and testing functions take the forms

$$\begin{aligned} \mathbf{V}_{im'}(\hat{k}) &= \int_s d\mathbf{r}' e^{i\mathbf{k}\cdot\mathbf{r}_{im'}} \left\{\hat{k} \times [\hat{n}' \times \mathbf{b}_i(\mathbf{r}_i)]\right\}, \\ \mathbf{V}_{jm}(\hat{k}) &= \int_s d\mathbf{r} e^{i\mathbf{k}\cdot\mathbf{r}_{jm}} \left[\bar{\mathbf{I}} - \hat{k}\hat{k}\right] \cdot \mathbf{t}_j(\mathbf{r}_j). \end{aligned} \tag{3.11}$$

Here, the identity $\left[\bar{\mathbf{I}} - \hat{k}\hat{k}\right] \cdot \left[\hat{k} \times \mathbf{a}\right] = \hat{k} \times \mathbf{a}$ was used so that we can use the same EFIE signatures of the testing functions.

The translation operator, being only dependent on the pre-chosen vector $\mathbf{r}_{mm'}$ between the points m and m', is key to reducing the $O(N^2)$ complexity of the MoM

matrix-vector product in the iterative solver. By grouping the basis functions into a pre-specified number of clusters and by reusing the translation operator to compute the interactions of the basis and testing functions, the $O(N^2)$ complexity can be reduced down to $O(N^\gamma)$ where $\gamma < 1.5$. For the conventional FMM algorithm [8] $\gamma = 1.5$, but γ can be reduced further and is close to unity for the multilevel version of FMM. As an example, Song et al. [13] applied a nested grouping strategy along with a flat triangular discretization and Rao-Wilton-Glisson (RWG) [2] basis functions and introduced a multilevel FMM algorithm with $O(N \log N)$ computational complexity and $O(N)$ memory requirement. This low complexity has been achieved by forming a multilevel grouping of the basis functions (by grouping smaller groups in larger groups) and reusing the translation operators computed between groups at each level, as will be outlined in the next section. We note here that there is no restriction in the FMM algorithm on the type of surface elements and sub-sectional basis functions used. Curvilinear surface elements and conformal basis functions as defined in Chapter 1 are used in our specific FMM implementations. As mentioned before, this curved surface modeling has the advantage of reducing geometry modeling error in the solution procedure substantially. This aspect also allows us to investigate the error performance of the FMM algorithm more accurately since the error in the final result is dominated by the error introduced by the FMM approximation rather than geometry modeling error, as is the case in flat triangular meshes and RWG basis functions. Also, on each element there are 4 basis functions associated with 4 edges, but only 2 of them are independent since adjacent elements are paired to form a basis function. That is, curved quadrilateral elements are associated with 2 degrees of freedom per element, whereas flat triangles are associated with 1.5 degrees of freedom on each element (3 basis functions are associated with the 3 edges on a single triangular element). Quadrilateral elements also allows for better representation of the induced surface current density on each element.

To take advantage of the expansion (3.7), the basis functions are grouped into M clusters where $r_{mm'}$ refers to the center-to-center distance between source group m' and the testing group m. Due to this *a priori* grouping, $T_L(kr_{mm'}, \hat{k} \cdot \hat{r}_{mm'})$ is independent of the specific source/testing function pair. It only depends on the inter-distances between the chosen groups and is pre-computed prior to the iterative solver execution. Also, when the groups are well-separated we need a few terms in the translation sum. Consequently, the spectral integral in (3.3) can be numerically evaluated quickly using small number of integration points.

A most important parameter in the FMM implementation is the number of terms kept for the evaluation of T_L in (3.9). This parameter, commonly denoted as L [8], is semi-empirically chosen as

$$L = kD_{max} + \alpha_L \ln(kD_{max} + \pi) \tag{3.12}$$

where D_{max} is the maximum diameter of the clusters and α_L is an accuracy control parameter. For $\alpha_L = 5$, the resulting value of L is reported to provide single precision accuracy in evaluating T_L; for $\alpha_L = 10$, double precision is obtained in evaluating T_L [8]. Typically, however, it has been reported [13] that setting $\alpha_L = 1$ gives acceptable accuracy for RCS computations. However, for geometries producing poorly conditioned matrices (e.g. scatterers with sharp corners and/or fine detail), any computation error in the FMM implementation is amplified by the condition number of the system and is reflected to the final solution. When using an iterative solver, keeping α_L small will result in faster execution time but may produce large error in the final solution. For such problems, it is a good practice to keep the FMM approximation very accurate in order to get a correct solution at the expense of slower solution time. As discussed in Appendix C, another remedy to this problem is to use a low-cost preconditioner.

The spectral integral in (3.3) is discretized using L points over the θ angular sector (Gaussian quadrature) and $2L$ points in the ϕ sector (trapezoidal rule). This

choice of integration points and associated integration weights ensure integration of spherical harmonics up to order $2L$ exactly [8]. With the numerical evaluation of the spectral integral, (3.7) can be rewritten as

$$Z_{ji} \approx \frac{ik}{4\pi} \sum_{\hat{k}_p} w_{\hat{k}_p} \mathbf{V}_{jm}(\hat{k}_p) \cdot T_L(k_p r_{mm'}, \hat{k}_p \cdot \hat{r}_{mm'}) \mathbf{V}^*_{im'}(\hat{k}_p) \tag{3.13}$$

where $\hat{k}_p$ denotes the $2L^2$ spectral integration points and $w_{\hat{k}_p}$ denotes the associated integration weights.

Similarly, the signature functions $\mathbf{V}_{im'}(\hat{k}_p)$ associated with all basis functions are pre-computed and stored for all discrete $\hat{k}$ directions. These pre-computed signature functions actually map each basis function in real space onto a signature in k-space, with corrections to the phase centers for each cluster. However, unlike the Green's function in real-space, the translation operator in the k-space is a diagonal operator. That is, the value of the radiation field of each basis function for a specific $\hat{k}$ direction only depends on the value of the signature of the basis functions for the same $\hat{k}$ direction. Furthermore, the collective signature function of a group of basis functions

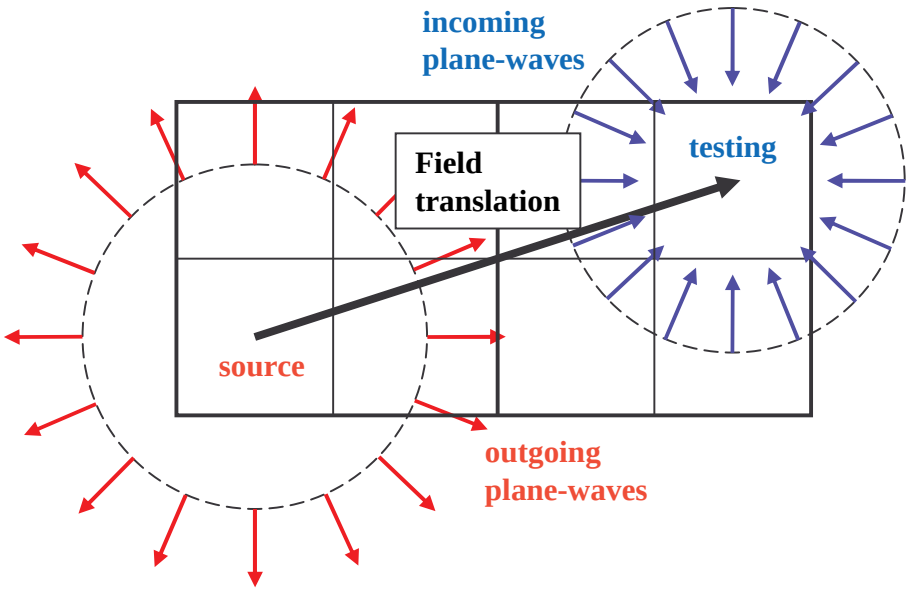


Figure 3.4. Illustration of single level interactions between source and testing clusters in FMM

can simply be constructed by adding the signatures of all basis functions inside that

group. These two observations form the foundation of the FMM and are exploited to obtain solution speed-up and memory reduction.

The numerical implementation of the indirect matrix-vector product computation consists of three successive steps:

Step 1. The radiation signatures of all groups are formed by summing the signatures of all basis functions within each group

$$\mathbf{V}_{m'}(\hat{k}_p) = \sum_{\mathbf{j}_i \in G_{m'}} \mathbf{V}^*_{im'}(\hat{k}_p) \tag{3.14}$$

where $G_{m'}$ denotes the set of all basis functions in the source group m'. This is called the *aggregation* step.

Step 2. Provided the testing and basis groups are well seperated, group signatures of all source groups are then translated over to the testing group via the operation

$$\mathbf{V}_m(\hat{k}_p) = \sum_{m' \in F_{m'}} T_L(k_p r_{mm'}, \hat{k}_p \cdot \hat{r}_{mm'}) \mathbf{V}_{m'}(\hat{k}_p) \tag{3.15}$$

where $F_{m'}$ is the set of source groups in the far-zone of the testing group m. Equation (3.15) basically transforms the outgoing plane wave signatures of the source groups onto an incoming plane wave signature on the testing group. This plane wave or spectral transformation is evocative of the Fourier transform process done for uniformly gridded testing and basis functions.

Step 3. The third and final sweep is the *disaggregation* used to translate the field signatures over to the testing functions. We must note here that (3.7) is only valid for far source and testing pairs. Hence, the interactions of basis and testing functions in close proximity of each other must be done using the original MoM procedure. With the disaggregation step, the field over all testing functions inside the testing domain is computed via

$$\sum_i Z^{far}_{ji} x_i \approx \frac{ik}{4\pi} \sum_{\hat{k}_p} w_{\hat{k}_p} \mathbf{V}_{jm}(\hat{k}_p) \cdot \mathbf{V}_m(\hat{k}_p). \tag{3.16}$$

Clearly, the smaller the groups, the less multipole terms would be required. For example, for a maximum group size of $D_{max} = \lambda$, using (3.12) with $\alpha_L = 2$ would result in $L+1 = 12$ terms ($L = kD_{max}+2\ln(kD_{max}+\pi) = 10.77$), whereas for $D_{max} = \lambda/2$, we only need $L + 1 = 8$ terms. With these numbers, the spectral integral would require $2L^2 = 242$ samples for the larger group size as compared to 98 samples for the latter. However, to form smaller groups, we need to increase the number of groups and this increases the computation of the translation operations. If N basis functions are grouped into M groups, each group will consist of N/M basis functions. The number of near groups for each source group will be assumed to be constant. Hence, the near-field matrix will have $O(N^2/M)$ non-zero entries representing the near interactions of the basis and testing functions. At the aggregation and disaggregation sweeps, the number of flops is proportional to $N \times 2L^2$. The translation step requires $M^2 \times 2L^2$ flops. Assuming that L is proportional to the maximum electrical diameter of the groups kD_{max}, and that the number of basis functions in each group is proportional to N/M, we can conclude that $L^2 \propto N/M$. Hence, the complexities of the aggregation and disaggregation steps are proportional to $O(N^2/M)$, and that of the translation step is $O(NM)$.

Hence, the overall complexity of the FMM matrix-vector product is governed by two competing complexities with the overall complexity given by $O(N^2/M + NM)$. This order can be minimized by choosing $M = \sqrt{N}$. The resulting algorithm will then have $O(N^{3/2})$ computational complexity. On top of the CPU savings of FMM, since not all entries of the MoM matrix are needed, we will also have memory savings. Namely, only $O(N^2/M)$ non-zero entries must be computed and stored. However, we still need to compute and store the signature functions and the translation operator. The storage requirement of the signature functions is $O(2L^2N) = O(N^2/M)$ and that of the translation operator is $O(2L^2M^2) = O(NM)$. The sum of these requirements is also minimized for $M = \sqrt{N}$ and the total storage complexity of the solution method

is reduced down to $O(N^{3/2})$.

The above CPU estimates are asymptotic in the sense that they represent values which are approached for very large N. The actual efficiency of the implementation depends on the choices of various parameters. These choices control the constant in front of the asymptotic behavior of the CPU requirements. However, choices of the FMM parameters for faster implementation inevitably lead to less accurate answers. Clearly, keeping L as small as possible reduces the CPU time but not the order. That is, if the CPU time is given by $CN^{3/2}$ for the matrix vector product computation ($C = constant$), lowering L implies smaller value for the constant. Lower C values imply that the CPU time crossover point between standard MoM and FMM implementations occurs for lower values of N.

Another key area that determines FMM efficiency is that of element grouping. Here, for the single level FMM implementation, we used the k-means algorithm [50] to form the clusters. This algorithm aims to minimize a defined error function for the purpose of achieving a fairly uniform set of groups. The specific error function used by the k-means algorithm minimizes the sum of the distances between the cluster centers and the centers of the basis functions in each cluster. If the resulting clustering is not uniform, the FMM implementation will be inefficient due to the presence of very large or very small clusters. This affects the computation time due to the spectral integral since the number of terms used in the translation sum will depend on the size of the largest cluster. Although the k-means algorithm provides a fairly uniform grouping, it is always unstructured as shown in Fig. 3.5. As will be discussed in the multilevel implementation of the FMM, a structured grouping, such as one that consists of cubic groups having translational symmetry, can provide further savings in the storage of the translation operator since some of the translation operators will be identical due to translational symmetry.

In the FMM implementation, the approximation (3.7) is only used for computing

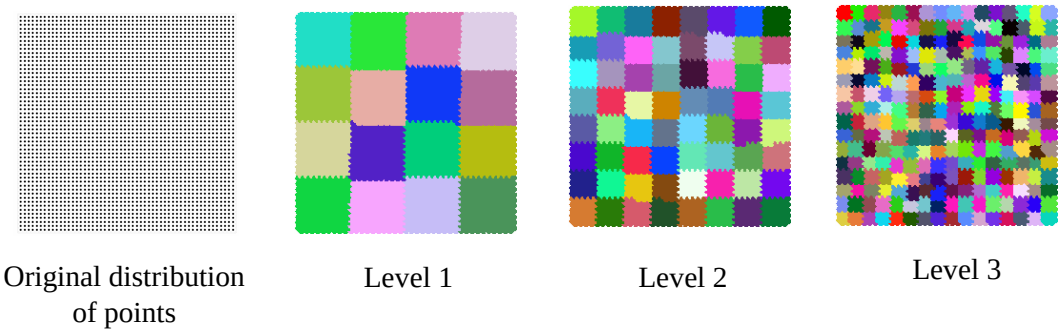


Figure 3.5. 3-levels of k-means clustering example.

the interactions of elements between well-separated clusters. As mentioned earlier, the functions within a cluster and within adjacent clusters are interacted without the FMM approximation. For greater accuracy, elements between clusters that are within a threshold distance can be computed without approximation as well. Hence, we can introduce an additional parameter Δ for controlling the transition distance between the exact evaluation and the FMM approximation. Specifically, if $r_{mm'}$ is the distance between clusters m and m', the FMM approximation is involved only if

$$r_{mm'} > \Delta(D_{max}) \quad \text{and} \quad kr_{mm'} > L \tag{3.17}$$

where D_{max} denotes the maximum diameter of the source and testing clusters. The second condition, namely $kr_{mm'} > L$ is due to unavoidable error in evaluating the translation sum, since for small $kr_{mm'}$ the Hankel functions in (3.4) tend to have very large values and this renders the calculation of T_L numerically unstable. Clearly, if $\Delta = 1$, then even neighboring clusters will be considered as far zone (and thus FMM will be used for their interactions). For $\Delta = 1.5$, the matrix elements in the neighboring (touching) clusters will be evaluated without approximation (see Fig.3.6). Basically, higher Δ implies that more interactions will be carried out without approximation, thus increasing the near-field matrix size and the CPU time for computing near-field matrix-vector products. Also, the requirement that $kr_{mm'} > L$ leads to more terms in the near-field matrix when the same problem mesh is used at lower frequencies. To further decrease the storage requirement of the near-field matrix, the

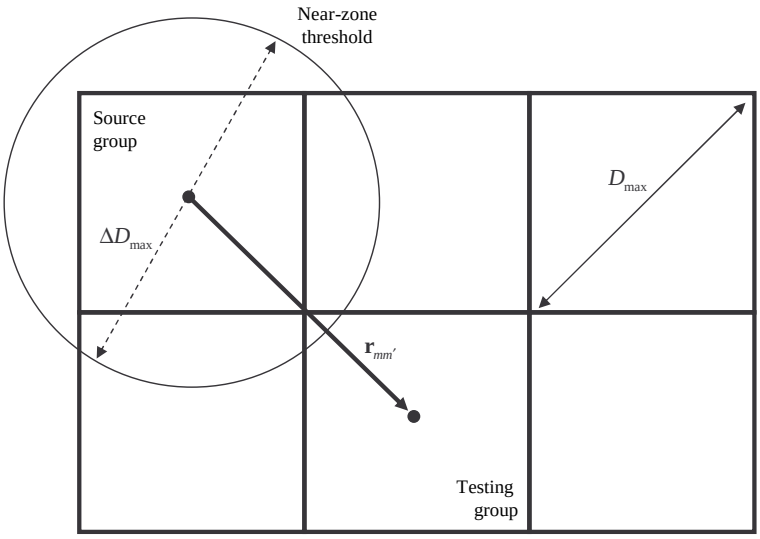


Figure 3.6. Illustration of the near-zone threshold in FMM.

signature functions, and translation operations as well as to accelerate the execution of the matrix-vector product, a multilevel nested strategy for the three governing FMM operations must be adopted.

3.2 Multilevel Fast Multipole Method

In this section, we first outline a multilevel grouping scheme to be used in the MLFMM implementation of the surface integral equation formulations. The tree structure of the grouping and the necessary information relating to the implementation of the MLFMM matrix-vector product are outlined here. The implicit assumption here is that each basis function is associated with a point in space (e.g. the mid-point of the edge associated with the basis function) and that clustering is carried out in terms of points in space rather than the actual basis functions themselves (see Fig.3.7).

Referring to Fig. 3.8, the first step in forming the multilevel clustering is to enclose the given geometry in a cubic box. To do so, the maximum dimension d_0^{max} of a given distribution of points (associated with the basis functions) in space and the center of gravity of the distribution must be calculated. This is called the 0^{th} level cluster.

For level 1, clusters are formed by subdividing the 0^{th} level cube into 8 smaller

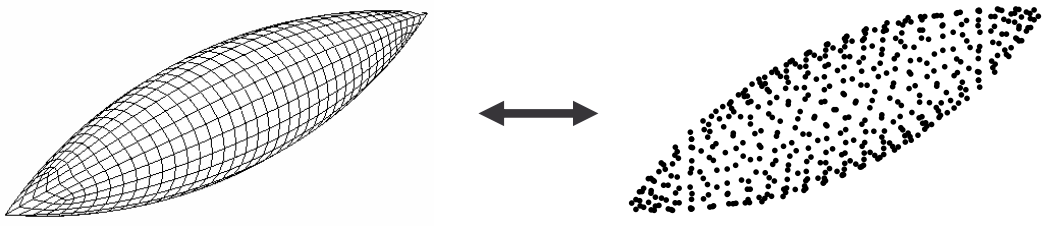


Figure 3.7. Association of basis functions with points in space (the mid-points of the edges in the surface mesh).

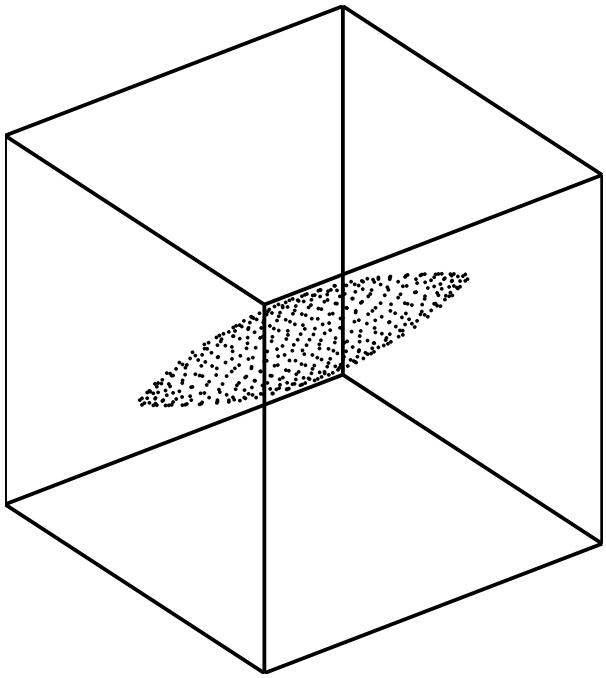


Figure 3.8. Zeroth level cube enclosing the geometry.

cubes. These 8 clusters are called the children, and the original cluster is called the parent. Likewise, each cluster in level 1 can be subdivided in a similar manner to form the clustering for level 2, and so on. Hence, at each level, the side length of the cubes are $d_l^{max} = d_0^{max}/2^l$, where l denotes the level number. The maximum linear dimension of the clusters for the l^{th} level is hence $D_l^{max} = \sqrt{3}d_l^{max}$. This subdivision is carried out until the $\mathcal{N}^{th}$ level where $D_{\mathcal{N}}^{max} \leq \lambda/\pi$, λ being the wavelength at the frequency of operation. We must note here that only non-empty cubes are considered when forming the clustering. If a cube at level l is empty, it is just discarded. This tree structured multilevel grouping (by forming 8 children from each parent) is referred

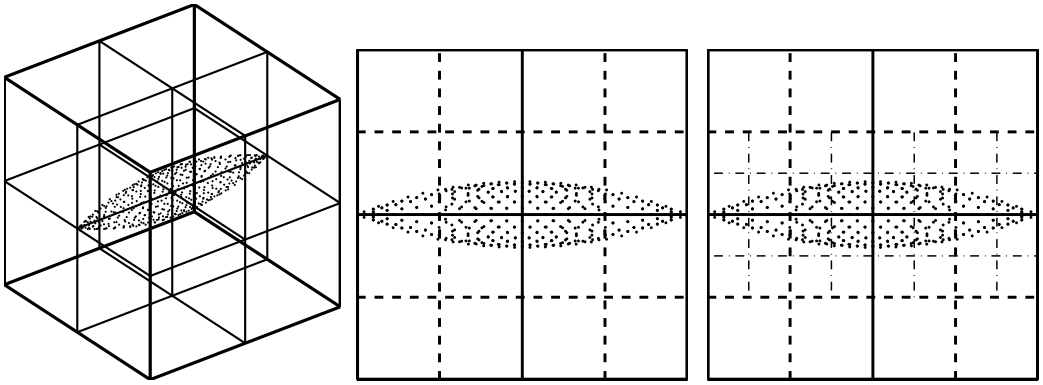


Figure 3.9. First, second, and third level grouping example.

to as the *oct-tree* clustering process. At the finest level $\mathcal{N}$, each basis function is associated with a cluster and the parent-children relationship of all levels is stored in the oct-tree structure. We note here that since the electrical sizes of clusters at each level are different, the number of multipoles that will be used to compute the translation operators at each level will be different. This impacts CPU cost and memory savings, and to clarify this, we need to look into a simple two-level formulation for the indirect evaluation of the interaction between a source point and an observation point.

3.3 Multilevel Fast Multipole Method Formulation

Consider the two level vector construct shown in Fig. 3.10. Introducing the decom-

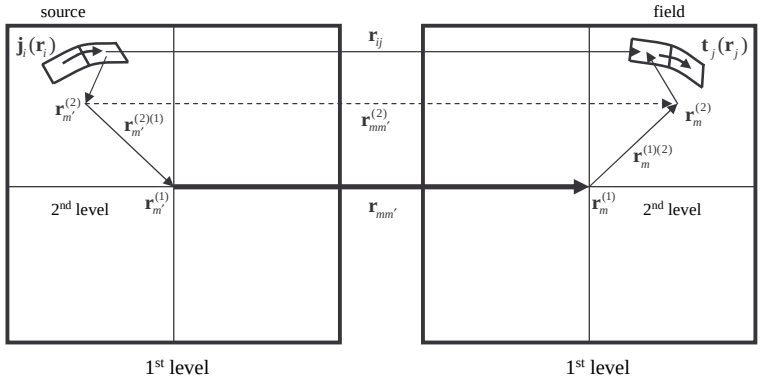


Figure 3.10. Two level FMM construct for a general multilevel case.

position,

$$\mathbf{r}_{ji} = \mathbf{r}_j - \mathbf{r}_i = \mathbf{r}^{(2)}_{m'i} + \mathbf{r}^{(1)(2)}_{m'} + \mathbf{r}_{mm'} + \mathbf{r}^{(2)(1)}_{m} + \mathbf{r}^{(2)}_{jm}, \tag{3.18}$$

we proceed to expand the scalar Green's function in the usual way as

$$\frac{e^{ikr_{ji}}}{r_{ji}} = \frac{ik}{4\pi} \int d^2\hat{k} e^{i\mathbf{k}\cdot(\mathbf{r}^{(2)(1)}_m + \mathbf{r}^{(2)}_{jm})} T_L(kr_{mm'}, \hat{k}\cdot\hat{r}_{mm'}) e^{-i\mathbf{k}\cdot(\mathbf{r}^{(2)}_{im'} + \mathbf{r}^{(2)(1)}_{m'})} \tag{3.19}$$

where again

$$T_L(kr_{mm'}, \hat{k}\cdot\hat{r}_{mm'}) = \sum_{l=0}^{L} i^l (2l+1) h^{(1)}_l(kr_{mm'}) P_l(\hat{k}\cdot\hat{r}_{mm'}) \tag{3.20}$$

is the translation operator, and the superscripts (2) and (1) appearing on the $\mathbf{r}$ vectors denote the grouping levels, (1) being the coarser level and (2) the finer. Hence, the dyadic Green's function can be written as

$$\overline{\mathbf{G}}(\mathbf{r}_j, \mathbf{r}_i) = \frac{ik}{4\pi} \int d^2\hat{k} \left[\mathbf{I} - \hat{k}\hat{k}\right] e^{i\mathbf{k}\cdot\mathbf{r}^{(2)(1)}_m} e^{i\mathbf{k}\cdot\mathbf{r}^{(2)}_{jm}} T_L(kr_{mm'}, \hat{k}\cdot\hat{r}_{mm'}) \left[\mathbf{I} - \hat{k}\hat{k}\right] e^{-i\mathbf{k}\cdot\mathbf{r}^{(2)}_{im'}} e^{-i\mathbf{k}\cdot\mathbf{r}^{(2)(1)}_{m'}}. \tag{3.21}$$

For the far–zone elements in (3.7), upon discretizing the spectral integral in (3.19) using numerical quadrature for level 1, we obtain

$$Z_{ji} = \frac{ik}{4\pi} \sum_{\hat{k}^{(1)}} w_{\hat{k}^{(1)}} e^{i\mathbf{k}\cdot\mathbf{r}^{(2)(1)}_m} \mathbf{V}_{jm}(\hat{k}^{(2)}) T_L(k^{(1)} r_{mm'}, \hat{k}^{(1)}\cdot\hat{r}_{mm'}) \mathbf{V}^*_{im'}(\hat{k}^{(2)}) e^{-i\mathbf{k}\cdot\mathbf{r}^{(2)(1)}_{m'}} \tag{3.22}$$

where again

$$\mathbf{V}_{im'}(\hat{k}) = \int_s d\mathbf{r}_i e^{i\mathbf{k}\cdot\mathbf{r}^{(2)}_{im'}} \left[\mathbf{I} - \hat{k}\hat{k}\right] \mathbf{j}_i(\mathbf{r}_i) \tag{3.23}$$

and

$$\mathbf{V}_{jm}(\hat{k}) = \int_s d\mathbf{r}_j e^{i\mathbf{k}\cdot\mathbf{r}^{(2)}_{jm}} \left[\mathbf{I} - \hat{k}\hat{k}\right] \mathbf{t}_j(\mathbf{r}_j) \tag{3.24}$$

are the signature functions (often referred to as the Fourier transforms of the basis functions) for the EFIE. We note here that we used the superscript (2) for the signature functions since we will use the finer grouping level in computing and storing these signatures to save memory.

It is observed in (3.22) that the discrete values of the signature functions $\mathbf{V}_{im'}(\hat{k}^{(2)})$ and $\mathbf{V}_{jm}(\hat{k}^{(2)})$ are for the k-space discretization at level 2. The computation of (3.22) requires their k-space values at level 1 since the translation operator is generated for the sampling at level 1. We can use interpolation to generate the required k-space samples. Referring to Fig. 3.11 for a smooth function $f(\theta)$, given the samples $f(\theta_i)$

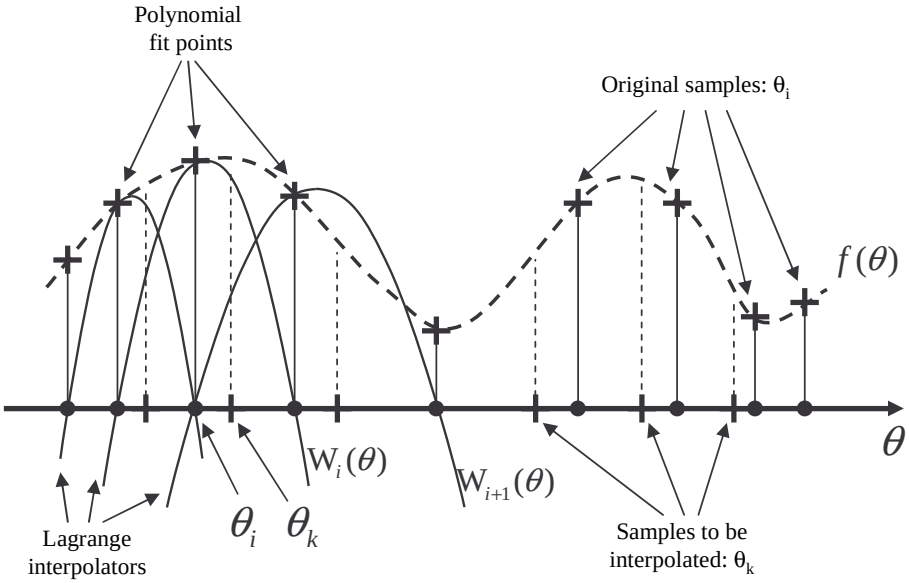


Figure 3.11. Illustration of interpolation matrices in FMM.

at $\theta_i, i = 1, ..., L_{(2)}$, we can generate a different sampling at $\theta_k, k = 1, ..., L_{(1)}$ via interpolation. From the known values of $f(\theta_i)$ we can form an approximation to $f(\theta)$ via

$$\tilde{f}(\theta) = \sum_i W_i(\theta) f(\theta_i) \tag{3.25}$$

where $W_i(\theta)$ are the interpolators which are simple quadratics for the example shown in Fig. 3.11. To generate the new sampling, (3.25) is evaluated at θ_k, i.e.

$$\tilde{f}(\theta_k) = \sum_i W_i(\theta_k) f(\theta_i), k = 1, ..., L_{(2)}. \tag{3.26}$$

The above equation can be cast in a matrix form as $\tilde{f}_k = W_{ki} f_i$, where $f_i = f(\theta_i)$ and $W_{ki} = W_i(\theta_k)$. For a general case, the interpolation matrix $\{W_{ki}\}$ is a full matrix. However, in this case, the overall performance of the MLFMM is dictated by the high

complexity of the interpolation operation. Hence, a sparse interpolation method is necessary not to exacerbate MLFMM performance. One possible method is to use a fixed order polynomial interpolation scheme as shown in Fig. 3.11. For this case, the generated interpolation matrix is sparse since only a few interpolators around the new sample θ_k are included in the computation of $\tilde{f}(\theta_k)$.

For the signature functions, same method can be employed on the unit sphere (i.e. on k space) via

$$\mathbf{V}_{im'}(\hat{k}^{(1)}) = W\left\{\mathbf{V}_{im'}(\hat{k}^{(2)})\right\} = \mathbf{W}_{(1),(2)}\mathbf{V}_{im'}(\hat{k}^{(2)}) \tag{3.27}$$

where $\mathbf{W}_{(1),(2)}$ refers to the interpolation coefficients between levels 1 and 2. We also note the property

$$\mathbf{V}_{im'}(\hat{k}^{(2)}) = \mathbf{W}^{T}_{(1),(2)}\mathbf{V}_{im'}(\hat{k}^{(1)}) \tag{3.28}$$

where the superscript T implies matrix transposition. This interpolation strategy enables the generation of signatures of parent clusters using the collective signatures of its children. We must note here that the interpolated signatures of the children must also be phase-corrected through a simple phase-shifting operation as given in (3.22).

The matrix-vector product for this two-level grouping can be carried out as follows. The aggregations are done for the source clusters at the finer level 2 for each $\hat{k}^{(2)}$ direction. The k-space sampling of the aggregated field signatures of the source clusters are then interpolated to compute the signatures on the required k-space samples at level 1 using (3.27). The phase centers of the aggregated field signatures of the source clusters are then shifted to the center of the parent source clusters at the coarser level 1 by simply multiplying with $e^{-i\mathbf{k}\cdot\mathbf{r}^{(2)(1)}_{m'}}$ as in (3.22). The same procedure is applied to all non-empty children corresponding to source clusters at level 1. After translating the signature onto the field cluster at this coarse level, the same operations are carried out in reverse order to compute the final matrix-vector

product. Namely, the phase center of the translated field is shifted to the center of the field cluster at the finer level on multiplying by $e^{i\mathbf{k}\cdot\mathbf{r}_m^{(2)(1)}}$. This phase shifted field is finally interpolated onto the k-space integration points at the finer level 2 using the transpose of the interpolation matrix. Once we have the signature of the incoming plane waves onto the testing cluster in level 2, we use the signatures of the testing functions in the testing group at this finest level to compute the actual reaction. Fig. 3.12 depicts the steps of this 2 level FMM.

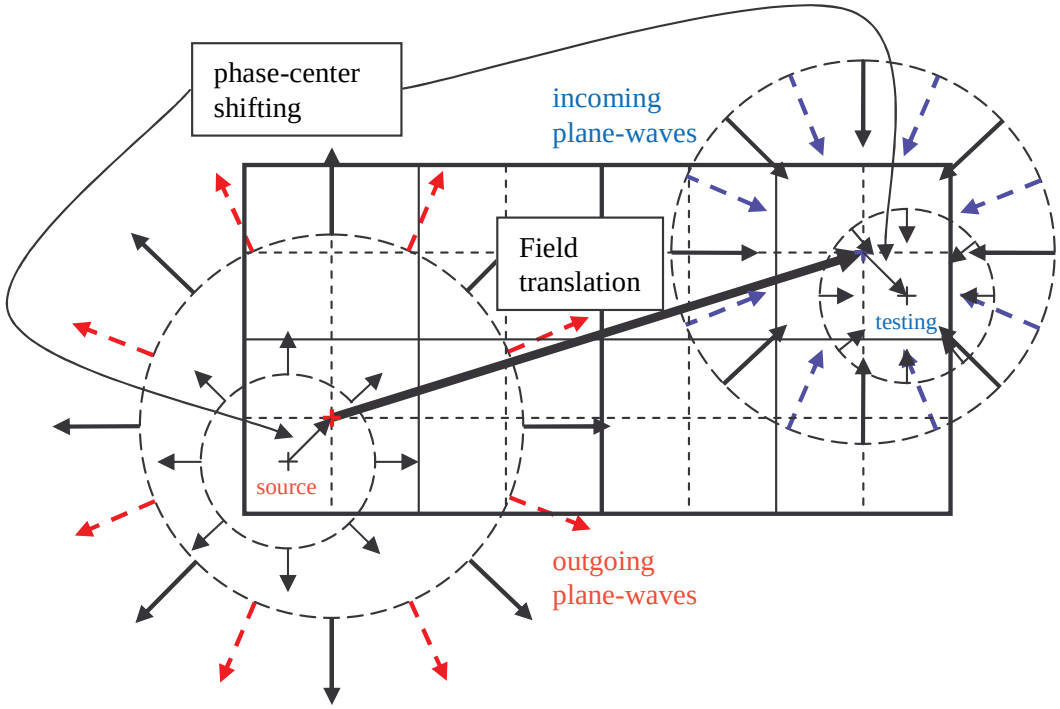


Figure 3.12. Illustration of two level interactions between source and testing groups in FMM.

As mentioned above, the complexity of the MLFMM relies on the use of a sparse interpolation technique. Since the signature functions are smooth on the unit sphere, a 4×4^{th} order two-dimensional piecewise-polynomial interpolation, generating a sparse interpolation matrix, has been adopted in this work. For increasing problem sizes, more accurate interpolation schemes may be necessary. Although there exists more accurate interpolation schemes in the literature [51, 52, 53], these are more costly than simple polynomial interpolation, deteriorating the execution speed

of the matrix-vector product.

It is necessary that the multilevel grouping algorithm provide the neighborhood information of all clusters at all levels. When constructing the far-zone clusters for a cluster i at level $(l+1)$, the condition is that the parents (at level (l)) of clusters i and j at level $(l+1)$ be in the near-zone of each other (see Fig. 3.13). Otherwise, the interactions of clusters i and j are computed through translations at the coarser level (l).

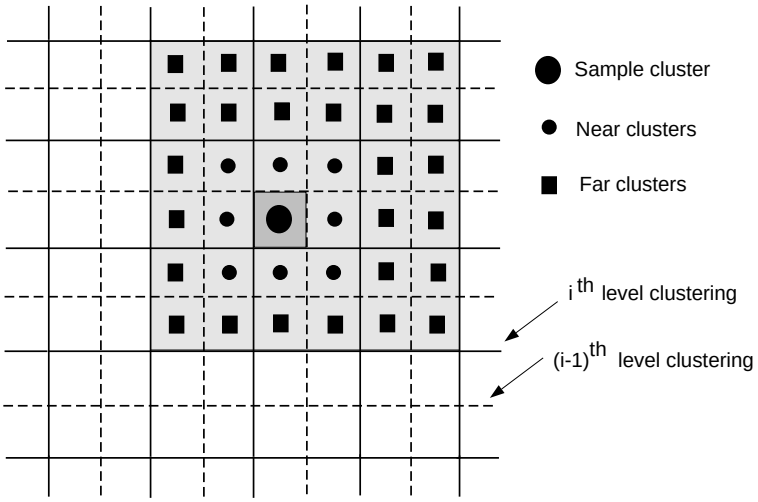


Figure 3.13. 2-dimensional multilevel clustering example.

In the most general multilevel form, there are five steps in the MLFMM:

Step 1. All the collective signatures of all source functions in the finest level are computed using the individual signatures of the basis functions.

Step 2. For all coarser levels, the collective signatures are calculated through interpolation and shifting using the signatures of the children.

Step 3. For each level (l) these signatures are translated onto far clusters such that their parents are in the near-zone of each other at level $(l+1)$.

Step 4. The translated incoming signatures are shifted and interpolated onto the children at the finer levels.

Step 5. At the finest level, the interactions are computed using the signatures of the

testing functions and the incoming plane wave decompositions.

This multilevel tree-structure provides $O(N \log N)$ CPU and memory complexity. Furthermore, due to the translational symmetry in the grouping, significant memory saving is achieved in the storage of the translation operator. This low $O(N \log N)$ complexity enables the solution of significantly larger problems on low-cost personal computer platforms. This is demonstrated in the next section. Nevertheless, as the target becomes larger, one must resort to supercomputers. The challenges in implementing the MLFMM on distributed memory computers are addressed in Appendix D.

3.4 Examples and Validations

Below, we present validations and complexity evaluations of the implemented MLFMM solver for electromagnetic scattering and radiation problems involving electrically large arbitrarily shaped targets in free space. For the presented data, the MLFMM strategy has been implemented using curved biquadratic surface elements and conformal rooftop basis functions as described above. However, the previous development of the MLFMM applies to any element shape and basis function.

Our first example is a radiation problem. A Hertzian dipole is placed above a PEC sphere. This problem has a closed form solution and hence constitutes a benchmark. The problem was solved at three different frequencies using the same surface mesh of the sphere. This also serves for the evaluation of using curvilinear elements aiming at reducing the sampling requirements. Fig. 3.14 depicts the induced surface currents for the simulation frequencies. The creeping wave effects in the shadow region of the sphere are clearly resolved in these current amplitude plots. A more important observable in case of radiation scenarios is the radiation pattern. Fig. 3.15 depicts the normalized patterns at the corresponding frequencies along with the analytical Mie series solution [27].

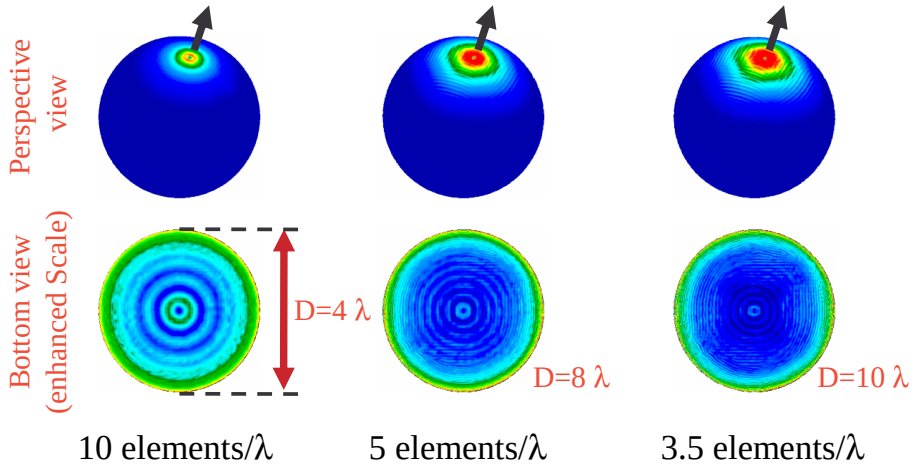


Figure 3.14. Induced surface currents on the sphere due to the Hertzian dipole.

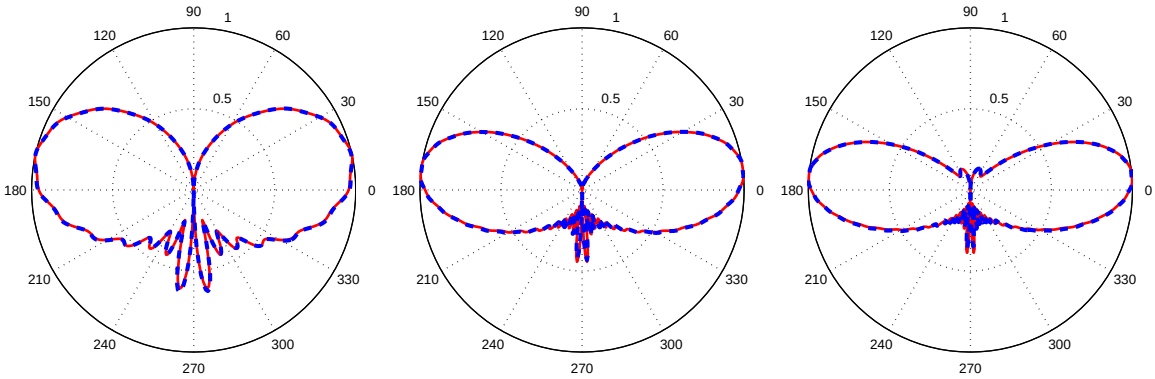


Figure 3.15. Radiation patterns (linear-scale) of the Hertzian dipole over the sphere, solid: Mie solution, dashed: MLFMM.

Computer timing results and FMM data for the sphere problem are summarized in Table 3.1. The important thing to note here is that, although the same mesh has been used to solve the same problem at increasing frequencies, the execution time for the matrix-vector product increases as the number of multipoles increase. In this sense, the FMM sampling requirements are decoupled from the spatial sampling requirements.

However, if we were to keep a $\lambda/10$ fixed sampling rate for the three frequencies, the problem size for the higher frequency would have been considerably larger. Table 3.2 summarizes the requirements if a constant sampling rate of $\lambda/10$ was to be used to solve the same problem at the corresponding frequencies. At the high-

mesh size	near-field matrix	number of multipoles	number of iterations	time per iteration	solution time
10 per λ	5.0×10^6	5,7,10	28	3.9s.	108.8s.
5 per λ	2.5×10^6	7,11,18	19	6.9s.	130.4s.
3.5 per λ	2.0×10^6	8,14,22	21	10.1s.	212.3s.

Table 3.1. Computational requirements for a fixed number of unknowns ($N = 11,164$) at increasing frequencies.

est frequency, a sampling rate of $\lambda/10$ would have resulted in $90,000$ unknowns and would have required 38.4 seconds of CPU time per matrix-vector product, whereas at this frequency, the lower sampling rate required 10.1 seconds, both being the timing results for the 3-level FMM. Hence, using curvilinear elements along with MLFMM resulted in a 4 times speed-up in the matrix-vector product for this specific example, proving the motivation behind the presented solution method.

mesh size	problem size	time per iteration $O(N \log N)$
10 per λ	10,000	3.9s.
10 per λ	40,000	15.9s.
10 per λ	90,000	38.4s.

Table 3.2. Computational requirements when a fixed $\lambda/10$ sampling is used for increasing simulation frequencies.

As a second geometry, we consider the PEC plate. Three levels of FMM were used to solve the scattering by this plate. Fig. 3.16 depicts the computed bistatic RCS along with the approximate physical optics solution. The plate is 30 wavelengths long at the simulation frequency, and even for broadside incidence, the inaccuracy of the PO approximation for grazing observation angles is resolved by the full-wave solution. Fig. 3.17 depicts the computational complexities of increasing levels of FMM for the plate problem. The curves start with an $O(N^{3/2})$ complexity for the single level FMM and tend toward the $O(N \log N)$ limit for the general multilevel case.

The third considered example is a true multilevel solution for a large sphere at two different frequencies. The parameter α_L for the multipole sum is set to 2 for

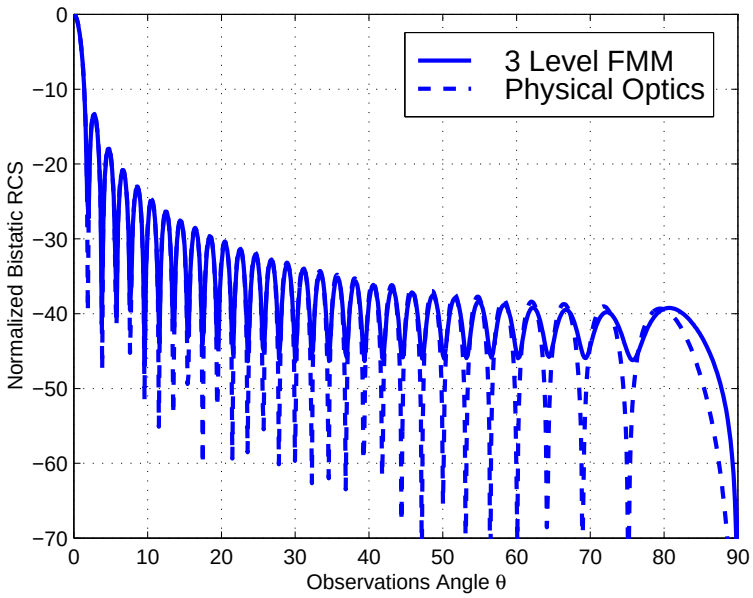


Figure 3.16. Bistatic RCS of a flat plate using 3-level FMM.

this example. For the first case, the diameter of the sphere is 8 wavelengths. When a system of $7,500$ unknowns is used to solve this problem, the solver generated 5 MLFMM levels with the number of multipoles for each level being $\{9, 11, 14, 20, 32\}$. The total memory used was 447 MBytes and it took 74 seconds of CPU time to fill the near-field matrix and 17 CGS iterations (972 seconds) to converge to an error less than 10^{-3}. The bistatic RCS is plotted in Fig. 3.18 and all values are within 2.9% of the analytical Mie series data. We note here that the employed spatial sampling was below the nominal $\lambda/10$ for this computation. When nominal sampling is used, the resulting system has $28,812$ unknowns (an increase by nearly a factor of 4). The MLFMM solution of this system for the same frequency is given in Fig. 3.19. This computation also generated 5 MLFMM levels with $L = \{7, 8, 12, 18, 30\}$. The required total memory was 508 MBytes and the near-field matrix was filled in 1034 seconds (on an Intel-PIII with 1 GHz clock speed). Specifically, the CGS solver took 20 iterations to converge and the total solution was completed in 1110 seconds. The results are within 0.4% of the analytical solution. This example demonstrates that at a given frequency, the MLFMM solution takes about the same time regardless of the

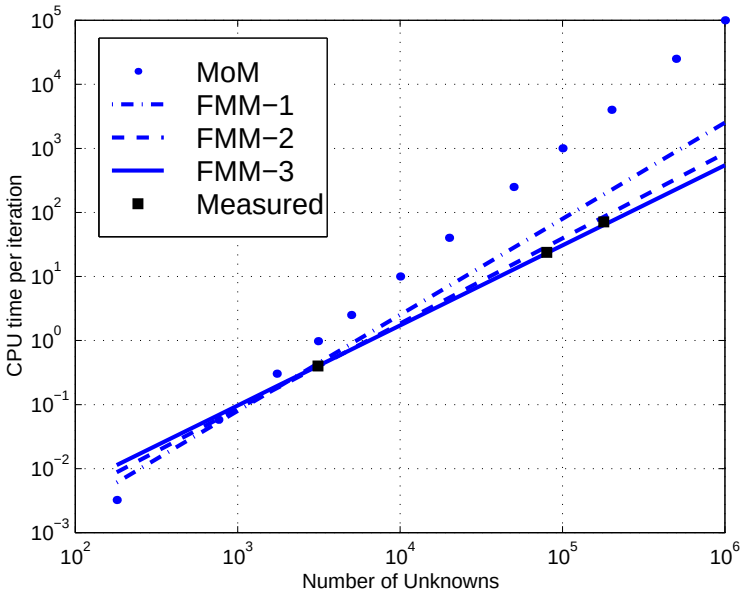


Figure 3.17. Complexities of various levels of the FMM.

spatial sampling, and this is due to the frequency domain nature of the formulation. As will be discussed later, this property proves to be very useful in solving VIEs involving high contrast material parameters where the sampling rate is determined by the wavelength inside the inhomogeneity and is therefore much higher as compared to that on PEC surfaces. However, since the k-space sampling is based on the Green's function, the number of samples is independent of the specific material parameters present in the problem.

When the $28,812$ unknown mesh is used to solve the problem at twice the frequency, i.e. for a 16λ diameter sphere, the solver generated 6 MLFMM levels with $L = \{9, 11, 14, 20, 32, 55\}$. The solution required 1.1 GBytes of memory and the near-field matrix was filled in 933 seconds. The CGS solver converged in 18 iterations taking a total time of 4185 seconds. The actual RCS is shown in Fig. 3.20 and is within 2.6% of the reference Mie series solution. When we used a nominally discretized mesh, $110,190$ unknowns are needed and the corresponding solution is given in Fig. 3.21. The computed bistatic RCS is now within 0.6% of the reference solution for this case. Were we to use traditional MoM solution procedures (with Gaussian

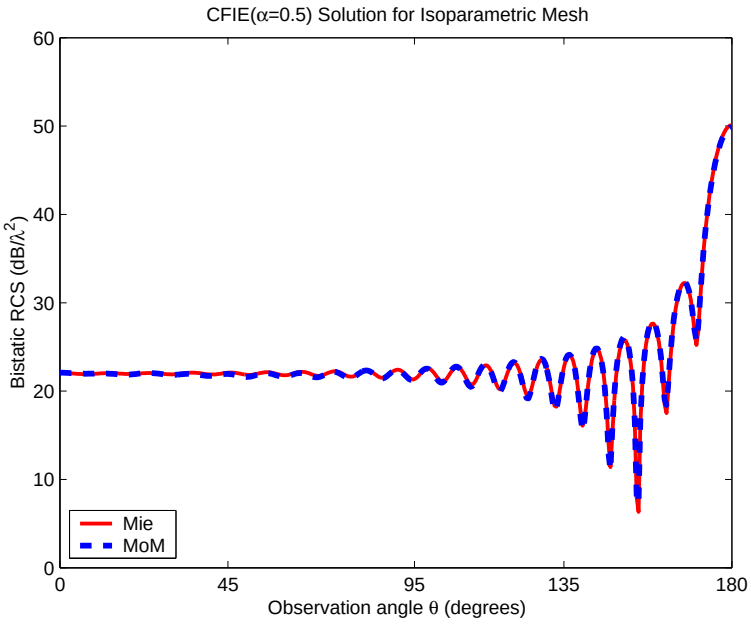


Figure 3.18. Bistatic RCS of an 8λ diameter PEC sphere using $7,500$ unknowns.

elimination or LU decomposition) 97.1 GBytes would have been required and this is well beyond the storage capacity even for supercomputing facilities.

Nevertheless, even with the MLFMM, when the storage exceeds 4 Gbytes, we must then resort to using multiprocessor supercomputing resources. Porting of the MLFMM algorithm on distributed memory supercomputers requires a carefully balanced distribution of the problem among the individual processors and minimization of inter-processor communication. A brief discussion on the challenges of distributed memory implementation of the MLFMM using the Message Passing Interface (MPI) programming paradigm is given in Appendix D.

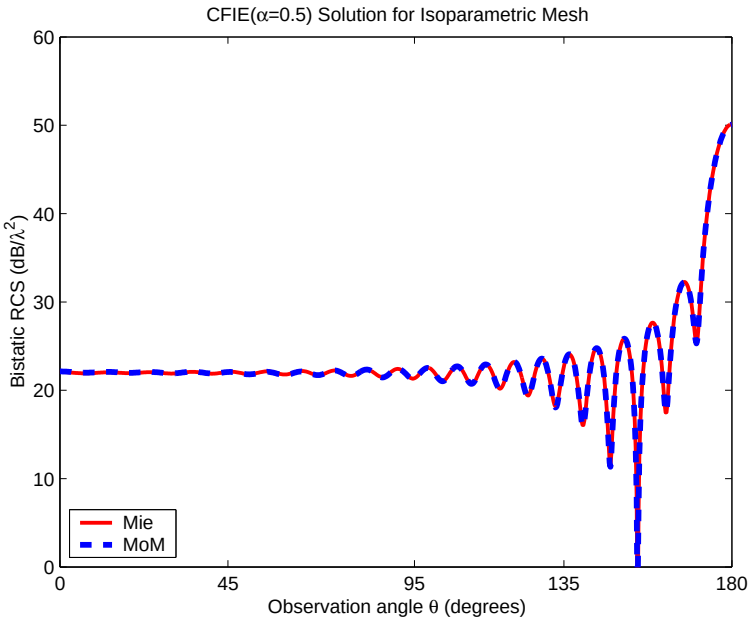


Figure 3.19. Bistatic RCS of an 8λ diameter PEC sphere using $28,812$ unknowns.

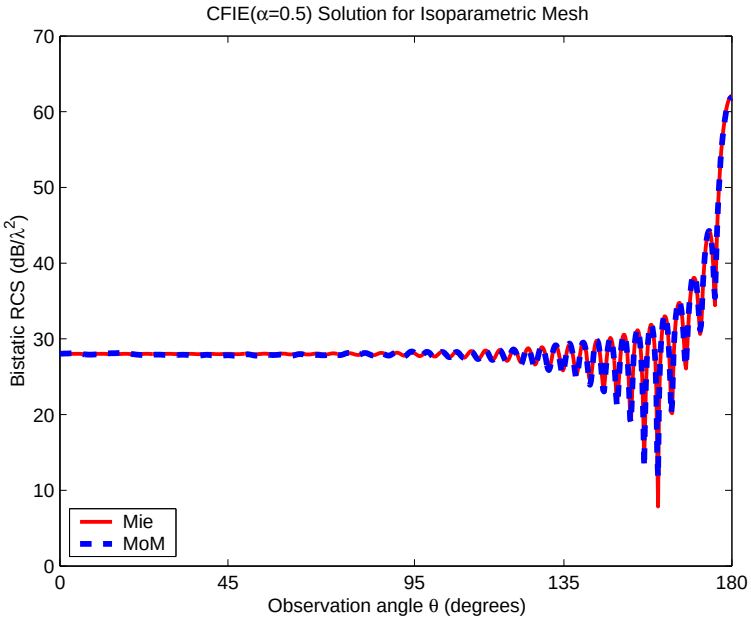


Figure 3.20. Bistatic RCS of an 16λ diameter PEC sphere using $28,812$ unknowns.

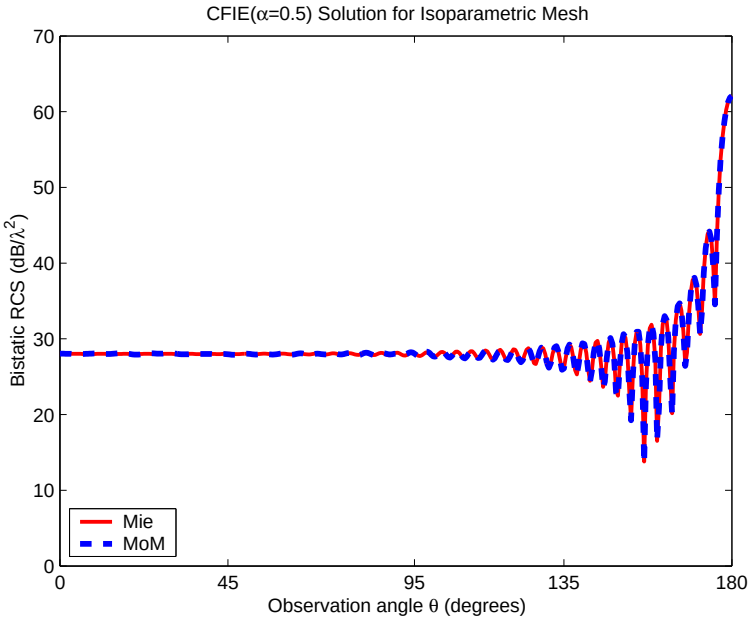


Figure 3.21. Bistatic RCS of an 16λ diameter PEC sphere using $110,190$ unknowns.

CHAPTER 4

Hybrid Finite Element-Boundary Integral Method for Volumetric Problems

Electromagnetic scattering by inhomogeneous structures is of great interest in evaluating the overall scattering by modern composite vehicles. The same mathematical formulations are also suited for antennas, high frequency microwave circuits, electromagnetic coupling and interference, and inverse scattering applications. Thus, much interest exists in developing efficient formulations and numerical solutions in modeling electrically large volumetric problems having arbitrary permittivity (ϵ) and permeability (μ).

During the 1990s, treatment of inhomogeneous scatterers focused on finite element (FE) methods [18] and their hybrid finite element-boundary integral (FE-BI) counterparts [18, 19, 54, 55]. The FE methods with approximate mesh termination schemes, such as absorbing boundary conditions, artificial absorbers, or perfectly matched layers, were found attractive because of their geometrical and material adaptability coupled with their low memory requirements. The introduction of fast integral methods [44, 56] prompted renewed interest in the solution of FE-BI formulations for large-scale simulations.

In this chapter, we consider the hybrid FE-BI formulations for scattering by penetrable structures. Again using curvilinear elements as in [19], we present the MLFMM implementation for electrically large inhomogeneous problems. For completeness, here we describe the geometry modeling by curvilinear hexahedral elements and conformal electric field basis functions as well as the mathematical tools used to develop

the formulations in parametric coordinates. This information will also be used in the next chapter when we consider the alternative VIE methods.

4.1 FE-BI Formulation

FE [18, 55, 57] and FE-BI methods have been among the workhorse techniques for frequency domain simulations over the past ten years. Here, we present the FE-BI from a general viewpoint as applied to non-planar structures [19]. However, our main contribution is the introduction of the MLFMM within the FE-BI context.

In formulating the FE-BI, we begin with the functional for the electric field in the solution domain v (see Fig. 4.1) given by

$$F(\mathbf{E}) = \frac{1}{2}\int_v d\mathbf{r}\left[(\nabla\times\mathbf{E})\cdot\overline{\mu}_r^{-1}\cdot(\nabla\times\mathbf{E}) - k_0^2\mathbf{E}\cdot\overline{\epsilon}_r\cdot\mathbf{E}\right] - ik_0\eta_0\int_s d\mathbf{r}\hat{n}\cdot(\mathbf{E}\times\mathbf{H}) \quad (4.1)$$

where $\overline{\mu}_r$ and $\overline{\epsilon}_r$ are the tensor constitutive parameters of the inhomogeneous medium, k_0 and η_0 are the wavenumber and characteristic impedance in free-space, and the boundary s encloses the volume v with $\hat{n}$ pointing outward from v as shown in Fig. 4.1. For a unique solution, it is necessary to relate $\mathbf{E}$ and $\mathbf{H}$ on the bounding surface s to

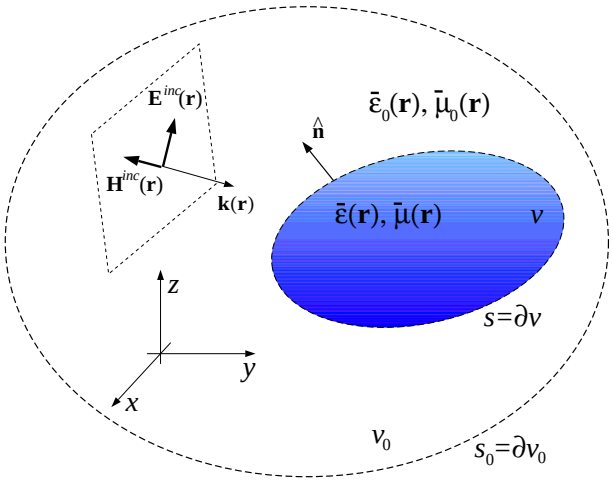


Figure 4.1. FE-BI solution domain.

obtain an equation in terms of $\mathbf{E}$ only. This is done by introducing the Stratton-Chu

integral equations [22]. The electric field integral equation (EFIE) is given by

$$\eta_0 \Theta\{\hat{n} \times \mathbf{H}\} - \Omega\{\mathbf{E} \times \hat{n}\} = \mathbf{E}^{inc}, \tag{4.2}$$

whereas the magnetic field integral equation (MFIE) has the form

$$\eta_0 \Omega\{\hat{n} \times \mathbf{H}\} + \Theta\{\mathbf{E} \times \hat{n}\} = \eta_0 \mathbf{H}^{inc}, \tag{4.3}$$

in which the integral operators $\Theta\{\cdot\}$ and $\Omega\{\cdot\}$ are defined as

$$\begin{aligned} \Theta(\mathbf{X}) &= -ik_0 \int_s d\mathbf{r}' \overline{\mathbf{G}}(\mathbf{r}, \mathbf{r}') \cdot \mathbf{X}(\mathbf{r}') \\ &= -ik_0 \left[\int_s d\mathbf{r}' g(\mathbf{r}, \mathbf{r}') \mathbf{X}(\mathbf{r}') + \frac{1}{k_0^2} \int_s d\mathbf{r}' \nabla g(\mathbf{r}, \mathbf{r}') \nabla' \cdot \mathbf{X}(\mathbf{r}') \right] \\ \Omega(\mathbf{X}) &= T\mathbf{Y}(\mathbf{r}) + \int\!\!\!\!\!\!- _s d\mathbf{r}' \mathbf{X}(\mathbf{r}') \times \nabla g(\mathbf{r}, \mathbf{r}') \end{aligned} \tag{4.4}$$

with $T = 1 - \beta/4\pi$ ($\beta = 2\pi$ for a smooth surface), $\hat{n}$ denoting the unit normal to the bounding surface, and the relation $\mathbf{X} = \hat{n} \times \mathbf{Y}$. Typically, a combination of the two, referred to as the CFIE, is employed to avoid internal resonance difficulties and poor conditioning.

The electric field is solved from (4.1) by setting $\partial F(\mathbf{E})/\partial \mathbf{E} = 0$. Discretizing the resulting equations using the appropriate expansion [19] (assuming no internal excitations inside v) results in the system

$$\begin{bmatrix} \mathbf{E^{vv}} & \mathbf{E^{vs}} & \mathbf{0} \\ \mathbf{E^{sv}} & \mathbf{E^{ss}} & \mathbf{B} \\ \mathbf{0} & \mathbf{P} & \mathbf{Q} \end{bmatrix} \begin{bmatrix} \mathbf{E_v} \\ \mathbf{E_s} \\ \mathbf{H_s} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{b} \end{bmatrix} \tag{4.5}$$

where the bold symbols $\mathbf{E}^{vv}$, $\mathbf{E}^{vs}$, $\mathbf{E}^{sv}$, $\mathbf{E}^{ss}$, $\mathbf{B}$, $\mathbf{P}$, and $\mathbf{Q}$ represent sub-matrices.

Specifically, the elements of the sub-matrices in (4.5) are given as

$$\begin{aligned}
E_{ji} &= \int_v d\mathbf{r} \nabla \times \mathbf{e}_j \cdot \overline{\mu}_r^{-1} \cdot \nabla \times \mathbf{e}_i + \int_v d\mathbf{r} \mathbf{e}_j \cdot \overline{\epsilon}_r \cdot \mathbf{e}_i \\
B_{ji} &= ik_0\eta_0 \int_s d\mathbf{r} \mathbf{e}_j \cdot (\hat{n} \times \mathbf{h}_{si}) \\
P_{ji} &= \int_s d\mathbf{r} (\hat{n} \times \mathbf{h}_{sj}) \cdot [-\alpha\Omega\{\mathbf{e}_i \times \hat{n}\} + (1-\alpha)\Theta\{\mathbf{e}_i \times \hat{n}\}] \\
Q_{ji} &= \eta_0 \int_s d\mathbf{r} (\hat{n} \times \mathbf{h}_{sj}) \cdot [\alpha\Theta\{\hat{n} \times \mathbf{h}_{si}\} + (1-\alpha)\Omega\{\hat{n} \times \mathbf{h}_{si}\}] \\
b_j &= \alpha \int_s d\mathbf{r} (\hat{n} \times \mathbf{h}_{sj}) \cdot \mathbf{E}^{inc} + (1-\alpha)\eta_0 \int_s d\mathbf{r} (\hat{n} \times \mathbf{h}_{sj}) \cdot \mathbf{H}^{inc}
\end{aligned} \tag{4.6}$$

where $\mathbf{e}_i$ and $\mathbf{e}_j$ denote the basis and testing functions for the volume electric field intensities, respectively. Also, the surface magnetic field intensity is expanded using the basis functions $\mathbf{h}_s$ and α is the CFIE scale factor chosen from zero to unity. However, we can use the definitions $\mathbf{J} = \hat{n} \times \mathbf{H}$ and $\mathbf{M} = \mathbf{E} \times \hat{n}$ and define expansions of these surface quantities in the integral operators. With $\mathbf{J} = \sum_i h_{si}\mathbf{j}_i$ and $\mathbf{M} = \sum_i e_{si}\mathbf{m}_i$, the matrix entries for the integral operators in (4.6) can be rewritten in a more compact form (for CFIE) as

$$\begin{aligned}
P_{ji} &= \int_s d\mathbf{r}\mathbf{j}_j \cdot [-\alpha\Omega\{\mathbf{m}_i\} + (1-\alpha)\Theta\{\mathbf{m}_i\}] \\
Q_{ji} &= \eta_0 \int_s d\mathbf{r}\mathbf{j}_j \cdot [\alpha\Theta\{\mathbf{j}_i\} + (1-\alpha)\Omega\{\mathbf{j}_i\}] \\
b_j &= \alpha \int_s d\mathbf{r}\mathbf{j}_j \cdot \mathbf{E}^{inc} + (1-\alpha)\eta_0 \int_s d\mathbf{r}\mathbf{j}_j \cdot \mathbf{H}^{inc}.
\end{aligned} \tag{4.7}$$

We note here that the basis functions used in expanding the surface unknowns are identical to those presented in Chapter 2. Specifically, for $\mathbf{m}_i$ this is a direct consequence of the definition $\mathbf{m}_i = \mathbf{e}_i \times \hat{n}$. The electric current unknowns which are independent in the formulation are defined using the same conformal basis functions.

For a general inhomogeneous structure consisting of material and conducting components, the resulting system in (4.5) is highly heterogeneous, consisting of a sparse FE part ($[E]$ and $[B]$ sub-matrices) and a full integral equation part ($[P]$ and $[Q]$ sub-matrices). This highly heterogeneous FE-BI system may result in a poorly convergent iterative solution, especially for large scale systems. To obtain an improved matrix

condition for (4.5), we can test the MFIE of (4.3) with $\hat{n} \times \mathbf{j}_j$ [45, 58], rather than simply testing it with $\mathbf{j}_j$ as done in (4.7). The aim is to generate a strong diagonal contribution in $[Q]$ and a weak diagonal contribution in $[P]$. Doing so, the resulting matrix entries take the form

$$\begin{aligned} P_{ji} &= -\alpha \int_s d\mathbf{r}\mathbf{j}_j \cdot \Omega\{\mathbf{m}_i\} + (1-\alpha) \int_s d\mathbf{r}(\hat{n} \times \mathbf{j}_j) \cdot \Theta\{\mathbf{m}_i\} \\ Q_{ji} &= \alpha\eta_0 \int_s d\mathbf{r}\mathbf{j}_j \cdot \Theta\{\mathbf{j}_i\} + (1-\alpha) \int_s d\mathbf{r}(\hat{n} \times \mathbf{j}_j) \cdot \Omega\{\mathbf{j}_i\} \\ b_j &= \alpha \int_s d\mathbf{r}\mathbf{j}_j \cdot \mathbf{E}^{inc} + (1-\alpha)\eta_0 \int_s d\mathbf{r}(\hat{n} \times \mathbf{j}_j) \cdot \mathbf{H}^{inc}, \end{aligned} \tag{4.8}$$

and this should be compared to (4.7).

Since the application of fast methods (such as the MLFMM) implies use of iterative solvers, a poor convergence behavior implies serious difficulties in terms of total solution time, especially for multi-spectral simulations. Preconditioning methods may therefore be necessary for certain problems to achieve convergence. Clearly, it is important to use preconditioners which can be implemented in favorable CPU times. Among them, the block-diagonal [39] and the ILU [40] preconditioners have been found quite effective. In modeling layered geometries, substantial improvement in matrix condition can be achieved by choosing hexahedral elements rather than tetrahedra. Curvilinear elements also allow for geometrical modeling fidelity and have been shown effective for antenna arrays [59] as well as scattering applications [19, 31]. When dealing with large antenna arrays, certain advantages associated with the repeatability of each array element can be exploited. Details of such a decomposition method can be found in [59].

We outlined the MLFMM in Chapter 3 for perfectly conducting targets. However, since the integral operators in (4.8) are identical to those used in Chapter 2, the outlined MLFMM implementation can be extended to the FE-BI method and to VIE formulations by merely incorporating the signature functions into their respective algorithms. Specifically, for the FE-BI method, since the same basis functions are

used for both surface unknowns ($\mathbf{E}_s$ and $\mathbf{H}_s$), the same signature functions can be utilized for both. Also, clustering of the surface unknowns is based solely on the surface magnetic field and thus the same clustering can be used for the surface electric field unknowns, leading to significant memory savings.

4.2 Geometry Modeling and Basis Functions

Before proceeding with the FEM tools (finite elements and basis functions), we first define the mathematical tools we use in the definitions and derivations that follow. We begin by defining the solution space in terms of three parameters (u, v, w) as shown in Fig. 4.2 for the most general case. The mathematical representation of this

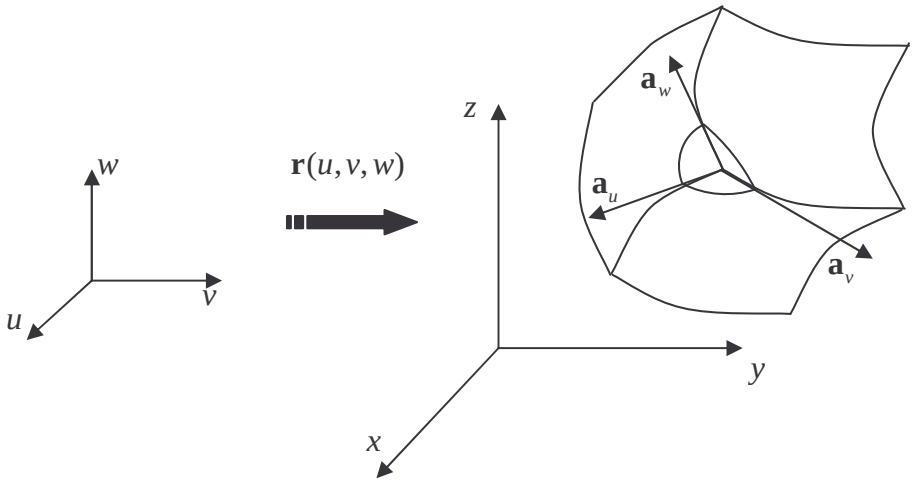


Figure 4.2. Real space as a mapping of the parametric space.

parametric space takes the general form $\mathbf{r} = \mathbf{r}(u, v, w)$. In order to study various properties of this curvilinear space, we need to define a reference system. This can be done using either the covariant unitary vectors

$$\mathbf{a}_u = \frac{\partial \mathbf{r}}{\partial u}, \qquad \mathbf{a}_v = \frac{\partial \mathbf{r}}{\partial v}, \qquad \mathbf{a}_w = \frac{\partial \mathbf{r}}{\partial w} \tag{4.9}$$

or equivalently the contravariant unitary vectors

$$\mathbf{a}^u = \frac{1}{V_+}\mathbf{a}_v \times \mathbf{a}_w, \qquad \mathbf{a}^v = \frac{1}{V_+}\mathbf{a}_w \times \mathbf{a}_u, \qquad \mathbf{a}^w = \frac{1}{V_+}\mathbf{a}_u \times \mathbf{a}_v \tag{4.10}$$

where $V_+ = \mathbf{a}_u \cdot (\mathbf{a}_v \times \mathbf{a}_w)$ is used for normalization. We must note the property that $\mathbf{a}^i \cdot \mathbf{a}_j = \delta_{ij}$ $(i, j = u, v, w)$ where δ_{ij} is the Kronecker delta function.

The Jacobian of the parametric transformation $\mathbf{r}(u, v, w)$ is given by

$$[J] = \begin{bmatrix} \frac{\partial x}{\partial u} & \frac{\partial y}{\partial u} & \frac{\partial z}{\partial u} \\ \frac{\partial x}{\partial v} & \frac{\partial y}{\partial v} & \frac{\partial z}{\partial v} \\ \frac{\partial x}{\partial w} & \frac{\partial y}{\partial w} & \frac{\partial z}{\partial w} \end{bmatrix} \tag{4.11}$$

and the metric tensor is

$$[G] = \begin{bmatrix} g_{uu} & g_{uv} & g_{uw} \\ g_{vu} & g_{vv} & g_{vw} \\ g_{wu} & g_{wv} & g_{ww} \end{bmatrix} \tag{4.12}$$

with $g_{ij} = \mathbf{a}_i \cdot \mathbf{a}_j$. This metric completely characterizes the geometrical properties of the solution space.

The differential volume and the differential area for a constant w-surface are given by $dv = \sqrt{|G|}dudvdw$ and $ds = (g_{uu}g_{vv} - g_{uv}g_{vu})dudv$, respectively. Here, $|G|$ is the determinant of the metric tensor. An arbitrary vector $\mathbf{F}$ can be represented in either the covariant projection form ($\mathbf{a}_u$, $\mathbf{a}_v$, and $\mathbf{a}_w$ are the covariant vectors)

$$\mathbf{F} = (\mathbf{F} \cdot \mathbf{a}_u)\mathbf{a}^u + (\mathbf{F} \cdot \mathbf{a}_v)\mathbf{a}^v + (\mathbf{F} \cdot \mathbf{a}_w)\mathbf{a}^w, \tag{4.13}$$

or the contravariant projection form ($\mathbf{a}^u$, $\mathbf{a}^v$, and $\mathbf{a}^w$ are the contravariant vectors)

$$\mathbf{F} = (\mathbf{F} \cdot \mathbf{a}^u)\mathbf{a}_u + (\mathbf{F} \cdot \mathbf{a}^v)\mathbf{a}_v + (\mathbf{F} \cdot \mathbf{a}^w)\mathbf{a}_w. \tag{4.14}$$

The basic vector operations in the curvilinear coordinate system can be summarized as

Gradient:

$$\nabla\phi = \frac{\partial\phi}{\partial u}\mathbf{a}^u + \frac{\partial\phi}{\partial v}\mathbf{a}^v + \frac{\partial\phi}{\partial w}\mathbf{a}^w \tag{4.15}$$

for a scalar function $\phi(u, v)$.

Divergence:

$$\nabla \cdot \mathbf{F} = \frac{1}{\sqrt{|G|}} \left\{ \frac{\partial}{\partial u} \left[(\mathbf{F} \cdot \mathbf{a}^u) \sqrt{|G|} \right] + \frac{\partial}{\partial v} \left[(\mathbf{F} \cdot \mathbf{a}^v) \sqrt{|G|} \right] + \frac{\partial}{\partial w} \left[(\mathbf{F} \cdot \mathbf{a}^w) \sqrt{|G|} \right] \right\}. \tag{4.16}$$

Curl:

$$\begin{aligned} \nabla \times \mathbf{F} &= \frac{1}{\sqrt{|G|}} \left\{ \left[\frac{\partial (\mathbf{F} \cdot \mathbf{a}_w)}{\partial v} - \frac{\partial (\mathbf{F} \cdot \mathbf{a}_v)}{\partial w} \right] \mathbf{a}_u \right. \\ &+ \left. \left[\frac{\partial (\mathbf{F} \cdot \mathbf{a}_u)}{\partial w} - \frac{\partial (\mathbf{F} \cdot \mathbf{a}_w)}{\partial u} \right] \mathbf{a}_v + \left[\frac{\partial (\mathbf{F} \cdot \mathbf{a}_v)}{\partial u} - \frac{\partial (\mathbf{F} \cdot \mathbf{a}_u)}{\partial v} \right] \mathbf{a}_w \right\}. \end{aligned} \tag{4.17}$$

With the above representations, we may next proceed to define the tri-quadratic hexahedral finite elements used in [19]. Given a set of 27 points in space $\{\mathbf{r}_{ijk}, i, j, k = 0, 1, 2\}$ on a topologically cubic grid, a curvilinear tri-quadratic volume element as shown in Fig. 4.3 can be constructed as a transformation of a unit cube in the (u, v, w) parametric space as

$$\mathbf{r}(u, v, w) = \sum_{i=0}^{2} \sum_{j=0}^{2} \sum_{k=0}^{2} L_{ijk}(u, v, w) \mathbf{r}_{ijk} \tag{4.18}$$

where $L_{ijk}(u, v, w)$ are the Cartesian products of the usual Lagrange interpolation functions (see Appendix A) and $0 \leq u \leq 1$, $0 \leq v \leq 1$, $0 \leq w \leq 1$. By simply differentiating the Lagrange interpolators, it is now straightforward to compute derivatives with respect to parameters u, v, or w for calculating various operations such as the divergence and the curl. Once the necessary reference system is generated using these derivatives, we can proceed to define the electric field basis functions in the parametric space.

The basis functions for the electric field expansion inside the parametric volume elements are constructed in terms of contravariant unitary vectors for the FE-BI formulations due to the inherent connection between the volume expansion and the necessary expansion of the surface electric and magnetic currents as well as the condition on the continuity of the tangential electric field component. Fig. 4.4 shows the

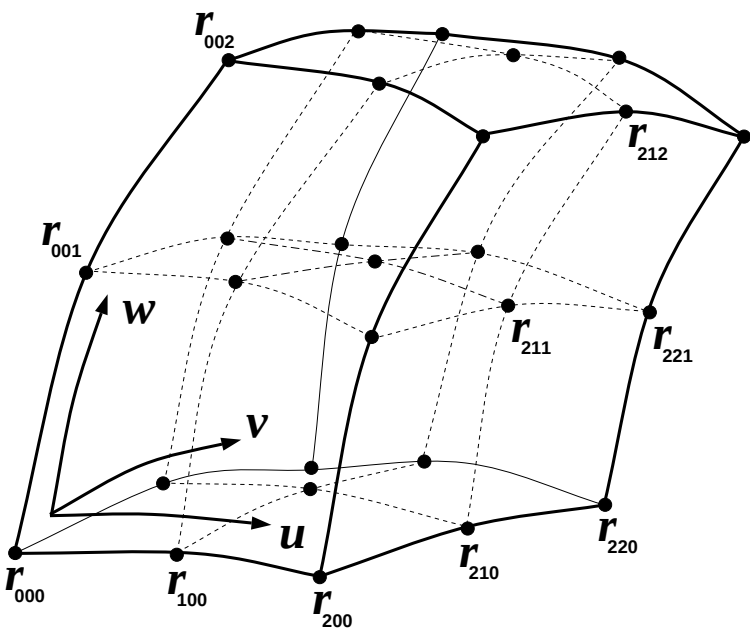


Figure 4.3. Quadratic hexahedral finite element.

convention used to associate basis functions with the edges of the elements. For each volume element, the 12 basis functions associated with the 12 edges of the element are given by the covariant projection form

$$\begin{aligned}
\mathbf{e}_1 &= (v)(w)\mathbf{a}^u, & \mathbf{e}_2 &= (1-v)(w)\mathbf{a}^u, \\
\mathbf{e}_3 &= (v)(1-w)\mathbf{a}^u, & \mathbf{e}_4 &= (1-v)(1-w)\mathbf{a}^u, \\
\mathbf{e}_5 &= (u)(w)\mathbf{a}^v, & \mathbf{e}_6 &= (1-u)(w)\mathbf{a}^v, \\
\mathbf{e}_7 &= (u)(1-w)\mathbf{a}^v, & \mathbf{e}_8 &= (1-u)(1-w)\mathbf{a}^v, \\
\mathbf{e}_9 &= (u)(v)\mathbf{a}^w, & \mathbf{e}_{10} &= (1-u)(v)\mathbf{a}^w, \\
\mathbf{e}_{11} &= (u)(1-v)\mathbf{a}^w, & \mathbf{e}_{12} &= (1-u)(1-v)\mathbf{a}^w.
\end{aligned} \tag{4.19}$$

These definitions ensure that the basis functions have the same tangential components across the element faces, since each basis function is constructed over a set of hexahedra sharing a common edge. Properly pairing the basis functions on neighboring elements, as is done for the surface elements in Chapter 2, we construct the volumetric basis function for the electric field expansion. A detailed discussion on

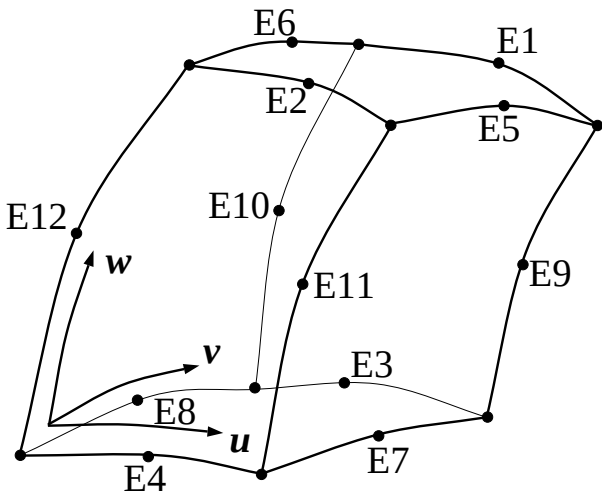


Figure 4.4. Local numbering of the edges of the hexahedral element

the FE-BI method using this approach can be found in [19]. We note, though, that a more general higher-order set of basis functions can be constructed if the linear variations in (4.19) are replaced by appropriate functions [18, 38].

With the above expansion functions for the electric field inside the FE domain, it can be shown that the basis functions for the surface electric field are identical to those described in Chapter 2. This can be done upon noting that the tangential surface electric field is related to the magnetic current via $\mathbf{M} = \mathbf{E} \times \hat{n}$. Then, $\mathbf{m}_i = \mathbf{e}_i \times \hat{n}$, with $\mathbf{e}_i$ as in (4.19) gives the conformal curvilinear rooftop basis functions [19] defined in (2.7). The same basis functions are used to approximate the surface magnetic fields, where we must now use $\mathbf{J} = \hat{n} \times \mathbf{H}$ with $\hat{n}$ being the outward unit normal on the FE boundary s.

To tackle large practical problems involving inhomogeneities, we again have to face the computational bottleneck of the integral operators. To circumvent this bottleneck, we will employ the MLFMM outlined in Chapter 3 tailored for the FE-BI method. The next section outlines the adaptation of the MLFMM to the hybrid FE-BI system described above.

4.3 MLFMM for FE-BI

If we assume that the outer boundary s of the solution domain v does not have a perfectly magnetically conducting (PMC) boundary (which is the case for realistic structures), we can make use of the fact that the tangential magnetic field being non-zero over the entire surface s and tangential electric field being zero over the PEC portions of s along with the fact that the same surface basis functions being used for both tangential fields, to devise an efficient MLFMM approach.

For the general FE-BI formulation, two surface unknowns are needed for the solution. These surface unknowns occupy the same physical location and thus a single FMM clustering can be used to reduce CPU time and memory. Specifically, we can use the clustering information for the surface magnetic field, i.e. $\hat{n} \times \mathbf{H}$ for the matrix $[Q]$ and use the information in the matrix-vector product with the $[P]$ matrix. Furthermore, since the same basis functions are used for both $\hat{n} \times \mathbf{H}$ and $\mathbf{E} \times \hat{n}$, only the signature functions for the tangential magnetic field basis functions (i.e. electric currents) need to be computed. These two facts provide considerable memory savings since the same translation operators and signature functions are reused for both surface unknowns. Following the same approach as outlined in Chapter 3, we decompose the far-field portions of the $[P]$ and $[Q]$ matrices in the spectral domain as

$$P_{ji} \approx \frac{ik}{4\pi} \int d^2\hat{k} \mathbf{V}_{jm}^{(P)}(\hat{k}) \cdot T_L(kr_{mm'}, \hat{k} \cdot \hat{r}_{mm'}) \mathbf{V}_{im'}^{*}(\hat{k}), \tag{4.20}$$

and

$$Q_{ji} \approx \frac{ik}{4\pi} \int d^2\hat{k} \mathbf{V}_{jm}^{(Q)}(\hat{k}) \cdot T_L(kr_{mm'}, \hat{k} \cdot \hat{r}_{mm'}) \mathbf{V}_{im'}^{*}(\hat{k}), \tag{4.21}$$

with

$$\begin{aligned} \mathbf{V}_{jm}^{(P)}(\hat{k}) &= -\alpha \mathbf{V}_{jm}^{(2)}(\hat{k}) + (1-\alpha)\mathbf{V}_{jm}^{(4)}(\hat{k}), \\ \mathbf{V}_{jm}^{(Q)}(\hat{k}) &= \alpha\eta_0 \mathbf{V}_{jm}^{(1)}(\hat{k}) + (1-\alpha)\eta_0 \mathbf{V}_{jm}^{(3)}(\hat{k}), \end{aligned} \tag{4.22}$$

where

$$
\begin{aligned}
\mathbf{V}_{im'}(\hat{k}) &= \int_S d\mathbf{r}' e^{i\mathbf{k}\cdot\mathbf{r}_{im'}} \left[\bar{\mathbf{I}} - \hat{k}\hat{k}\right] \cdot \mathbf{b}_i(\mathbf{r}_i), \\
\mathbf{V}^{(1)}_{jm}(\hat{k}) &= \int_S d\mathbf{r} e^{i\mathbf{k}\cdot\mathbf{r}_{jm}} \left[\bar{\mathbf{I}} - \hat{k}\hat{k}\right] \cdot \mathbf{j}_j(\mathbf{r}_j), \\
\mathbf{V}^{(2)}_{jm}(\hat{k}) &= \int_S d\mathbf{r} e^{i\mathbf{k}\cdot\mathbf{r}_{jm}} \left[\bar{\mathbf{I}} - \hat{k}\hat{k}\right] \cdot \left[\hat{n} \times \mathbf{j}_j(\mathbf{r}_j)\right], \\
\mathbf{V}^{(3)}_{jm}(\hat{k}) &= \int_S d\mathbf{r} e^{i\mathbf{k}\cdot\mathbf{r}_{jm}} \left[\hat{n} \times \hat{k} \times \mathbf{j}_j(\mathbf{r}_j)\right], \\
\mathbf{V}^{(4)}_{jm}(\hat{k}) &= \int_S d\mathbf{r} e^{i\mathbf{k}\cdot\mathbf{r}_{jm}} \left[\hat{k} \times \mathbf{j}_j(\mathbf{r}_j)\right], \qquad (4.23)
\end{aligned}
$$

in which $\mathbf{b}_i$ represents either of the basis functions $\mathbf{j}_i$ or $\mathbf{m}_i$, and $\mathbf{j}_j$ represent the testing functions.

The above signature functions for all basis functions for the unknown surface magnetic field $\hat{n} \times \mathbf{H}$ are computed for the finest level of MLFMM and stored along with the translation operators for all levels for the oct-tree clustering. As noted in Chapter 3, interpolation is used to transfer the signature functions to the lower clustering levels. As can be realized, the $[E]$ and $[B]$ sub-matrices from the FEM portion of the equations are not affected by the MLFMM implementation.

At the iterative solution stage, the usual three sweeps of MLFMM are computed for all levels of clustering. As illustrated in Fig. 4.5, the three MLFMM sweeps

$$[P]\{E_s\} + [Q]\{H_s\} = \{b\}$$

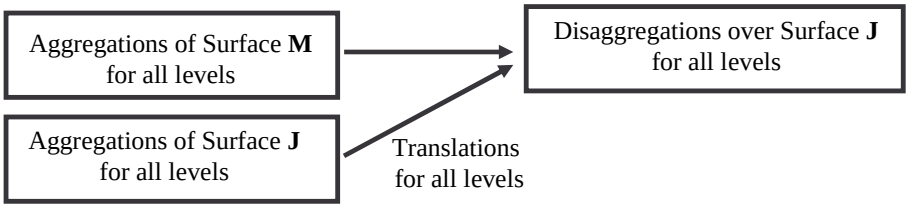


Figure 4.5. MLFMM procedure for the FE-BI system.

for each level are applied in the same manner as done for the SIE formulation. In the aggregation step, the signature functions of the children for each cluster are interpolated, shifted, and summed to form the signature of the parent group. This

signature is then translated onto all far-clusters whose parents lie in the near-field of each other. In the disaggregation step, these fields are distributed over the children of each cluster through shifting and interpolation. This procedure is done starting with the finest level, for which the group signatures are formed using the basis function signatures belonging to each group, and proceeding towards coarser levels to carry out the interpolation and shifting steps. Translations are applied at all levels and the reverse steps (i.e. shifting and interpolation) are carried out starting from the coarsest level proceeding toward to finest. At the finest level, the far-field contributions to the matrix-vector product are computed by testing local fields with the signatures of the testing functions belonging to each group. In this multilevel manner, the complexity of the matrix-vector product is reduced down to $O(N \log N)$ from the conventional $O(N^2)$, where N represented the size of the $[Q]$ matrix. The advantages and savings of the MLFMM implementation are demonstrated below along with a validation of the approach for electrically large inhomogeneous structures.

4.4 FE-BI-MLFMM Validation

The validation of the outlined FE-BI method is given in this section. The examples are chosen such that they have an analytical solution that can be used as a reference.

The first example is a solid dielectric sphere of $\epsilon_r = 1.75 + 0.3i$. This is an electrically small problem having a radius of only $0.2\lambda_0$ where λ_0 is the free-space wavelength at the solution frequency. The FE-BI solution for two different volume tessellations is depicted in Fig. 4.6 along with the reference Mie solution. As observed in the figure, the FE-BI solution converges to the reference Mie series result as the mesh density is increased. The convergence behavior of the BiCG iterative solver [18] used for the denser mesh problem (with 512 hexahedra) was observed to be extremely poor. Quantitatively, the problem resulted in 1176 volume unknowns and 768 surface unknowns. Using a diagonal preconditioner, the BiCG solver converged in 302 iter-

ations (executing 2 matrix-vector products per iteration) to within an error of 10^{-2}. Similar poor convergence behavior is also observed for our second example, that of

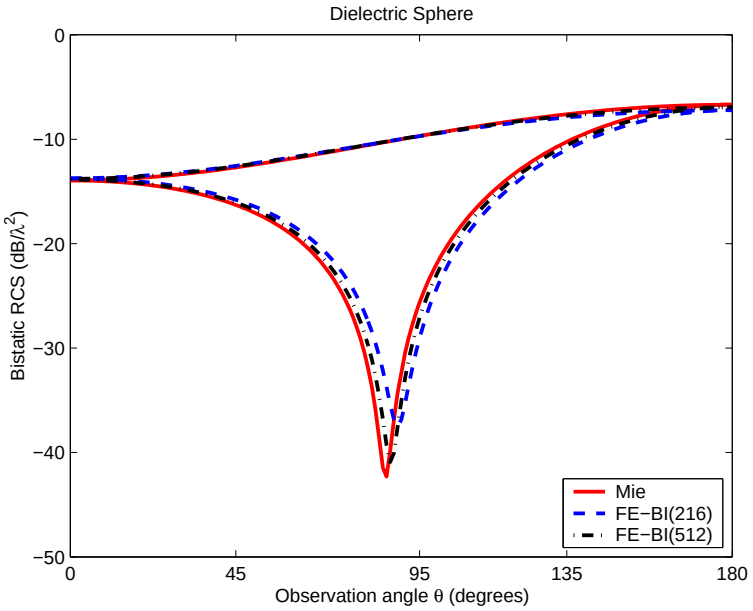


Figure 4.6. Bistatic RCS of a dielectric sphere using the FE-BI method.

a dielectric coated PEC sphere. The outer radius of the coated sphere is $0.75\lambda_0$ and the coating itself has a thickness of $0.075\lambda_0$ with $\epsilon_r = 1.75 + 0.3i$. A total of 384 curvilinear hexahedra were used to model the coating and this problem resulted in a higher number of surface unknowns (1536) and fewer FE unknowns (386). The bistatic RCS results for both polarizations of the incident field are given in Fig. 4.7. The BiCG solver with diagonal preconditioning took 85 iterations to converge to an error of 10^{-2}. These observations suggest that the convergence behavior of the FE-BI system is severely deteriorated by the presence of large FE domains. Such poor convergence behavior is due to the highly heterogeneous nature of the generated FE-BI system, which is partly sparse due to the FE domain and partly dense due to the BI operators. One remedy is to use better preconditioners. However, the specific choice and performance of the preconditioner depends on the geometry under investigation and puts a computational burden on the solver when electrically large problems are considered. Nevertheless, when a very large number of decoupled

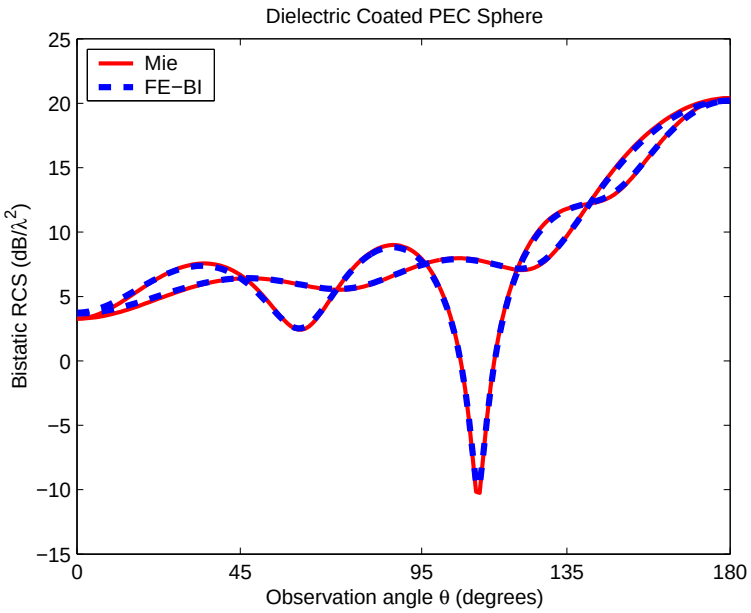


Figure 4.7. Bistatic RCS of a coated sphere using the FE-BI method.

FEM domains are considered, as in the case with large finite antenna arrays, efficient block diagonal preconditioning using the self FE-BI matrix of each antenna element has been demonstrated to considerably improve convergence [59] leading to a very successful application of the method.

When the FE-BI system is not exceedingly large, a direct solution method using a decoupling approach as done in [19] has been commonly used. However, for electrically large scatterers of arbitrary shape, the poor iterative convergence behavior renders MLFMM savings useless. Nevertheless, as a validation of the outlined MLFMM implementation, the same coated sphere problem was solved using MLFMM. The result is depicted in Fig. 4.8 and is in good agreement with the Mie solution as well as the conventional FE-BI solution of Fig. 4.7. To verify our comments on convergence, we remark that the MLFMM solution converged in 112 iterations as compared to 85 iterations with the conventional FE-BI method. This also indicates that the FE-BI system is very badly conditioned and small numerical errors in the MLFMM process may result in different convergence rates. The total execution time of the MLFMM solver was more than twice that of the FE-BI solution. However, we must note here

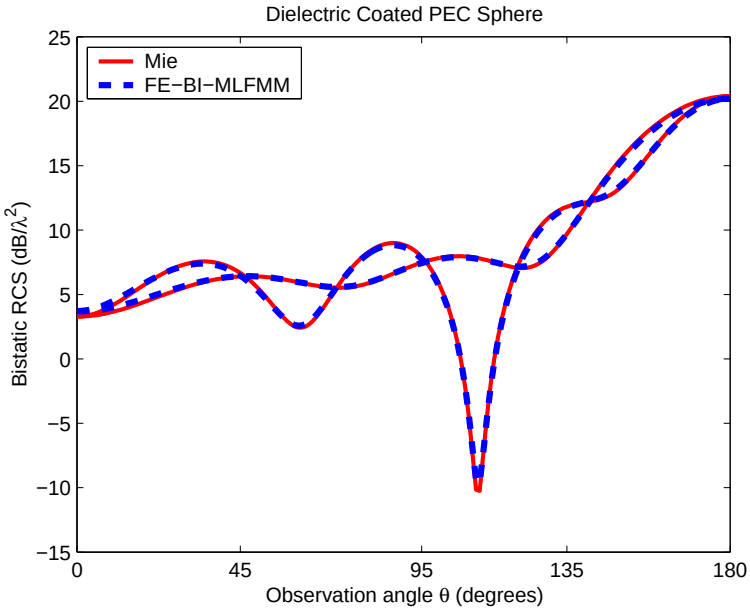


Figure 4.8. Bistatic RCS of a coated sphere using the MLFMM accelerated FE-BI.

that the MLFMM-FE-BI solver has a lower computational complexity and should be expected to be faster than the conventional FE-BI as the problem size increases.

Poor conditioning of the FE-BI system and the need to use iterative solvers in the context of MLFMM is a major reason for considering VIEs as an alternative to analyzing the electromagnetic behavior of inhomogeneous bodies. The next chapter outlines the governing integral equations and MLFMM solutions of these integral equations.

CHAPTER 5

Volume Integral Equations

Scattering by dielectric and composite volumes is of significant interest for radar cross section (RCS) and electromagnetic compatibility/electromagnetic interference (EMI/EMC) evaluation of modern composite vehicles and structures. Over the past three decades several authors considered the scattering by dielectric volumes using the MoM [30]. However, because the number of unknowns needed to formulate the moment method for volumetric dielectrics quickly grows to many thousands, even for a structure as small as $1\lambda^3$ in volume, the method is difficult to use for practical applications. An alternative is to use a surface integral formulation for dielectrics [60], but this is only applicable to piecewise–homogeneous structures. To alleviate the $O(N^3)$ CPU bottleneck for modeling dielectric volumes, k-space methods utilizing iterative solution approaches were introduced. However, by virtue of the FFT, these often lack geometrical conformality.

With the recent success of fast methods, such as the MLFMM [8, 44], it is important to re-examine VIEs as an alternative to modeling inhomogeneous volumetric structures (see [61] for an application of VIEs to indoor radio wave propagation). In this chapter, we develop a VIE solution method using curvilinear hexahedral elements for conformal modeling of penetrable structures and introduce a MLFMM solution of the pertinent VIEs. It is demonstrated that even piecewise constant basis functions provide excellent accuracy at a nominal sampling rates when conformal elements are used. Results are given to show the accuracy of the method as well as the CPU and

memory savings provided by the MLFMM.

5.1 Volume Integral Equation: General Formulation

The derivation of an integral equation for the electric field in an inhomogeneous (possibly anisotropic) medium involves standard mathematical manipulations of Maxwell's equations. We start with the vector wave equation

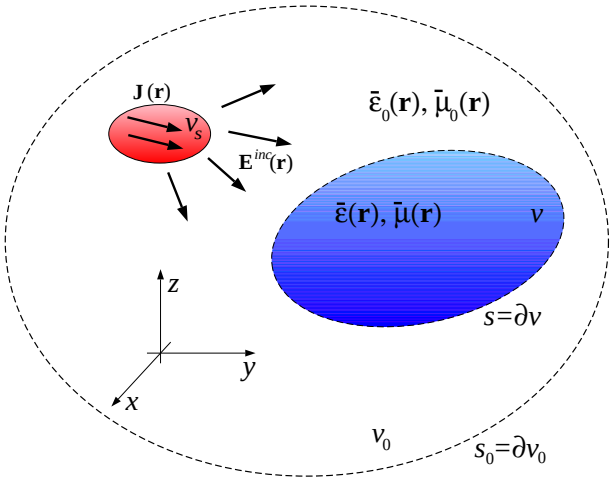


Figure 5.1. Geometrical setup for the VIE derivations.

$$\nabla \times \left[\overline{\mu}^{-1}(\mathbf{r}) \cdot \nabla \times \mathbf{E}(\mathbf{r})\right] - \omega^2 \overline{\epsilon}(\mathbf{r}) \cdot \mathbf{E}(\mathbf{r}) = i\omega \mathbf{J}(\mathbf{r}) \tag{5.1}$$

where $\overline{\epsilon}(\mathbf{r})$ and $\overline{\mu}(\mathbf{r})$ are the dyadic material parameters describing the electromagnetic behavior of the solution domain and $\omega = 2\pi f$ is the operation frequency. Here, $\mathbf{J}(\mathbf{r})$ represents all impressed current sources and/or external excitations (see Fig. 5.1). To arrive at a wave equation in terms of the electric field in free space (or possibly an anisotropic inhomogeneous background medium), we subtract $\nabla \times \left[\overline{\mu}_0^{-1}(\mathbf{r}) \cdot \nabla \times \mathbf{E}(\mathbf{r})\right] - \omega^2 \overline{\epsilon}_0(\mathbf{r}) \cdot \mathbf{E}(\mathbf{r})$ from both sides of (5.1). Doing so and rearranging, we get

$$\begin{aligned} \nabla \times \left[\overline{\mu}_0^{-1}(\mathbf{r}) \cdot \nabla \times \mathbf{E}(\mathbf{r})\right] - \omega^2 \overline{\epsilon}_0(\mathbf{r}) \cdot \mathbf{E}(\mathbf{r}) &= i\omega \mathbf{J}(\mathbf{r}) + \omega^2 \left[\overline{\epsilon}(\mathbf{r}) - \overline{\epsilon}_0(\mathbf{r})\right] \cdot \mathbf{E}(\mathbf{r}) \\ &- \nabla \times \left\{\left[\overline{\mu}^{-1}(\mathbf{r}) - \overline{\mu}_0^{-1}(\mathbf{r})\right] \cdot \nabla \times \mathbf{E}(\mathbf{r})\right\}, \end{aligned} \tag{5.2}$$

in which $\overline{\epsilon}_0(\mathbf{r}) = \epsilon_0\overline{\mathbf{I}}$ and $\overline{\mu}_0(\mathbf{r}) = \mu_0\overline{\mathbf{I}}$ refer to the permittivity and permeability of the background medium, chosen as free-space here, and $\overline{\mathbf{I}}$ is the identity dyad. The right hand side of (5.2) can now be identified as a source term and accounts for the presence of the inhomogeneous medium. The solution for $\mathbf{E}(\mathbf{r})$ can now be carried out using the dyadic Green's function satisfying

$$\nabla \times \left[\overline{\mu}_0^{-1}(\mathbf{r}) \cdot \nabla \times \overline{\mathbf{G}}(\mathbf{r}, \mathbf{r}')\right] - \omega^2 \overline{\epsilon}_0(\mathbf{r}) \cdot \overline{\mathbf{G}}(\mathbf{r}, \mathbf{r}') = \overline{\mu}_0^{-1}(\mathbf{r})\delta(\mathbf{r} - \mathbf{r}'). \tag{5.3}$$

The Green's function takes the well-known form

$$\overline{\mathbf{G}}(\mathbf{r}, \mathbf{r}') = \left(\overline{\mathbf{I}} + \frac{1}{k_0^2}\nabla\nabla\right) g(\mathbf{r}, \mathbf{r}'), \quad g(\mathbf{r}, \mathbf{r}') = \frac{e^{ik_0|\mathbf{r}-\mathbf{r}'|}}{|\mathbf{r} - \mathbf{r}'|}, \tag{5.4}$$

where $k_0 = \omega\sqrt{\epsilon_0\mu_0}$. Using the right hand side of (5.2) as the source, we readily obtain [62]

$$\begin{aligned} \mathbf{E}(\mathbf{r}) &= i\omega\mu_0 \int_{v+v_0} d\mathbf{r}' \overline{\mathbf{G}}(\mathbf{r}, \mathbf{r}') \cdot \mathbf{J}(\mathbf{r}') \\ &+ k_0^2 \int_{v+v_0} d\mathbf{r}' \overline{\mathbf{G}}(\mathbf{r}, \mathbf{r}') \cdot (\overline{\epsilon}_r(\mathbf{r}') - \overline{\mathbf{I}}) \cdot \mathbf{E}(\mathbf{r}') \\ &- \int_{v+v_0} d\mathbf{r}' \overline{\mathbf{G}}(\mathbf{r}, \mathbf{r}') \cdot \nabla' \times \left[\left(\overline{\mu}_r^{-1}(\mathbf{r}') - \overline{\mathbf{I}}\right) \cdot \nabla' \times \mathbf{E}(\mathbf{r}')\right] \end{aligned} \tag{5.5}$$

where $\overline{\epsilon}_r = \overline{\epsilon}/\epsilon_0$ and $\overline{\mu}_r = \overline{\mu}/\mu_0$. The first term in the above is identified as the excitation field generated by the main $\mathbf{J}(\mathbf{r})$, viz.

$$\mathbf{E}^{inc}(\mathbf{r}) = i\omega\mu_0 \int_{v_s} d\mathbf{r}' \overline{\mathbf{G}}(\mathbf{r}, \mathbf{r}') \cdot \mathbf{J}(\mathbf{r}') \tag{5.6}$$

where the integration domain $(v + v_0)$ is conveniently replaced by v_s since this represents the only volume over which $\mathbf{J}(\mathbf{r})$ is non-zero. The other two terms vanish when $\overline{\epsilon}(\mathbf{r}) = \epsilon_0\overline{\mathbf{I}}$, $\overline{\mu}(\mathbf{r}) = \mu_0\overline{\mathbf{I}}$ and consequently represent the scattering due to v.

To relax the differentiability requirement on $\left(\overline{\mu}_r^{-1}(\mathbf{r}) - \overline{\mathbf{I}}\right) \cdot \nabla' \times \mathbf{E}(\mathbf{r}')$ in the third term in (5.5), we can make use of the divergence theorem ($\hat{n}$ is outward from v)

$$\int_v d\mathbf{r} \nabla \cdot \left[\overline{\mathbf{G}} \times \mathbf{y}\right] = \oint_s d\mathbf{r} \hat{n} \cdot \left[\overline{\mathbf{G}} \times \mathbf{y}\right] = \oint_s d\mathbf{r} \overline{\mathbf{G}} \cdot (\hat{n} \times \mathbf{y}) \tag{5.7}$$

and the dyadic identity

$$\nabla \cdot \left[\overline{\mathbf{G}} \times \mathbf{y}\right] = \overline{\mathbf{G}} \cdot \nabla \times \mathbf{y} - \nabla \times \overline{\mathbf{G}} \cdot \mathbf{y} \tag{5.8}$$

where $\mathbf{y}$ is an arbitrary vector chosen here to be $\mathbf{y} = \left(\overline{\mu}_r^{-1}(\mathbf{r}) - \overline{\mathbf{I}}\right) \cdot \nabla' \times \mathbf{E}(\mathbf{r}')$. This manipulation leads to

$$\begin{aligned} \mathbf{E}(\mathbf{r}) &= \mathbf{E}^{inc}(\mathbf{r}) + k_0^2 \int_{v+v_0} d\mathbf{r}' \overline{\mathbf{G}}(\mathbf{r}, \mathbf{r}') \cdot (\overline{\epsilon}_r(\mathbf{r}') - \overline{\mathbf{I}}) \cdot \mathbf{E}(\mathbf{r}') \\ &- \int_{v+v_0} d\mathbf{r}' \nabla' \times \overline{\mathbf{G}}(\mathbf{r}, \mathbf{r}') \cdot \left[\left(\overline{\mu}_r^{-1}(\mathbf{r}') - \overline{\mathbf{I}}\right) \cdot \nabla' \times \mathbf{E}(\mathbf{r}')\right] \\ &+ \oint_{s_0} d\mathbf{r}' \overline{\mathbf{G}}(\mathbf{r}, \mathbf{r}') \cdot \hat{n}' \times \left[\left(\overline{\mu}_r^{-1}(\mathbf{r}') - \overline{\mathbf{I}}\right) \cdot \nabla' \times \mathbf{E}(\mathbf{r}')\right]. \end{aligned} \tag{5.9}$$

In comparison to (5.5), this does not include any differentiation on $\left(\overline{\mu}_r^{-1}(\mathbf{r}') - \overline{\mathbf{I}}\right) \cdot \nabla' \times \mathbf{E}(\mathbf{r}')$. However, a surface integral is introduced for enclosing $v + v_0$. As $s_0 \to \infty$, the radiation condition implies that the surface integral vanishes, giving

$$\begin{aligned} \mathbf{E}(\mathbf{r}) &= \mathbf{E}^{inc}(\mathbf{r}) + k_0^2 \int_{v+v_0} d\mathbf{r}' \overline{\mathbf{G}}(\mathbf{r}, \mathbf{r}') \cdot (\overline{\epsilon}_r(\mathbf{r}') - \overline{\mathbf{I}}) \cdot \mathbf{E}(\mathbf{r}') \\ &- \int_{v+v_0} d\mathbf{r}' \nabla' \times \overline{\mathbf{G}}(\mathbf{r}, \mathbf{r}') \cdot \left[\left(\overline{\mu}_r^{-1}(\mathbf{r}') - \overline{\mathbf{I}}\right) \cdot \nabla' \times \mathbf{E}(\mathbf{r}')\right]. \end{aligned} \tag{5.10}$$

An alternative way to derive (5.10) is through the standard use of equivalent electric and magnetic currents. In accordance with [27], we could introduce the volumetric electric and magnetic current density sources

$$\mathbf{J}_{eq}(\mathbf{r}) = -i\omega\epsilon_0 \left[\overline{\epsilon}_r(\mathbf{r}) - \overline{\mathbf{I}}\right] \cdot \mathbf{E}(\mathbf{r}), \quad \mathbf{M}_{eq}(\mathbf{r}) = -\left[\overline{\mu}_r(\mathbf{r}) - \overline{\mathbf{I}}\right] \cdot \mathbf{H}(\mathbf{r}) \tag{5.11}$$

to represent the scattering by the dielectric volume. These sources radiate (in free space) and are responsible for the scattered fields

$$\mathbf{E}^{scat}(\mathbf{r}) = i\omega\mu_0 \int_v d\mathbf{r}' \left[\overline{\mathbf{G}}(\mathbf{r}, \mathbf{r}') \cdot \mathbf{J}_{eq}(\mathbf{r}') + \nabla' \times \overline{\mathbf{G}}(\mathbf{r}, \mathbf{r}') \cdot \mathbf{M}_{eq}(\mathbf{r}')\right] \tag{5.12}$$

and

$$\mathbf{E}(\mathbf{r}) = \mathbf{E}^{inc}(\mathbf{r}) + \mathbf{E}^{scat}(\mathbf{r}) \tag{5.13}$$

with the dual expressions valid for $\mathbf{H}^{scat}(\mathbf{r})$. Equation (5.13) and its dual yield a pair of integral equations for the solution of the unknown volume equivalent currents $\mathbf{J}_{eq}(\mathbf{r})$ and $\mathbf{M}_{eq}(\mathbf{r})$ (or $\mathbf{H}(\mathbf{r})$ and $\mathbf{E}(\mathbf{r})$, respectively).

The above formulation involving equivalent volume currents was used by Richmond [29] for two dimensional applications, and results in six scalar unknowns per volume location for problems involving having non-trivial permittivity and permeability. In contrast, (5.10) involves only a single vector field unknown (three scalar unknowns) per volume location. However, (5.10) requires differentiation (curl) of the unknown field $\mathbf{E}(\mathbf{r})$ in much the same way done in the finite element method [18]. One could actually choose to remove the differentiation from $\mathbf{E}(\mathbf{r})$ but would then obtain an integrand which involves differentiation of the permeability across the volume. This was done in [32] and although the final expression avoids the curl of the unknown $\mathbf{E}(\mathbf{r})$, the need to differentiate $\overline{\mu}_r(\mathbf{r})$ is undesirable for numerical implementation when $\overline{\mu}_r(\mathbf{r})$ is not available in analytic form. Clearly, among the representations (5.10), (5.12), and Equation (28) of [32], the one given by (5.10) is most attractive for numerical implementation.

5.2 MoM formulation of the VIE for dielectrics

For the specific case when we are dealing only with a dielectric inhomogeneity ($\mu_r = 1$), (5.10) simplifies to

$$\mathbf{E}(\mathbf{r}) = \mathbf{E}^{inc}(\mathbf{r}) + k_0^2 \int_v d\mathbf{r}' \overline{\mathbf{G}}(\mathbf{r}, \mathbf{r}') \cdot (\overline{\epsilon}_r(\mathbf{r}') - \overline{\mathbf{I}}) \cdot \mathbf{E}(\mathbf{r}'). \tag{5.14}$$

We proceed with the standard MoM expansion of the unknown electric field intensity $\mathbf{E}$ as

$$\mathbf{E}(\mathbf{r}) = \sum_{i=1}^{N} x_i \mathbf{e}_i(\mathbf{r}) \tag{5.15}$$

and Galerkin's testing to arrive at a matrix equation $[Z]\{x\} = \{b\}$ where,

$$Z_{ji} = \langle \mathbf{e}_j(\mathbf{r}), \mathbf{e}_i(\mathbf{r}) \rangle - k_0^2 \langle \mathbf{e}_j(\mathbf{r}), \int_v d\mathbf{r}' \overline{\mathbf{G}}(\mathbf{r}, \mathbf{r}') \cdot \left(\overline{\epsilon}_r - \overline{\mathbf{I}} \right) \cdot \mathbf{e}_i(\mathbf{r}') \rangle. \tag{5.16}$$

The right hand side of the matrix system in the parametric space takes the form $b_j = \langle \mathbf{e}_j(\mathbf{r}), \mathbf{E}^{inc}(\mathbf{r}) \rangle$. Similar to the SIE implementation, the hyper-singularity in (5.16) is

relaxed using the divergence theorem, resulting

$$\begin{aligned} Z_{ji} &= \int_v d\mathbf{r}\mathbf{e}_j(\mathbf{r})\cdot\mathbf{e}_i(\mathbf{r}) - k_0^2\int_v d\mathbf{r}\mathbf{e}_j(\mathbf{r})\cdot\int_v d\mathbf{r}' g(\mathbf{r},\mathbf{r}')\left(\overline{\epsilon}_r - \overline{\mathbf{I}}\right)\cdot\mathbf{e}_i(\mathbf{r}') \quad (5.17)\\ &- \int_v d\mathbf{r}\nabla\cdot\mathbf{e}_j(\mathbf{r})\int_v d\mathbf{r}' g(\mathbf{r},\mathbf{r}')\nabla'\cdot\left[\left(\overline{\epsilon}_r - \overline{\mathbf{I}}\right)\cdot\mathbf{e}_i(\mathbf{r}')\right]\\ &+ \oint_s d\mathbf{r}\mathbf{e}_j(\mathbf{r})\cdot\int_v d\mathbf{r}'\nabla g(\mathbf{r},\mathbf{r}')\nabla'\cdot\left[\left(\overline{\epsilon}_r - \overline{\mathbf{I}}\right)\cdot\mathbf{e}_i(\mathbf{r}')\right]. \end{aligned}$$

To further simplify this, we assume that the permittivity is constant within each element hence, we can rewrite (5.17) over a pair of source and testing hexahedra as

$$\begin{aligned} Z_{ji} &= \int_v d\mathbf{r}\mathbf{e}_j(\mathbf{r})\cdot\mathbf{e}_i(\mathbf{r}) - k_0^2\int_v d\mathbf{r}\mathbf{e}_j(\mathbf{r})\cdot\left(\overline{\epsilon}_r - \overline{\mathbf{I}}\right)\int_v d\mathbf{r}' g(\mathbf{r},\mathbf{r}')\cdot\mathbf{e}_i(\mathbf{r}') \quad (5.18)\\ &- \int_v d\mathbf{r}\nabla\cdot\mathbf{e}_j(\mathbf{r})\left(\overline{\epsilon}_r - \overline{\mathbf{I}}\right)\int_v d\mathbf{r}' g(\mathbf{r},\mathbf{r}')\nabla'\cdot\mathbf{e}_i(\mathbf{r}')\\ &+ \oint_s d\mathbf{r}\mathbf{e}_j(\mathbf{r})\cdot\left(\overline{\epsilon}_r - \overline{\mathbf{I}}\right)\cdot\int_v d\mathbf{r}'\nabla g(\mathbf{r},\mathbf{r}')\nabla'\cdot\mathbf{e}_i(\mathbf{r}'). \end{aligned}$$

Numerical evaluation of the above matrix elements are again carried out using a suitable order Gaussian quadrature. The singular integral appearing in the self-cell interactions is treated separately using an annihilation technique similar to that for SIEs. The annihilation technique for 3 dimensions is given in Appendix B.

We have explored several different orders of volumetric basis functions to be used in (5.15). Below, we describe in detail three different orders of basis functions, each having specific qualities attractive for volumetric modeling. For purely dielectric bodies, all three basis functions are observed to generate accurate results. However, when the target geometry has both a non-trivial permittivity and permeability, as discussed in the next section, the only possible choice is to use the 2^{nd} order basis functions due to the need to evaluate $\nabla\times\mathbf{E}$ in (5.10).

5.2.1 Zeroth-Order Volumetric Basis Functions

For each volume element, 3 basis functions associated with the 3 parametric directions are defined as

$$\mathbf{e}_1 = \frac{1}{\sqrt{G}}\mathbf{a}_u, \qquad \mathbf{e}_2 = \frac{1}{\sqrt{G}}\mathbf{a}_u, \qquad \mathbf{e}_3 = \frac{1}{\sqrt{G}}\mathbf{a}_u. \quad (5.19)$$

It is easy to verify, with the above definition, that the divergence of the basis function inside the element is identically zero. This is a property also shared by the electric field inside a uniform dielectric region. Another advantage of this expansion is that the basis functions are defined only on separate elements and hence this expansion allows for a natural decomposition of the problem into different domains. It is important to note that the normal components of the basis functions are allowed to be discontinuous as well as the tangential components. These basis functions are conformal generalizations of those presented in [23] for rectangular domains. Fig. 5.2 illustrates the behavior of the vector field inside the curvilinear hexahedron at two different isoparametric cuts.

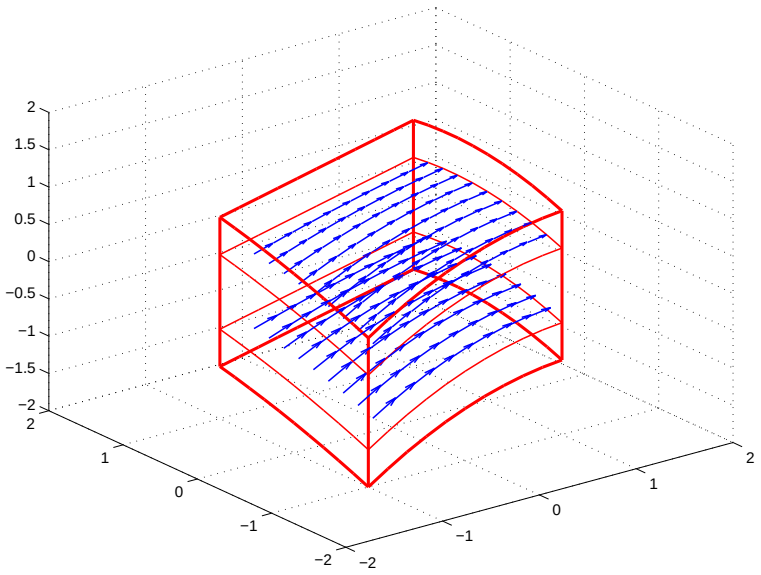


Figure 5.2. Generic illustration of zeroth-order electric field basis functions.

5.2.2 First-Order Volumetric Basis Functions

For each volume element, 6 basis functions associated with the 6 faces of the element are defined as

$$\begin{aligned} \mathbf{e}_1 &= \frac{1}{\sqrt{G}}(u)\mathbf{a}_u, & \mathbf{e}_2 &= \frac{1}{\sqrt{G}}(1-u)\mathbf{a}_u, & \mathbf{e}_3 &= \frac{1}{\sqrt{G}}(v)\mathbf{a}_v \\ \mathbf{e}_4 &= \frac{1}{\sqrt{G}}(1-v)\mathbf{a}_v, & \mathbf{e}_5 &= \frac{1}{\sqrt{G}}(w)\mathbf{a}_w, & \mathbf{e}_6 &= \frac{1}{\sqrt{G}}(1-w)\mathbf{a}_w. \end{aligned} \tag{5.20}$$

With this definition, the divergence of the basis functions is found to be constant inside the elements in the parametric space, viz. $\nabla \cdot \mathbf{e}_i = 1/\sqrt{G}$. The advantage of this expansion is that the normal components of the basis functions, defined across pairs of elements sharing a common face, are continuous. Fig. 5.3 illustrates the behavior of the vector field inside a pair of curvilinear hexahedra (sharing a common face) at two different isoparametric cuts.

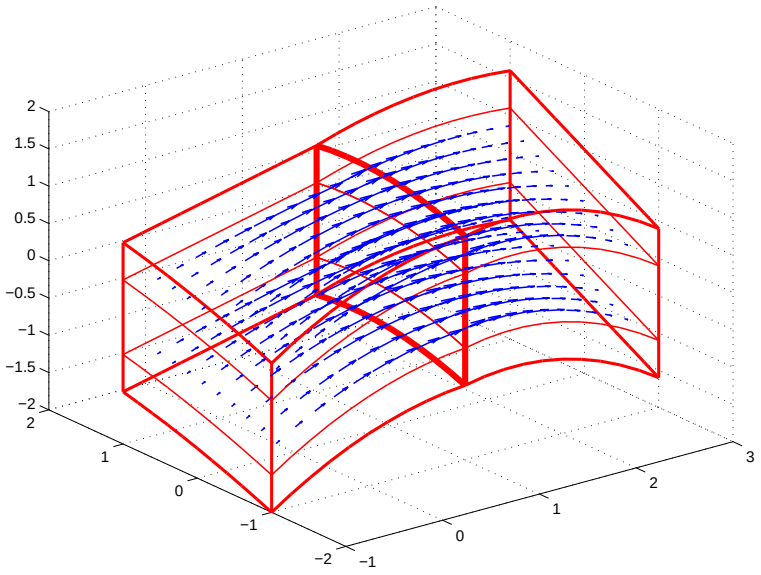


Figure 5.3. Generic illustration of first-order electric field basis functions.

5.2.3 Second-Order Volumetric Basis Functions

For each volume element, 12 basis functions associated with the 12 edges of the element are defined as

$$\begin{aligned}
\mathbf{e}_1 &= \frac{1}{\sqrt{G}}(v)(w)\mathbf{a}_u, & \mathbf{e}_2 &= \frac{1}{\sqrt{G}}(1-v)(w)\mathbf{a}_u, \\
\mathbf{e}_3 &= \frac{1}{\sqrt{G}}(v)(1-w)\mathbf{a}_u, & \mathbf{e}_4 &= \frac{1}{\sqrt{G}}(1-v)(1-w)\mathbf{a}_u, \\
\mathbf{e}_5 &= \frac{1}{\sqrt{G}}(u)(w)\mathbf{a}_v, & \mathbf{e}_6 &= \frac{1}{\sqrt{G}}(1-u)(w)\mathbf{a}_v, \\
\mathbf{e}_7 &= \frac{1}{\sqrt{G}}(u)(1-w)\mathbf{a}_v, & \mathbf{e}_8 &= \frac{1}{\sqrt{G}}(1-u)(1-w)\mathbf{a}_v, \\
\mathbf{e}_9 &= \frac{1}{\sqrt{G}}(u)(v)\mathbf{a}_w, & \mathbf{e}_{10} &= \frac{1}{\sqrt{G}}(1-u)(v)\mathbf{a}_w, \\
\mathbf{e}_{11} &= \frac{1}{\sqrt{G}}(u)(1-v)\mathbf{a}_w, & \mathbf{e}_{12} &= \frac{1}{\sqrt{G}}(1-u)(1-v)\mathbf{a}_w.
\end{aligned} \tag{5.21}$$

Switching to a covariant unitary vector expansion for the VIE formulation as opposed to a contravariant form as in FE-BI basis functions has the advantage of providing zero divergence inside the element and is the main reason for choosing it. As in the case with zeroth-order basis functions, this property conforms with that of the electric field. As opposed to a contravariant expansion (i.e. a covariant projection form), this expansion does not have continuous tangential components across the common faces of the elements that share a common edge. Fig. 5.4 illustrates the behavior of the vector field inside a pair of curvilinear hexahedra (sharing a common edge) at two different isoparametric cuts.

5.2.4 Validations for Dielectric Structures

Here, we show that all three different orders of volumetric basis functions defined above produce accurate results. As a test geometry, a dielectric spherical shell is considered. Fig. 5.5 depicts the computed RCS values for a dielectric shell of outer radius $0.2\lambda_0$ and of thickness $0.02\lambda_0$. All three results agree well with the analytical solution. However, we must note here that, although the same volume mesh is used

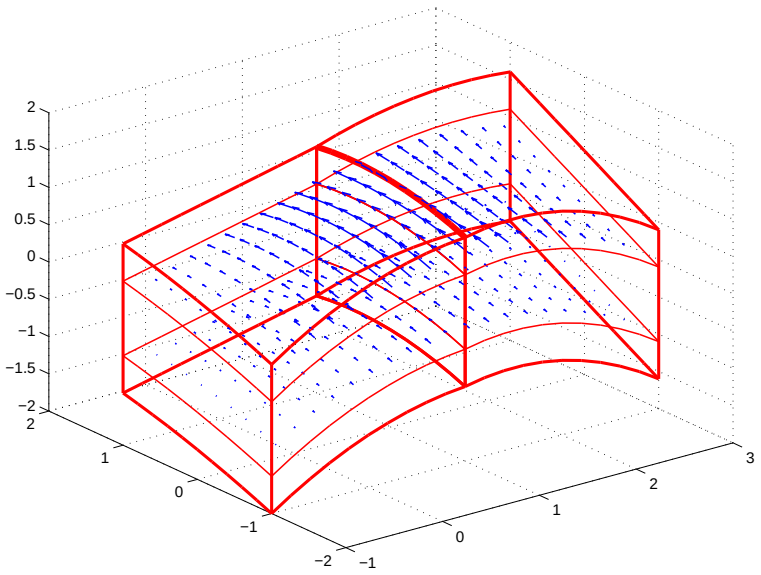


Figure 5.4. Generic illustration of second-order electric field basis functions.

in all three cases, the system sizes for different basis functions are inherently different. For this example, zeroth-order basis functions resulted in 648 unknowns, first-order basis functions resulted in 864 unknowns, and second-order basis functions resulted in 1082 unknowns. Correspondingly, the matrix fill times were different, the zeroth-order being the fastest and the second-order being the slowest.

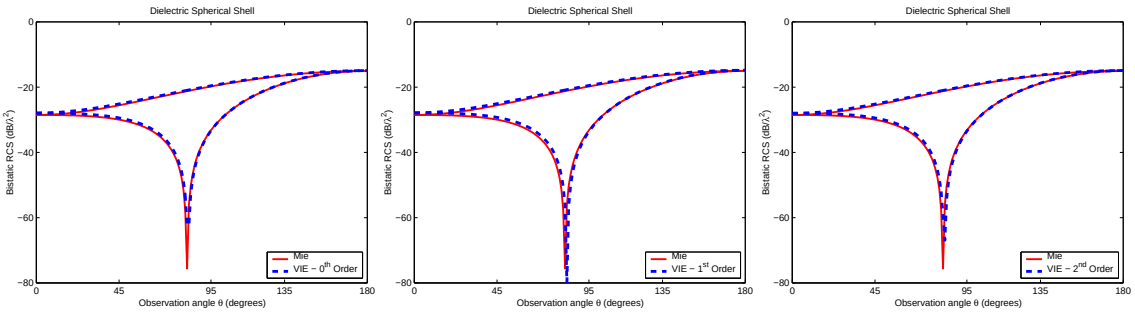


Figure 5.5. Bistatic RCS of a $0.2\lambda_0$ radius, $0.02\lambda_0$ thick spherical shell with $\epsilon_r = 2.2$ using three different orders of basis functions.

As an electrically larger problem, the same geometry was meshed using 384 hexahedra and the results at twice the frequency are given in Fig. 5.6. Again, all three methods produced very good agreement with the analytical solution.

In the case of a dielectric cube of side length $0.5\lambda_0$ and $\epsilon_r = 2.2$, we do not have

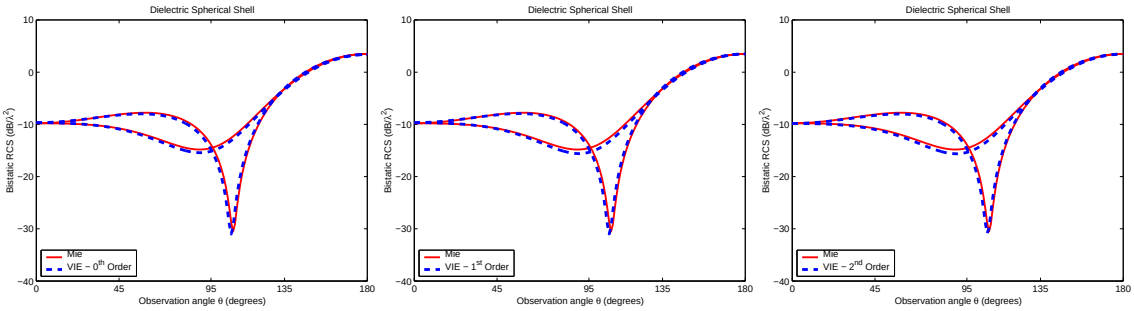


Figure 5.6. Bistatic RCS of a $0.4\lambda_0$ radius, $0.04\lambda_0$ thick spherical shell with $\epsilon_r = 2.2$ using three different orders of basis functions.

a closed form reference solution. However, the bistatic RCS of the dielectric cube using three different orders of basis functions is plotted in Fig. 5.7 and all results are observed to be in good to fair agreement.

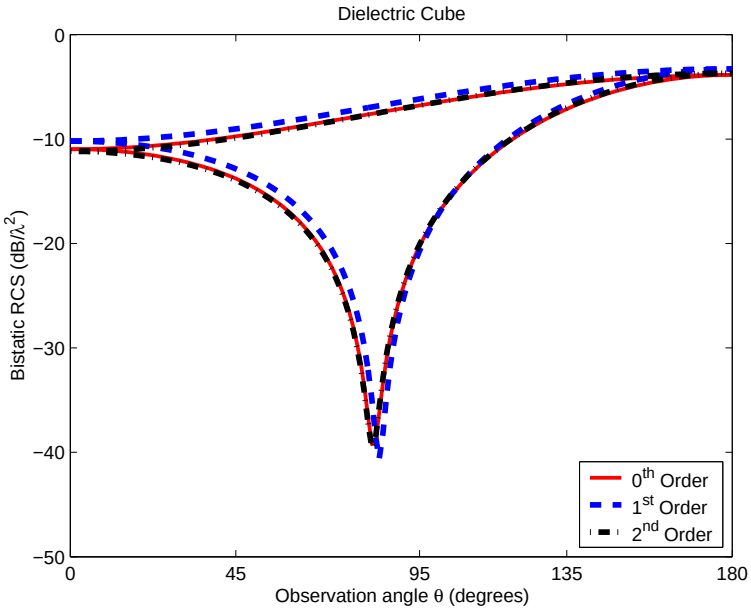


Figure 5.7. Bistatic RCS of a $0.5\lambda_0$ side length dielectric cube with $\epsilon_r = 2.2$ using three different orders of basis functions.

The three examples given above demonstrate that for dielectric scatterers the present integral equation solutions produce accurate RCS results regardless of the expansion order for the unknown electric field intensity. Based on this conclusion, we will adopt the lowest order basis function in the MLFMM implementation for volumetric dielectric bodies. This also allows for a physical domain decomposition since

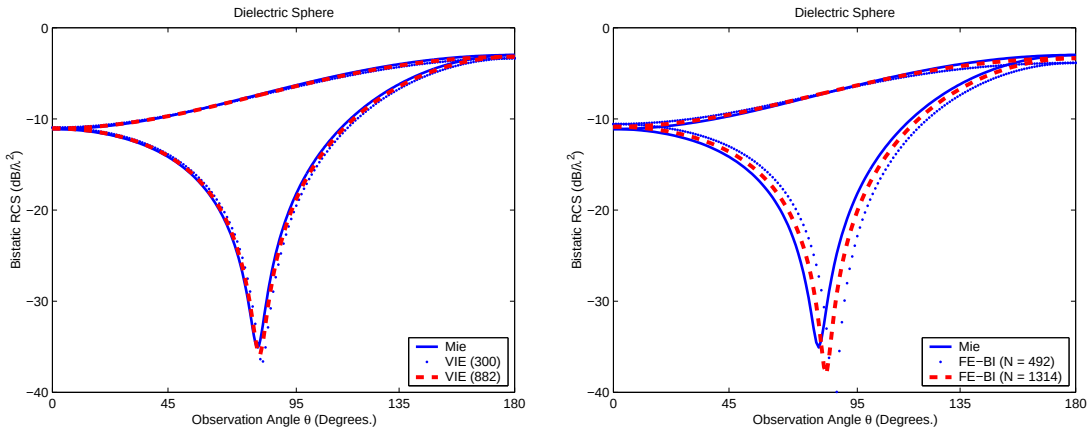
the basis functions are defined on physically distinct elements rather than extending over a collection of elements as in the case of first-order and second-order basis functions. However, when dealing with magnetic materials, second-order basis functions must inevitably be used due to the differentiability requirement on the unknown electric field intensity. This point will be discussed in detail in the following sections and the solution of the governing integral equation for magnetic and composite structures will be outlined. Before we proceed with the general composite material case, a comparison between the FE-BI and VIE methods in terms of accuracy and computational cost is considered. This is a preliminary comparison based on a single problem. Nevertheless, it provides insight regarding the significant differences between the two methods and the potential of VIE methods. Using the same hexahedral elements to model the geometry in both methods is a distinct feature of this comparison since the modeling error in both solutions is the same and the error performance is entirely governed by the numerical solutions of the respective methods.

5.3 A Comparison of VIE and FE-BI Methods

With the above validations of both FE-BI and VIE approaches to the same problem, we consider a comparison between the two methods for bistatic scattering by a dielectric sphere. Our goal is to provide an initial comparison of the solution performance for the VIE and FE-BI methods for this specific structure.

The considered sphere example refers to a purely dielectric problem. The sphere radius is 0.2λ and $\epsilon_r = 2.592$. As depicted in Fig. 5.8, the bistatic radar cross section (RCS) patterns calculated using the FE-BI and VIE methods are in good agreement with the exact Mie series data. However, the FE-BI required more unknowns to achieve the same level of accuracy. Thus, to better compare the VIE and FE-BI methods, we consider a comparison of the solution error vs. element edge length. Here, we note again that in both cases the same curvilinear hexahedra were

used. This comparison was done for the sphere example and resulted in the curves given in Fig. 5.9. They depict that the error is proportional to the square of the element's edge length for both methods. However, the VIE can afford larger elements for the same error ($\Delta_{VIE} \approx \sqrt{2}\Delta_{FE-BI}$). Also, as shown in Tab. 5.1, the VIE system converges much faster but the matrix fill time is significantly higher. The latter is a clear disadvantage of the VIE method, but the rapid convergence of its associated matrix system provides incentives for further examination of the method. It should be noted from Fig. 5.10 that the convergence behavior of FE-BI as a function of system size scales linearly, whereas for this problem the VIE convergence is much faster and almost independent of the rank of the matrix. This can be attributed to the fact that the FE-BI system is highly heterogeneous and thus the generated matrix is ill-conditioned, whereas the VIE is a second kind integral equation producing a well-conditioned matrix.



(a) (b)

Figure 5.8. Bistatic RCS of a dielectric sphere of radius 0.2λ ($\epsilon_r = 2.592$). (a) VIE Solution, (b) FE-BI Solution.

Fig. 5.11 depicts the percent error in the RCS solution for the same sphere as a function of the resulting systems for both methods. The VIE is again observed to have a better performance for a fixed system size.

On the basis of the above example, we can remark that the FE-BI method will

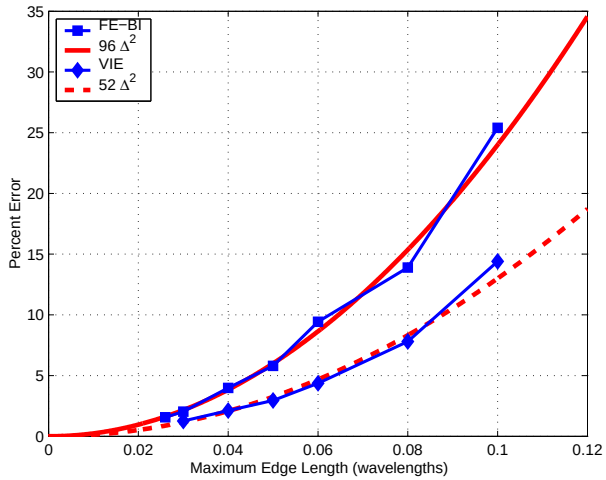


Figure 5.9. Convergence curves of the FE-BI and VIE solvers with respect to maximum edge length.

	Number of Unknowns	Matrix Fill Time (s.)	Number of Iterations	Time per Solution (s.)
VIE	300	107	11	0.41
VIE	882	1211	13	5.00
FE-BI	492	1.3	160	4.08
FE-BI	1314	6.3	354	50.14

Table 5.1. Performances of VIE and FE-BI formulations for the sphere.

be the choice method for many composite structures but the VIE offers advantages that should be further examined.

The CPU and memory requirements of the VIE can significantly be reduced by employing the MLFMM. In the next section, we outline the adaptation of the MLFMM to VIE formulations.

5.4 MLFMM for Volume Integral Equations

In this section, we describe the implementation of the MLFMM for dielectric structures using zeroth-order basis functions. The validations and performance assessments of the implementation is given for various different examples. We note here that the SIE implementation should be combined with the VIE implementation to broaden the range of applications that can be analyzed using integral equation ap-

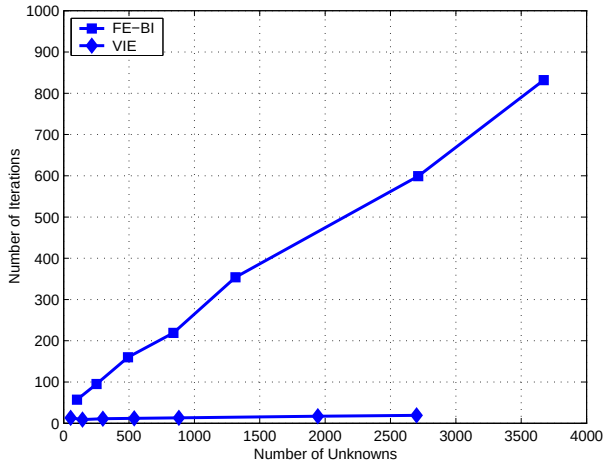


Figure 5.10. Iterative convergence of FE-BI and VIE solvers with respect to system size.

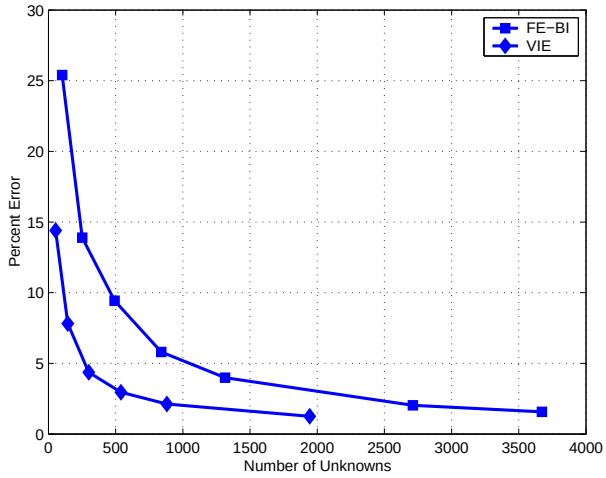


Figure 5.11. Error in FE-BI and VIE methods with respect to number of unknowns.

proaches.

So far, the MLFMM has been applied in the context of SIE formulations both for impenetrable (PEC) target geometries [13] and FE–BI formulations of inhomogeneous penetrable targets [43]. Here, we apply the MLFMM to the VIE formulations outlined above. To point out the potential of MLFMM as applied to VIEs, let us consider a sphere of unit radius. In proceeding with the numerical SIE solution, a meshing package, such as the `MSC-PATRAN`® finite element mesher, is first used to produce a surface mesh consisting of, e.g., quadrilateral elements or triangular elements. The size of the surface elements is determined by the wavelength of interest (the rule

of thumb is 10 elements per linear wavelength) and the resulting numerical system dimension is proportional to the number of elements in the mesh. As a function of element size Δ, the data in Table 5.2 were produced from the MSC-PATRAN®'s mesher. When the same gridding package is used to generate a volumetric mesh, the number

Δ	Quadrilaterals	Triangles
0.250	150	300
0.200	216	432
0.150	384	768
0.100	864	1728
0.075	1536	3072
0.050	3750	7500
0.040	5766	11532

Table 5.2. Surface mesh for a sphere.

of resulting finite elements are given in Table 5.3 as a function of the specified maximum edge length. Two observations can be made between Tables 5.2 and 5.3. First, the number of tetrahedra needed for the same discretization rate is much larger (five times larger) and thus tetrahedral meshing leads to many more unknowns. Second, in comparing Tables 5.2 and 5.3, the number of unknowns grow dramatically for volume formulations. Specifically, for volume formulations, the number of unknowns is proportional to $1/\Delta^3$ as compared to $1/\Delta^2$ for surface formulations. For high contrast dielectrics, $\Delta \approx \lambda/10\sqrt{\epsilon_r}$ must necessarily be smaller and this further exacerbates the situation. Therefore, for VIE formulations, the problem size quickly grows to many

Δ	Hexahedra	Tetrahedra
0.250	125	625
0.200	216	1080
0.150	512	2560
0.100	1728	8640
0.075	4096	20480
0.050	15625	78125
0.040	29791	148955

Table 5.3. Volumetric mesh for a sphere.

thousands, even for a physically small geometry. To overcome the high computational demands of VIEs, one must resort to more efficient strategies, like the MLFMM for solving the MoM matrix.

Core equations of the MLFMM algorithm were presented in Chapter 3. Summarizing, far-zone elements in (5.18), upon discretizing the spectral integral using numerical quadrature (see [8]) are

$$Z_{ji} = -\frac{ik}{4\pi} \sum_{\hat{k}} w_{\hat{k}} \mathbf{V}_{jm}(\hat{k})\, T_L(kr_{mm'}, \hat{k} \cdot \hat{r}_{mm'})\, \mathbf{V}^*_{im'}(\hat{k}) \tag{5.22}$$

where

$$\mathbf{V}_{jm}(\hat{k}) = \int_v dv' e^{i\mathbf{k}\cdot\mathbf{r}_{jm}} \left[\mathbf{I} - \hat{k}\hat{k}\right] \mathbf{e}_j(\mathbf{r}_{jm}) \tag{5.23}$$

and

$$\mathbf{V}_{im'}(\hat{k}) = \int_v dv' e^{i\mathbf{k}\cdot\mathbf{r}_{im'}} \left[\mathbf{I} - \hat{k}\hat{k}\right] \mathbf{e}_i(\mathbf{r}_{im'}) \tag{5.24}$$

are the signature functions. As seen, the only difference between the SIE-MLFMM signature functions and the VIE-MLFMM signature functions is the domains of the integration. By merely incorporating these signatures in the MLFMM algorithm as outlined for SIEs, we can adapt the MLFMM to VIEs.

Once the signature functions for each basis function are computed and stored, the rest of the MLFMM algorithm is identical to that of a surface formulation discussed in Chapter 3. However, the MLFMM parameters must be re-adjusted for optimal performance since the unknowns are distributed in a volume, rather than on a surface. Consequently, the cost of the MLFMM part in the matrix-vector product for the near-field section of the matrix will differ from that of the SIE-MLFMM solver. A fine tuning of the implementation is also necessary if different clustering approaches must be employed, since the MLFMM performance relies heavily on the clustering setup. For example, the MLFMM parameters (i.e. the number of multipoles and the bandwidth of the near–field matrix) are predetermined using the size and neighborhood information of the clusters. Furthermore, clustering information must be utilized for

the parallel implementation of the MLFMM solver to balance the computational load across the nodes. For example, for the oct–tree subdivision algorithm outlined above (for surface formulations), assuming a locally flat geometry, leads to $3 \times 3 = 9$ near clusters for each level, whereas the number of far clusters is $6 \times 6 - 3 \times 3 = 27$ (see Fig. 3.13). For volumetric structures, for each level, the number of near clusters is $3 \times 3 \times 3 = 27$ and the number of far clusters is $6 \times 6 \times 6 - 3 \times 3 \times 3 = 189$. Due to a volumetric distribution of unknowns, a larger number of neighboring clusters is associated with a given cluster. Hence, the computational burden of VIEs may be as much as 3 times higher for the near-field matrix-vector product and 8 times higher for the far-field portion compared to SIE-MLFMM implementations.

A CGS solver was used to carry out the numerical solutions given in this section. We note that since the VIE in (5.5) is a second kind integral equation, good convergence is expected and indeed observed. In all cases, for the number of multipoles used in the MLFMM implementation, the semi–empirical formula $L = kD_{max} + 2\log(kD_{max} + \pi)$ was adopted [8].

The first dielectric geometry considered is a spherical dielectric shell of radius 2λ and a thickness 0.2λ having $\epsilon_r = 2.75 + 0.3i$. Fig. 5.12 shows the computed bistatic RCS as compared to the analytic data obtained from the Mie series. The excellent agreement (within less than 2% rms error) between the computed and analytic results over a 40 dB dynamic range can be attributed to the higher-order geometry modeling using curved elements and conformal basis functions. The geometry was modeled using $57,624$ curvilinear hexahedra and resulted in $172,872$ unknowns ($\Delta \approx 0.07$). The MLFMM used 6 levels to converge to a residual of 10^{-2} within only 32 iterations. The number of multipoles used at each level were $\{4, 5, 7, 10, 16, 28\}$.

As a second example, we considered the scattering by a dielectric sphere coated with a dielectric shell. The sphere is of radius 0.9λ and has a dielectric constant of $\epsilon_r = 1.75 + 0.3i$, whereas the encapsulating shell is of thickness 0.1λ and has a

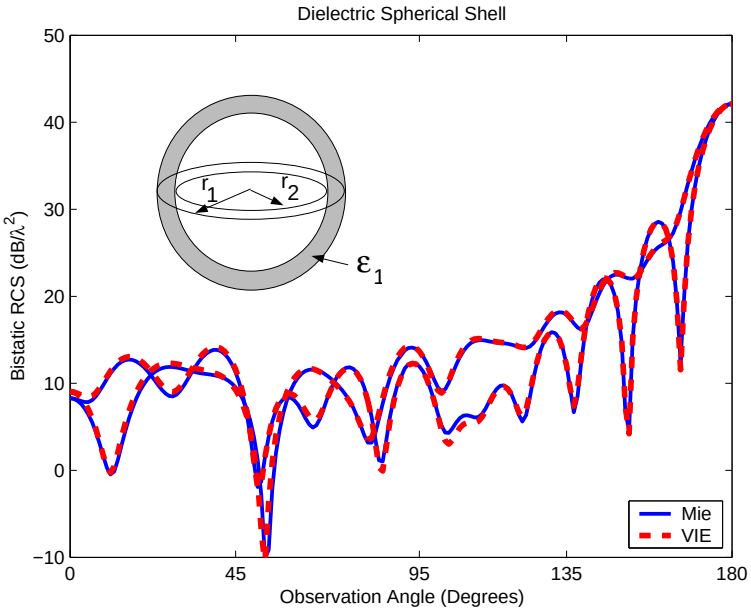


Figure 5.12. Bistatic RCS of a 2λ radius dielectric spherical shell ($\epsilon_r = 2.75 + 0.3i$).

dielectric constant of $\epsilon_r = 1.25 + 1.25i$. This geometry was modeled using $23,125$ elements ($69,375$ unknowns), and the simulation was done using 5 MLFMM levels. The CGS solver converged in 231 iterations and the number of multipoles used for this example were $\{4, 5, 7, 10, 16\}$. Again, we observe the remarkably fast convergence rate, and accuracy (within less than 2.5% rms error) using nominal sampling (see Fig. 5.13).

Our final example demonstrates the accuracy of the method for high contrast dielectric materials. The geometry is a two-layer material coated sphere with the permittivity of the sphere's core being $\epsilon_{r1} = 1.25$ up to a radius of $0.18\lambda_0$, λ_0 being the free-space wavelength. For the first layer coating, the thickness is $0.02\lambda_0$ and $\epsilon_{r2} = 15.0$ representing a contrast of 12 to 1 at the boundary between the layer and the sphere's core. The outermost layer also has a thickness of $0.02\lambda_0$ and a permittivity ϵ_{r3}. Considering the wavelength in the high permittivity shell ($\epsilon_{r2} = 15.0$), the volumetric mesh for this region is constructed by specifying the maximum edge length to be around $(\lambda_0/10)/\sqrt{\epsilon_{r2}} \approx 0.0258\lambda_0$. The rest of the volume mesh is necessarily

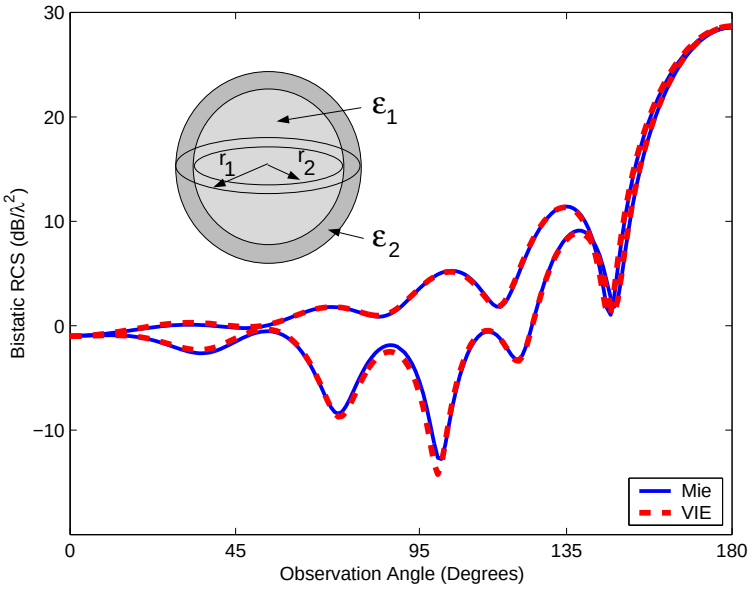


Figure 5.13. Bistatic RCS of a dielectric coated dielectric sphere ($\epsilon_1 = 1.75{+}0.3i$, $\epsilon_2 = 1.25{+}1.25i$).

constructed to be conformal with the mesh in this region. Consequently, even though the electrical size of the geometry is only $0.4\lambda_0$, the resulting matrix system is on the order of tens of thousands of unknowns. By comparison, a PEC sphere of the same electrical size would require only a few hundred unknowns. This example clearly demonstrates the need of using fast integral methods for volumetric scatterers. We considered two separate cases for the outermost layer. In one case, this layer was assumed to be a dummy air layer ($\epsilon_{r3} = 1.0$). The bistatic RCS for $\epsilon_{r3} = 1.0$ is given in Fig. 5.14 and is seen to be in very good agreement with the analytical Mie series result. For the other case, the outermost layer was assumed to have $\epsilon_{r3} = 2.2$ and the bistatic RCS result is again given in Fig. 5.14. It is interesting to observe that a change of ϵ_{r3} from 1.0 to 2.2 ($0.02\lambda_0$ thick) leads to as much as 4 dB difference between the curves at their low values.

Based on the above examples, we computed the CPU and memory requirements of the MLFMM for VIEs as functions of unknowns. Fig. 5.15 shows the CPU and memory curves of the proposed VIE-MLFMM formulation. The actual CPU times refer to a 1 GHz Pentium III personal computer. It is not surprising that as compared

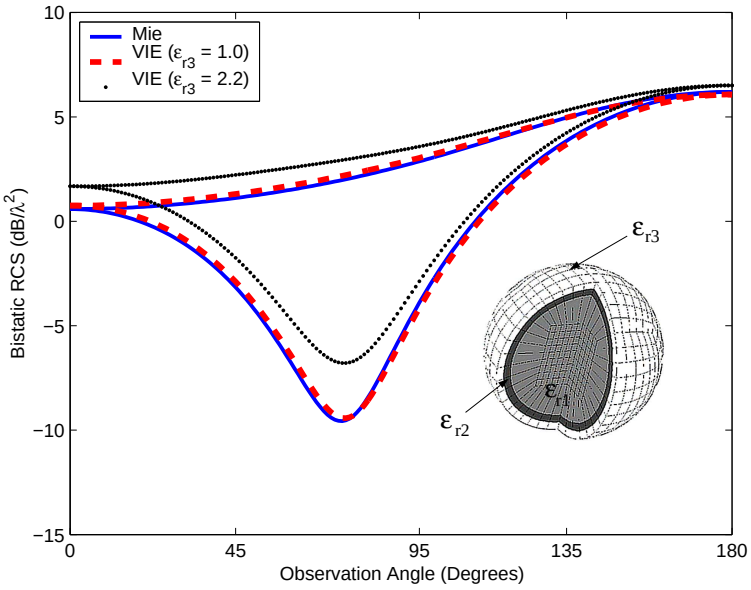


Figure 5.14. Bistatic RCS of a 2-layer dielectric coated sphere.

to an FE–BI solver, the VIE solver converges much faster, and is also free of internal resonance issues. However, since the generation of the matrix entries Z_{ji} involves double volume integrals, more time is required to fill the system as compared to the FE–BI approach. Also, the system matrix for the VIE formulation is completely dense, whereas the FE–BI formulation leads to partly sparse and partly dense matrix.

A final important point to mention is that the MLFMM becomes more efficient as the number of volume unknowns increases. Hence, the VIE is also attractive for high contrast dielectrics where higher sampling and more unknowns are required.

5.5 MoM formulation of VIE for Magnetically Permeable Structures

Having derived the volume integral representation (5.10), we next proceed with its numerical implementation and validation for composite structures where both dielectric and magnetic material parameters are involved. Specifically, in this section we describe the MoM to cast (5.10) in the form $[Z_{ji}]\{x_i\} = \{b_j\}$.

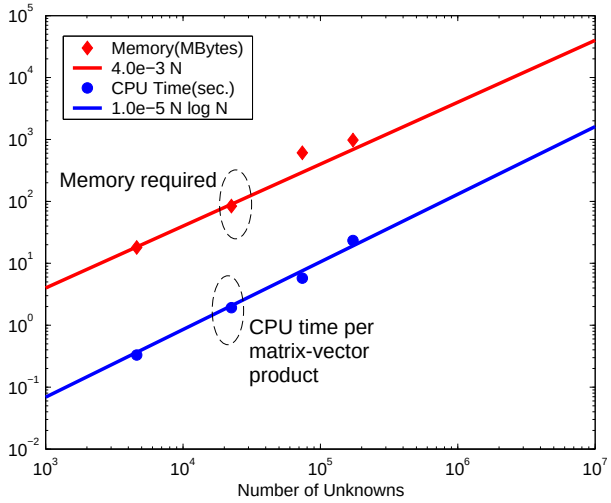


Figure 5.15. CPU and memory performance of volumetric MLFMM.

To discretize (5.10), we employ the same parametric hexahedral modeling and second-order electric field basis functions as given in (5.21).

The discretized electric field within the inhomogeneous domain v is constructed using the connectivity of the hexahedral mesh in the same manner as it is done in a finite element ordering scheme, assigning an unknown to each edge in the mesh. The hyper-singularity in $\overline{\mathbf{G}}(\mathbf{r},\mathbf{r}')$ is relaxed through the use of the divergence theorem as discussed above. This results in divergence operations on the basis and testing functions along with surface integrals over the faces of the hexahedral element (see appendix of [63]). Hence, evaluation of $\nabla\cdot\mathbf{e}$ and $\nabla\times\mathbf{e}$ are necessary to compute the matrix entries Z_{ji}, $\mathbf{e}$ denoting the basis or testing functions. We note here that electric field basis functions are expressed in the contravariant projection form, i.e. in terms of covariant unitary vectors. In this form, it is easier to evaluate the divergence of the basis functions in parametric coordinates using

$$\nabla\cdot\mathbf{e}=\frac{1}{\sqrt{G}}\left\{\frac{\partial}{\partial u}\left(\left[\mathbf{e}\cdot\mathbf{a}^{u}\right]\sqrt{G}\right)+\frac{\partial}{\partial v}\left(\left[\mathbf{e}\cdot\mathbf{a}^{v}\right]\sqrt{G}\right)+\frac{\partial}{\partial w}\left(\left[\mathbf{e}\cdot\mathbf{a}^{w}\right]\sqrt{G}\right)\right\}. \qquad (5.25)$$

It is straightforward to show that the divergences of the basis functions defined in (5.21) are identically zero. However, the evaluation of $\nabla\times\mathbf{e}$ is rather cumbersome for a contravariant projection representation. The curl operation in parametric coordinates

requires the covariant components of the basis functions, i.e.

$$\begin{aligned}\nabla \times \mathbf{e} &= \frac{1}{\sqrt{G}}\left\{\left[\frac{\partial\left(\mathbf{e}\cdot\mathbf{a}_w\right)}{\partial v}-\frac{\partial\left(\mathbf{e}\cdot\mathbf{a}_v\right)}{\partial w}\right]\mathbf{a}_u\right. \\ &+ \left.\left[\frac{\partial\left(\mathbf{e}\cdot\mathbf{a}_u\right)}{\partial w}-\frac{\partial\left(\mathbf{e}\cdot\mathbf{a}_w\right)}{\partial u}\right]\mathbf{a}_v+\left[\frac{\partial\left(\mathbf{e}\cdot\mathbf{a}_v\right)}{\partial u}-\frac{\partial\left(\mathbf{e}\cdot\mathbf{a}_u\right)}{\partial v}\right]\mathbf{a}_w\right\}\end{aligned} \tag{5.26}$$

in which the covariant components must be evaluated using the metric of the parametric transformation that forms the curvilinear element as

$$\left\{\begin{array}{c}\mathbf{e}\cdot\mathbf{a}_u\\ \mathbf{e}\cdot\mathbf{a}_v\\ \mathbf{e}\cdot\mathbf{a}_w\end{array}\right\}=\left[\begin{array}{ccc}(\mathbf{a}_u\cdot\mathbf{a}_u) & (\mathbf{a}_u\cdot\mathbf{a}_v) & (\mathbf{a}_u\cdot\mathbf{a}_w)\\ (\mathbf{a}_v\cdot\mathbf{a}_u) & (\mathbf{a}_v\cdot\mathbf{a}_v) & (\mathbf{a}_v\cdot\mathbf{a}_w)\\ (\mathbf{a}_w\cdot\mathbf{a}_u) & (\mathbf{a}_w\cdot\mathbf{a}_v) & (\mathbf{a}_w\cdot\mathbf{a}_w)\end{array}\right]\left\{\begin{array}{c}\mathbf{e}\cdot\mathbf{a}^u\\ \mathbf{e}\cdot\mathbf{a}^v\\ \mathbf{e}\cdot\mathbf{a}^w\end{array}\right\}. \tag{5.27}$$

With this substitution, evaluation of (5.26) is rather cumbersome, requiring parametric derivatives of $\sqrt{G}$. As an example, the computation of $\frac{\partial(\mathbf{e}\cdot\mathbf{a}_w)}{\partial v}$ for the specific case when $\mathbf{e}=\frac{1}{\sqrt{G}}(1-v)(1-w)\mathbf{a}_u$ gives

$$\begin{aligned}\frac{\partial\left(\mathbf{e}\cdot\mathbf{a}_w\right)}{\partial v} &= \frac{\partial}{\partial v}\left[(\mathbf{a}_w\cdot\mathbf{a}_u)(\mathbf{e}\cdot\mathbf{a}^u)+(\mathbf{a}_w\cdot\mathbf{a}_v)(\mathbf{e}\cdot\mathbf{a}^v)+(\mathbf{a}_w\cdot\mathbf{a}_w)(\mathbf{e}\cdot\mathbf{a}^w)\right] \\ &= \frac{\partial}{\partial v}\left[\left(\frac{\partial\mathbf{r}}{\partial w}\cdot\frac{\partial\mathbf{r}}{\partial u}\right)\left(\frac{1}{\sqrt{G}}(1-v)(1-w)\right)\right] \\ &= \left[\frac{\partial}{\partial v}\left(\frac{\partial\mathbf{r}}{\partial w}\cdot\frac{\partial\mathbf{r}}{\partial u}\right)\right]\left(\frac{1}{\sqrt{G}}(1-v)(1-w)\right) \\ &+ \left(\frac{\partial\mathbf{r}}{\partial w}\cdot\frac{\partial\mathbf{r}}{\partial u}\right)\left[\frac{\partial}{\partial v}\frac{1}{\sqrt{G}}\right](1-v)(1-w)-\frac{1}{\sqrt{G}}\left(\frac{\partial\mathbf{r}}{\partial w}\cdot\frac{\partial\mathbf{r}}{\partial u}\right)\end{aligned} \tag{5.28}$$

Nevertheless, once implemented, the same singularity annihilation method as in [63] (also see Appendix B) can be used to calculate the resulting matrix elements for the self-cell terms given by

$$\begin{aligned}Z_{ji} &= \langle\mathbf{e}_j(\mathbf{r}),\mathbf{e}_i(\mathbf{r})\rangle-k_0^2\langle\mathbf{e}_j(\mathbf{r}),\int_v d\mathbf{r}'\overline{\mathbf{G}}(\mathbf{r},\mathbf{r}')\cdot\left(\overline{\epsilon}_r-\overline{\mathbf{I}}\right)\cdot\mathbf{e}_i(\mathbf{r}')\rangle \\ &+ \langle\mathbf{e}_j(\mathbf{r}),\int_v d\mathbf{r}'\nabla'\times\overline{\mathbf{G}}(\mathbf{r},\mathbf{r}')\cdot\left(\overline{\mu}_r^{-1}-\overline{\mathbf{I}}\right)\cdot\nabla'\times\mathbf{e}_i(\mathbf{r}')\rangle.\end{aligned} \tag{5.29}$$

The right hand side of the matrix system in the parametric space takes the form $b_j=\langle\mathbf{e}_j(\mathbf{r}),\mathbf{E}^{inc}(\mathbf{r})\rangle$.

Having presented the implementation of the integral equation (5.10) for the volume fields **E**, we next proceed to validate this implementation. First, we consider calculations for purely magnetic materials. Consider the scattering by a permeable cube (plane wave incidence is normal to the cube's face). The specific cube has a side length of $0.5\lambda_0$ and $\mu_r = 2.2$. As seen, the VIE and FE-BI data are in good agreement and demonstrate the validity of (5.10). Also shown in Fig. 5.16 (b) is the radar scattering by a spherical shell having an outer radius of $0.2\lambda_0$, an inner radius of $0.18\lambda_0$, and a relative permeability of $\mu_r = 2.2$. Again, the agreement between the FE-BI and VIE data is excellent.

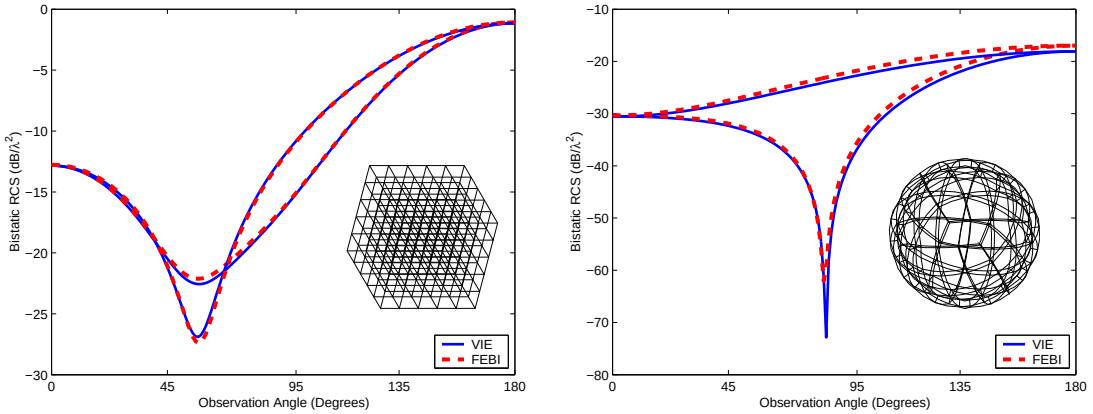


Figure 5.16. Bistatic RCS of two magnetically permeable scatterers, (a) A cube of side length 0.5λ and $\mu_r = 2.2$, (b) A spherical shell of $\mu_r = 2.2$, 0.2λ outer radius, and 0.18λ inner radius.

For the validation of the VIE for composite structures, we present scattering computations from three different configurations. The solution of (5.10) will be compared with corresponding results based on the FE-BI method. Our first example is a composite cube of side length $0.2\lambda_0$. The incident plane wave is propagating in the negative z direction and the bistatic RCS of the cube is plotted in Fig. 5.17. For this special case, the covariant and contravariant components of the basis functions are just scaled versions of each other, $\sqrt{G}$ is constant, and hence the curl operation is fairly straightforward to calculate. As seen, the agreement between the VIE and

FE-BI solution is very good for both polarizations.

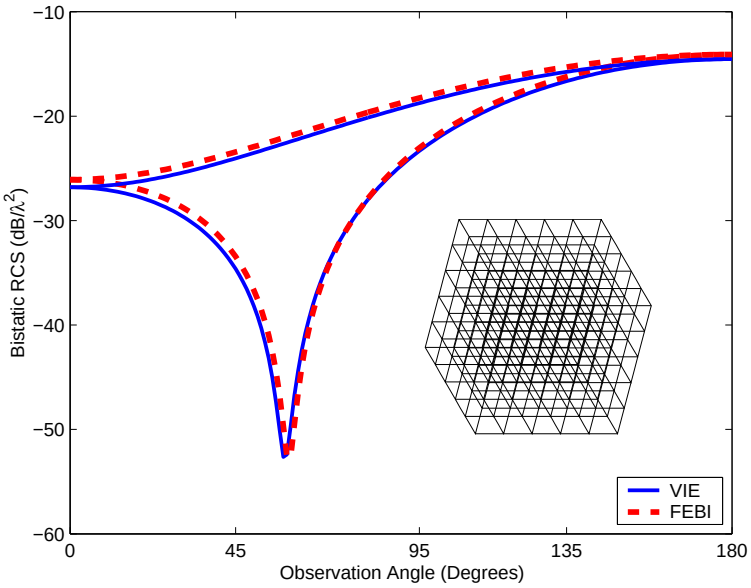


Figure 5.17. Bistatic radar cross section (RCS) of a homogeneous composite cube of side length $a = 0.2\lambda$, and $\epsilon_r = 1.5$, $\mu_r = 2.2$.

As a second example, we consider a thin spherical shell of outer radius of $0.2\lambda_0$, thickness $0.02\lambda_0$ and of the same material parameters ($\epsilon_r = 1.5, \mu_r = 2.2$) as in the previous example. For modelling, the mesh is constructed entirely as a thin layer of distorted (non-rectangular) curvilinear elements, and both VIE and FE-BI solution used the same mesh. The bistatic RCS is shown in Fig. 5.18 and it is clear that the solutions are again in full agreement.

Finally, as our third example, we evaluate the RCS of a solid sphere of radius $0.15\lambda_0$. The permittivity and permeability of the sphere are again $\epsilon_r = 1.5$ and $\mu_r = 2.2$, respectively. We again used the same volumetric mesh as shown in Fig. 5.19 for both methods. Again, excellent agreement is seen between the VIE and FE-BI solutions.

We note here that, since the same volumetric mesh is used for both the VIE and FE-BI solutions, and since both the FE-BI volume basis functions and the VIE basis are functions associated with the edges of the mesh, the resulting number of volumet-

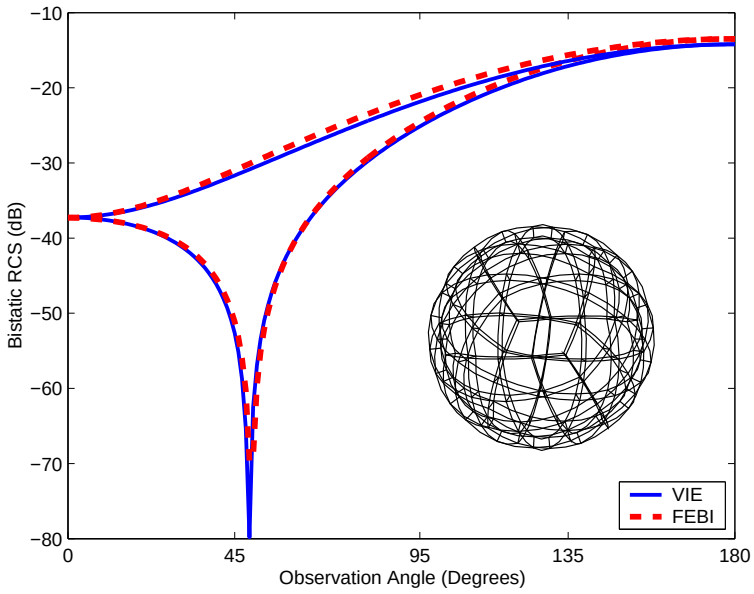


Figure 5.18. Bistatic RCS of a composite spherical shell. The outer shell radius is $r_o = 0.2\lambda$, its thickness is $d = 0.02\lambda$, and its relative constitutive parameters are $(\epsilon_r = 1.5, \mu_r = 2.2)$.

ric unknowns in both methods are the same. However, the FE-BI method requires an additional set of surface unknowns since both **E** and **H** are solved on the surface s bounding of the volume v. Hence, the resulting FE-BI system is of higher rank than the VIE system. Nevertheless, the FE-BI system is partly full and mostly sparse. In contrast, the VIE system is fully populated and thus requires substantial computation time for its generation. However, since the VIE is a second kind integral equation, its iterative solution converges quickly. As an example, for a system of 1082 (second example) VIE unknowns, the conjugate gradient squared (CGS) solver converged in 5 iterations for a relative error of 10^{-2} and the solution was completed in only 0.7 seconds using a 1 GHz PIII processor. For the third example (880 VIE unknowns), the iterative CGS solver converged in 13 iterations and took 1.1 seconds. To our knowledge, this is the first published implementation of the VIE for inhomogeneous volumes having both dielectric and magnetic properties.

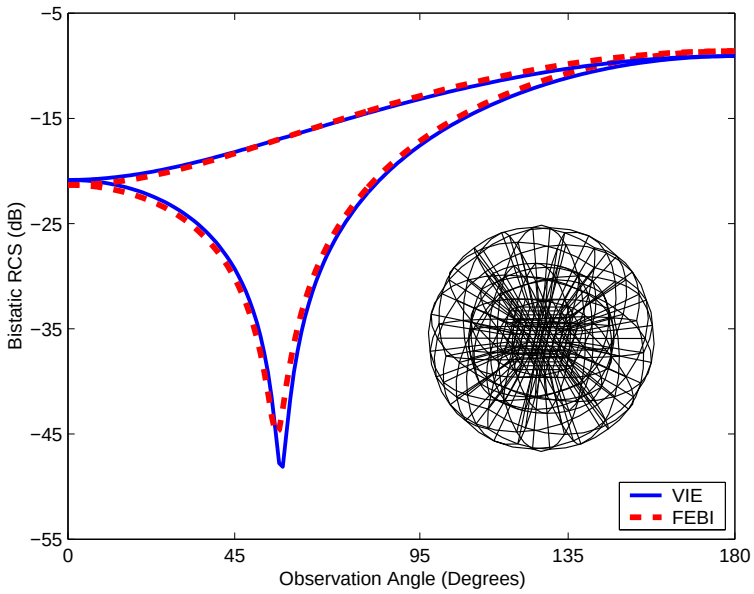


Figure 5.19. Bistatic RCS (incidence at zero degrees) for a homogeneous composite sphere of radius $r = 0.15\lambda$. The sphere has the relative constitutive parameters ($\epsilon_r = 1.5, \mu_r = 2.2$).

CHAPTER 6

Conclusions

Owing to their lower numerical error and stability, integral equation solution methods have increasingly become popular over the last five years. Although more costly than partial differential equation methods, their superior error performance has provided accurate solutions to electromagnetics problems. Key to the increased utilization of integral methods has been the introduction of fast solution procedures such as the MLFMM.

The fidelity of the mathematical model is also a critical factor in attaining highly accurate numerical solutions that reflect the actual problem. The contribution of this thesis has primarily been on the introduction of curvilinear elements for high fidelity geometry modeling in conjunction with the MLFMM. Also, for the first time, integral methods were developed and applied to penetrable volumetric structures of arbitrary composition.

Most methods developed till now dealt with low-order geometry modeling, using flat-facets or low-order tetrahedra. This approximation introduces numerical error in the solution process. Also, fine details and large surface curvatures result in very fine tessellations increasing the computational cost of the solution process. Better geometry modeling schemes are thus necessary to minimize geometry modeling error.

In the thesis we presented the application of the MLFMM to efficiently solve surface and volume integral equation using curvilinear geometry modeling. Particular contributions of the thesis are

- MLFMM solution for PEC scattering problems using curvilinear biquadratic surface modeling.
- MLFMM solutions of hybrid FE-BI systems using curvilinear hexahedral volume elements.
- Development and numerical solutions of VIEs for modeling inhomogeneous geometries involving both dielectric and magnetic material parameters.

Also, accuracy and computational savings of the MLFMM were presented and contrasted to other solution methods.

To summarize, Chapter 2 outlined a framework for conformal geometry representation using curvilinear elements for surface modeling. This framework provided a generalization of flat-facet modeling using low-order elements and forms a basis to extend various formulations and definitions to higher-order geometry modeling and higher-order field expansions. We also showed that the size of the resulting system can be reduced by using curvilinear elements. Hence larger problems can be solved on the same computing platform.

In addition to using better geometry models to reduce the solution error, we presented the MLFMM in Chapter 3 for the rapid solution of SIEs and hybrid FE-BI systems as well as VIEs. This removed the $O(N^2)$ computational bottleneck of the conventional MoM. The accuracy and efficiency of the methods were demonstrated and ILU preconditioning was employed for better convergence. This is especially important for the evaluation of multiple incidence angles (e.g. monostatic RCS predictions) where the same system must be solved many times.

In Chapter 4, we outlined the FE-BI procedure described in [19] and appended it with the MLFMM. We also gave an extensive presentation of the curvilinear geometry descriptions for volumetric elements.

As an alternative to the FE-BI method for inhomogeneous targets, a VIE formulation was presented in Chapter 5. Derivation of a new integral equation was given

and its implementation using curvilinear volumetric elements and conformal basis functions of various orders was presented. The numerical solution of such a VIE for magnetically permeable materials using a reduced number of unknowns and conformal geometry modeling is a major contribution of this work. Second-order basis functions defined in Chapter 5 enabled the solution of the proposed VIE in terms of only the electric field unknown. Furthermore, the MLFMM was applied to the solution of VIEs for possibly lossy dielectric targets using zeroth-order (parametrically constant) basis functions that allow for a physical domain decomposition. The accuracy and the numerical efficiency of VIE-MLFMM were demonstrated by comparing the generated results with reference solutions. Implementation issues for MLFMM on distributed memory computing platforms was presented in Appendix D. Another appendix presented the important details of the evaluation of the singular kernels in the VIE. Specifically, an annihilation method that involved a series of parametric transformations to remove the singularity was presented. Performance and advantages of the annihilation method over the conventional singularity extraction method were demonstrated for SIE implementations.

Much progress has indeed been achieved in this work for modeling high contrast penetrable materials. However, greater efficiencies can be achieved by considering higher-order basis functions with conformal geometrical features. This remains a future extension of the work. Combination of different elements is also advantageous in modeling geometry details.

For a multi-spectral analysis, the methods presented in this thesis must be employed repeatedly for each frequency. Depending on the electrical size of the problem, each solution may require considerable amount of CPU time and memory. However, significant speed-ups can be achieved by using the previous frequency solution information. Similarly, for a single frequency, the matrix system needs to be solved repeatedly for different excitations. In this case, block solvers may be utilized to

concurrently solve for multiple right hand sides.

Although the complexity of the MLFMM is low, the implementation will have an upper limit on the size of the largest problem to be analyzed. To solve even larger problems, one has to resort to using high-frequency approximations such as the PO or the geometrical theory of diffraction. Hybridization of MLFMM with high-frequency methods is an area that deserves serious investigation and a significant investment.

Although the thesis did not focus on applications, the presented methods have a wide range of applicability. RCS and antenna analysis are certainly the most direct applications. However, the developed algorithms have great applicability to bio-medical imaging, indoor propagation in wireless systems, EMI/EMC for aircraft and radio frequency (RF) circuits. Moreover, the speed-up offered opens for the first time possibilities for designing RF devices and possibly complete RF systems.

APPENDICES

APPENDIX A

Lagrange Interpolation Coefficients

As depicted in Fig. A.1, for a one dimensional quadratic interpolation problem, the three Lagrange coefficients can be constructed as

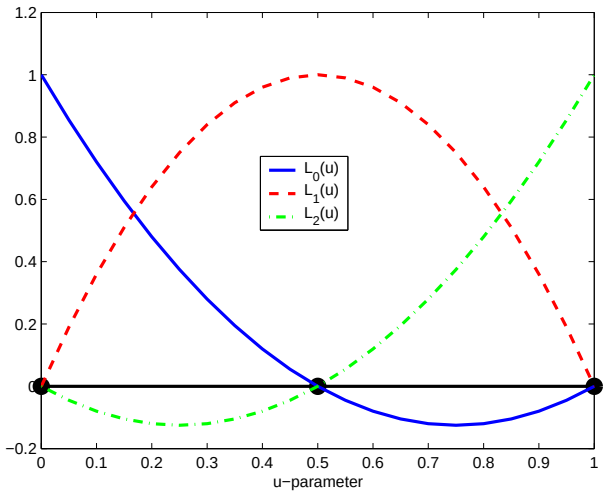


Figure A.1. Quadratic Lagrange interpolators.

$$L_0(u) = 2(u - 0.5)(u - 1), \quad L_1(u) = -4(u)(u - 1), \quad L_2(u) = 2(u)(u - .5). \quad \text{(A.1)}$$

With these definitions, a quadratic curve in three-dimensions passing through a given set of points $\mathbf{r}_i, (i = 0, 1, 2)$ is given by

$$\mathbf{r}(u) = \sum_{i=0}^{2} L_i(u)\mathbf{r}_i. \quad \text{(A.2)}$$

Using (A.1), it is straightforward to construct the two and three dimensional Lagrange interpolation coefficients as Cartesian products. For 2-dimensions they take the form

$$L_{ij}(u, v) = L_i(u)L_j(v) \quad \text{for} \quad \{i, j\} = \{0, 1, 2\}, \quad \text{(A.3)}$$

and a biquadratic surface patch defined by 9 points in space ($\mathbf{r}_{ij}, \{i,j\} = \{0,1,2\}$) on a topologically rectangular grid is given by

$$\mathbf{r}(u,v) = \sum_{i=0}^{2}\sum_{j=0}^{2} L_{ij}(u,v)\mathbf{r}_{ij}. \tag{A.4}$$

Similarly, for 3-dimensions the Lagrange interpolators are constructed as

$$L_{ijk}(u,v,w) = L_i(u)L_j(v)L_k(w) \quad \text{for} \quad \{i,j,k\} = \{0,1,2\}, \tag{A.5}$$

and for the tri-quadratic hexahedral element

$$\mathbf{r}(u,v,w) = \sum_{i=0}^{2}\sum_{j=0}^{2}\sum_{k=0}^{2} L_{ijk}(u,v,w)\mathbf{r}_{ijk} \tag{A.6}$$

where $\mathbf{r}_{ijk}, \{i,j,k\} = \{0,1,2\}$ are the 27 defining points.

APPENDIX B

Singularity Annihilation Methods for Surfaces and Volumes

The matrix elements in (5.29) can be rewritten as

$$Z_{ji} = \langle \mathbf{e}_j(\mathbf{r}), \mathbf{e}_i(\mathbf{r}) \rangle - \langle \mathbf{e}_j(\mathbf{r}), \mathbf{I}_1(\mathbf{r}) \rangle - \frac{1}{k_0^2} \langle \mathbf{e}_j(\mathbf{r}), \mathbf{I}_2(\mathbf{r}) \rangle, \tag{B.1}$$

where

$$\mathbf{I}_1(\mathbf{r}) = \int_v dv' g(\mathbf{r}, \mathbf{r}')(k^2(\mathbf{r}') - k_0^2)\mathbf{e}_i(\mathbf{r}') \tag{B.2}$$

and

$$\mathbf{I}_2(\mathbf{r}) = \nabla \int_v dv' \nabla g(\mathbf{r}, \mathbf{r}') \cdot (k^2(\mathbf{r}') - k_0^2)\mathbf{e}_i(\mathbf{r}'). \tag{B.3}$$

To deal with the $1/\left|\mathbf{r} - \mathbf{r}'\right|^3$ singularity in $\mathbf{I}_2(\mathbf{r})$ we proceed as usual by transferring the ∇ operator onto the testing function. We have,

$$\begin{aligned}
\int_v dv \mathbf{e}_j(\mathbf{r}) \cdot \mathbf{I}_2(\mathbf{r}) &= \int_v dv \mathbf{e}_j(\mathbf{r}) \cdot \nabla \int_v dv' \nabla g(\mathbf{r}, \mathbf{r}') \cdot (k^2(\mathbf{r}') - k_0^2)\mathbf{e}_i(\mathbf{r}') \\
&= \int_v dv \nabla \cdot \left[\mathbf{e}_j(\mathbf{r}) \int_v dv' \nabla g(\mathbf{r}, \mathbf{r}') \cdot (k^2(\mathbf{r}') - k_0^2)\mathbf{e}_i(\mathbf{r}') \right] \\
&- \int_v dv \left[\nabla \cdot \mathbf{e}_j(\mathbf{r}) \right] \int_v dv' \nabla g(\mathbf{r}, \mathbf{r}') \cdot (k^2(\mathbf{r}') - k_0^2)\mathbf{e}_i(\mathbf{r}') \\
&= \oint_s d\mathbf{s} \cdot \left[\mathbf{e}_j(\mathbf{r}) \int_v dv' \nabla g(\mathbf{r}, \mathbf{r}') \cdot (k^2(\mathbf{r}') - k_0^2)\mathbf{e}_i(\mathbf{r}') \right] \\
&- \int_v dv \nabla \cdot \mathbf{e}_j(\mathbf{r}) \int_v dv' \nabla g(\mathbf{r}, \mathbf{r}') \cdot (k^2(\mathbf{r}') - k_0^2)\mathbf{e}_i(\mathbf{r}'),
\end{aligned} \tag{B.4}$$

where s denotes the surface enclosing v. The source (primed) integral in (B.4) can further be rewritten as

$$\begin{aligned}\int_v dv'\nabla g(\mathbf{r},\mathbf{r}')\cdot(k^2(\mathbf{r}')-k_0^2)\mathbf{e}_i(\mathbf{r}') &= -\int_v dv'\nabla' g(\mathbf{r},\mathbf{r}')\cdot(k^2(\mathbf{r}')-k_0^2)\mathbf{e}_i(\mathbf{r}') \\ &= -\int_v dv'\nabla'\cdot\Big[g(\mathbf{r},\mathbf{r}')\cdot(k^2(\mathbf{r}')-k_0^2)\mathbf{e}_i(\mathbf{r}')\Big] \\ &+ \int_v dv' g(\mathbf{r},\mathbf{r}')\nabla'\cdot\Big[(k^2(\mathbf{r}')-k_0^2)\mathbf{e}_i(\mathbf{r}')\Big] \\ &= -\oint_s d\mathbf{s}'\cdot g(\mathbf{r},\mathbf{r}')(k^2(\mathbf{r}')-k_0^2)\mathbf{e}_i(\mathbf{r}') \\ &+ \int_v dv' g(\mathbf{r},\mathbf{r}')\nabla'\cdot\Big[(k^2(\mathbf{r}')-k_0^2)\mathbf{e}_i(\mathbf{r}')\Big] \quad \text{(B.5)}\end{aligned}$$

The above integrals are not improper, nevertheless, their numerical integration must be carefully done due to the first order singularity of the Green's function. To evaluate the singular source integrals in (B.5), we herewith employ an annihilation technique that converts the singular integrands from the original (u,v,w) parametric space to smooth, non-singular integrands on an auxiliary (α,β,γ) parametric space. To illustrate this, let us consider the volume integral in (B.5) involving the divergence of the basis functions. First, we recast it in the (u,v,w) space as

$$I=\int_0^1\int_0^1\int_0^1 g(\mathbf{r},\mathbf{r}')\nabla'\cdot\Big[(k^2(\mathbf{r}')-k_0^2)\mathbf{e}_i(\mathbf{r}')\Big]\sqrt{G}dudvdw. \quad \text{(B.6)}$$

For further discussion, we conveniently rewrite (B.6) in a more general form as a ratio of two well behaved functions in the (u,v,w) space

$$I=\int_0^1\int_0^1\int_0^1 \frac{f(u_0,v_0,w_0,u,v,w)}{g(u_0,v_0,w_0,u,v,w)}dudvdw \quad \text{(B.7)}$$

where $g(u_0,v_0,w_0,u,v,w)=0$ when $(u,v,w)=(u_0,v_0,w_0)$ and $f(u_0,v_0,w_0,u,v,w)$ is well behaved at the same point. We note that in (B.7), the original observation point has been mapped to (u_0,v_0,w_0).

The next step in the annihilation process is to divide the unit cube into eight quadrants using the three isoparametric planes defined by $u=u_0$, $v=v_0$, and $w=w_0$ (see Fig. B.1 (a)). When these quadrants are considered separately, the

original singularity now appears at a vertex of each of the eight quadrants. For integration, we next proceed to map each of the eight quadrants to a unit cube in the (η, ζ, ξ) parametric space as shown in Fig. B.1 (b). We need to again be careful to choose our transformations so that the singularity for one of the vertices of each quadrant appears at the origin of the (η, ζ, ξ) space. For example, the transformation to map the quadrant defined by $(u_0 < u < 1), (0 < v < v_0), (w_0 < w < 1)$ is

$$\begin{aligned} u &= (1 - u_0)\eta + u_0 \\ v &= v_0(1 - \zeta) \\ w &= (1 - w_0)\xi + w_0. \end{aligned} \tag{B.8}$$

Similar transformations must be used for the other seven quadrants.

Using this eight-quadrant decompositions, the integral (B.7) can be written as

$$I = \sum_{i=1}^{8} \int_0^1 \int_0^1 \int_0^1 \frac{f_i(\eta, \zeta, \xi)}{g_i(\eta, \zeta, \xi)} J_i(\eta, \zeta, \xi) d\eta d\zeta d\xi \tag{B.9}$$

where $J_i(\eta, \zeta, \xi)$ are the Jacobians of the transformations for each quadrant and f_i and g_i are the portions of the original f and g that fall in respective quadrants. We should remark that f_i, g_i, and J_i do depend on (u_0, v_0, w_0), but this dependence has been suppressed. Also, since the transformations used to map (η, ζ, ξ) to (u, v, w) are simple scaling transformations, the Jacobians $J_i(\eta, \zeta, \xi)$ are constants. For the example given in (B.8), the Jacobian is simply $J_i = (1 - u_0)(-v_0)(1 - w_0)$.

A final step for integration is to propose another parametric transformation whose Jacobian has a zero (of order equal to or higher than that of g) at the origin of the (η, ζ, ξ) space (Fig. B.1 (c)). One such transformation is

$$\begin{aligned} \eta &= \alpha^3 \\ \zeta &= \beta^3 \\ \xi &= \gamma^3 \end{aligned} \tag{B.10}$$

and we note that a 3^{rd} order transformation is used for sufficient smoothness of the integrand in the (α, β, γ) space. Doing this transformation allows us to rewrite the integral (B.9) as

$$I = \sum_{i=1}^{8} \int_0^1 \int_0^1 \int_0^1 \frac{f_i(\alpha, \beta, \gamma)}{g_i(\alpha, \beta, \gamma)} J_i j(\alpha, \beta, \gamma) d\alpha d\beta d\gamma \tag{B.11}$$

In this, $j(\alpha, \beta, \gamma) = 27\alpha^2\beta^2\gamma^2$ is the Jacobian of the (η, ζ, ξ) to (α, β, γ) transformation. Since the Jacobian has a second order zero at the origin of the (α, β, γ) space, it serves to annihilate the singularity of $1/g_i(\alpha, \beta, \gamma)$ in each of the eight quadrants. Thus, (B.11) has a non-singular integrand and can be evaluated numerically using a sufficient order quadrature rule (e.g. Gaussian quadrature).

A similar integration procedure can be applied to evaluate the surface source integral in (B.5), considering each six faces of the source hexahedron separately.

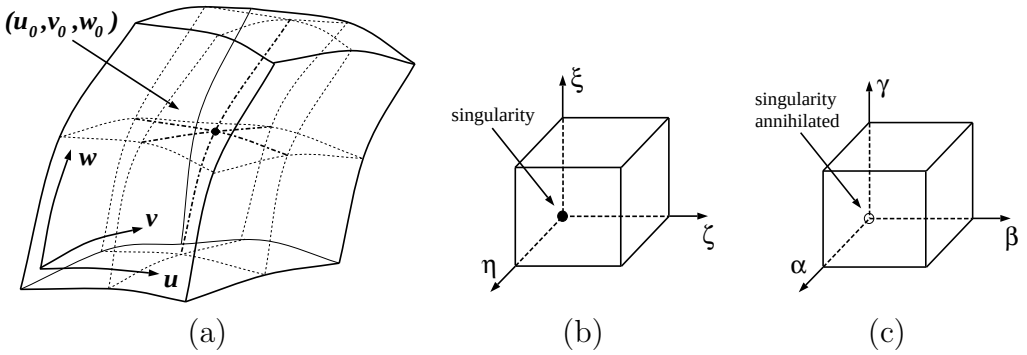


Figure B.1. (a) Hexahedron in real space, (b) Auxiliary parametric space, (c) Integration space

Below, we present the annihilation procedure for surfaces. The generic integral considered here is

$$\int_{(u,v)} du dv \frac{f(u, v)}{R(u, v, u_0, v_0)} \tag{B.12}$$

where $R(u, v, u_0, v_0)$ has a first order zero at the observation point(u_0, v_0). Referring to Fig. B.2, each of the four quadrants can be mapped into an auxiliary (s, t) unit

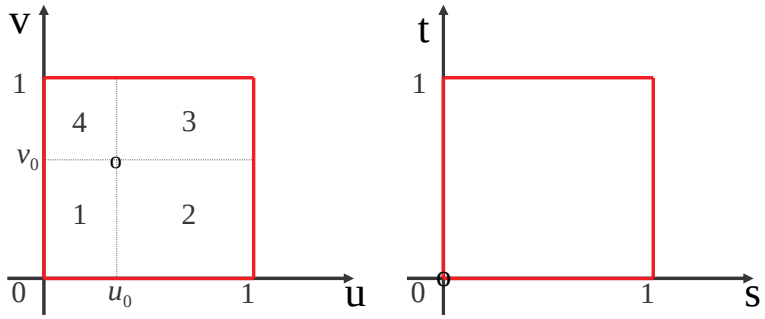


Figure B.2. Auxiliary transformation of the four quadrants in the (u, v) space

square through these four transformations:

$$
\begin{aligned}
(1) \quad &: \quad u(s) = u_0 - u_0 s, \quad v(t) = v_0 - v_0 t, \text{ with } J_{(1)} = u_0 v_0 \qquad \text{(B.13)} \\
(2) \quad &: \quad u(s) = u_0 + (1 - u_0)s, \quad v(t) = v_0 - v_0 t, \\
&\quad \text{with } J_{(2)} = -(1 - u_0)v_0 \\
(3) \quad &: \quad u(s) = u_0 + (1 - u_0)s, \quad v(t) = v_0 + (1 - v_0)t, \\
&\quad \text{with } J_{(3)} = (1 - u_0)(1 - v_0) \\
(4) \quad &: \quad u(s) = u_0 - u_0 s, \quad v(t) = v_0 + (1 - v_0)t, \\
&\quad \text{with } J_{(4)} = -u_0(1 - v_0)
\end{aligned}
$$

where $J_{(i)}$ denote the Jacobian of each transformation. So, (B.12) can be rewritten in the (s, t) space as

$$\int_{(u,v)} dudv \frac{f(u,v)}{R(u,v,u_0,v_0)} = \sum_{i=1}^{4} \int_{(s,t)} dsdt \frac{f_{(i)}(s,t)}{R_{(i)}(s,t)} J_{(i)} \qquad \text{(B.14)}$$

where the subscripts denote the corresponding portions of the functions on the four quadrants in the (u, v) space. With these transformations, each quadrant is mapped into the (s, t) space so that the singularity point (u_0, v_0) appears at the origin on the (s, t) space. If we further devise a transformation on (s, t) that has a zero of higher order than $R(u, v)$, we expect the Jacobian of this transformation to effectively annihilate the zero of the denominator. With $s(\eta) = \eta^3$ and $t(\xi) = \xi^3$, we have the

Jacobian $J(\eta,\xi) = 9\eta^2\xi^2$. Hence, the integrand in the (η,ξ) space reads as

$$\int_{(u,v)} dudv \frac{f(u,v)}{R(u,v,u_0,v_0)} = \sum_{i=1}^{4} \int_{(\eta,\xi)} d\eta d\xi \frac{f_{(i)}(\eta,\xi)}{R_{(i)}(\eta,\xi)} J_{(i)} J(\eta,\xi) \tag{B.15}$$

and the introduced $J(\eta,\xi)$ will cancel the zero of $R_{(i)}(\eta,\xi)$ at $(\eta = 0, \xi = 0)$. Hence, (B.15) can safely be evaluated using a suitable quadrature rule.

The alternative method relies on adding and subtracting a singular function to (B.12) which can analytically be integrated, i.e.

$$\begin{aligned} \int_{(u,v)} dudv \frac{f(u,v)}{R(u,v,u_0,v_0)} &= \int_{(u,v)} dudv \left[\frac{f(u,v)}{R(u,v,u_0,v_0)} - \frac{f(u_0,v_0)}{R_0(u,v,u_0,v_0)} \right] \\ &+ \int_{(u,v)} dudv \frac{f(u_0,v_0)}{R_0(u,v,u_0,v_0)} \end{aligned} \tag{B.16}$$

where $R_0(u,v,u_0,v_0)$ is constructed using the Taylor approximation to the curved surface at the observation point (u_0,v_0) as shown in Fig. B.3. This is the so called singularity extraction method. The second term in (B.16) can be analytically evalu-

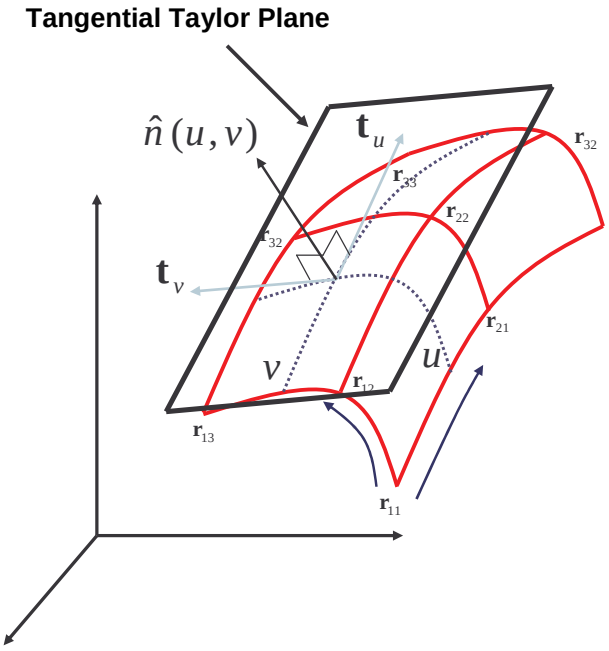


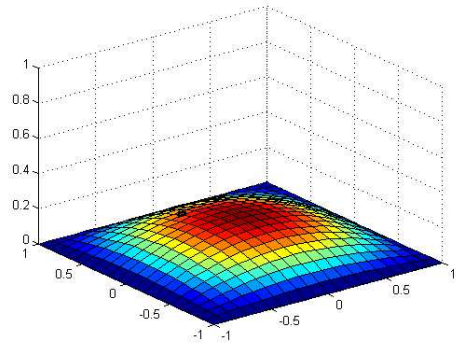
Figure B.3. Illustration of the Taylor plane at (u_0,v_0).

ated as given in [4] and the kernel of the first integral is no longer singular. However, as we will demonstrate, this integrand is highly oscillatory around the observation point.

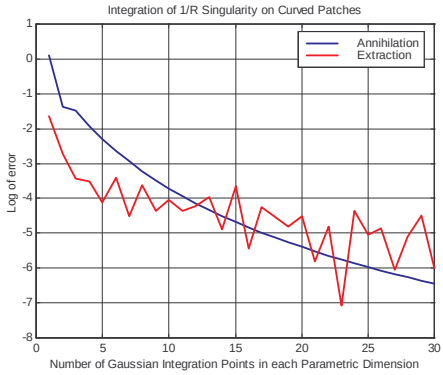
We consider three cases as shown in Fig. B.4 with three different integration domains having increasing curvatures. The numerical integral of (B.12) using Gaussian quadrature is given in Fig. B.4. The error is evaluated with respect to a 50 point Gaussian quadrature using the annihilation method. As seen in this figure, the annihilation method provides almost a monotonic convergence as compared to the extraction method. As the element curvature gets larger, the annihilation method fails to accurately integrate (B.12). The reason behind this behavior can be found when we look at the actual integrands of both methods. Shown in Fig. B.5 are the integrands that are numerically evaluated for the three cases. For the annihilation method, the integrands are plotted in the (η, ξ) space where there is no singularity. On the other hand, we can obviously observe that an attempt to numerically evaluate the kernels of the extraction method will be problematic due to high oscillations around the observation point. Even though the kernels are not singular, rapid changes around (u_0, v_0) will render a numerical integration scheme based on polynomial approximations of the kernel useless.

Case I , Slight curvature

$u_0 = 0.4, v_0 = 0.7$

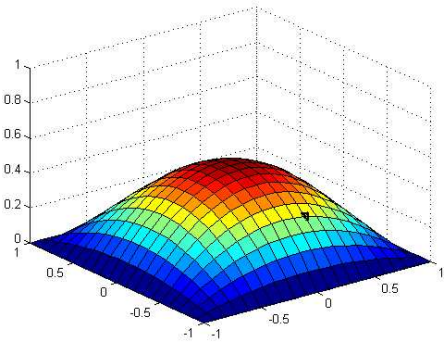


Log of error in the integral

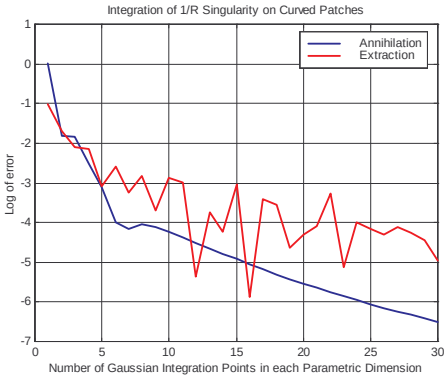


Case II , Medium curvature

$u_0 = 0.6, v_0 = 0.2$

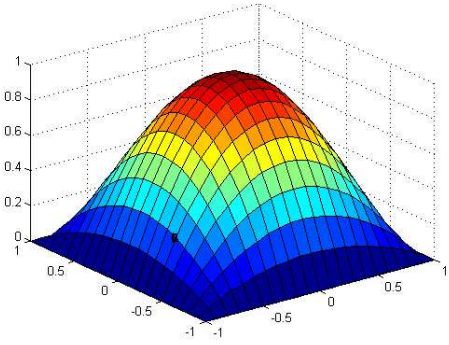


Log of error in the integral



Case III, High curvature

$u_0 = 0.1, v_0 = 0.3$



Log of error in the integral

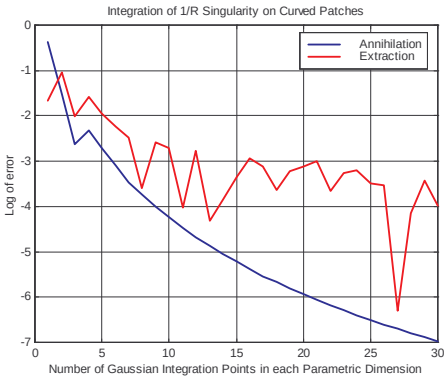


Figure B.4. Performances of annihilation and extraction methods for three different cases.

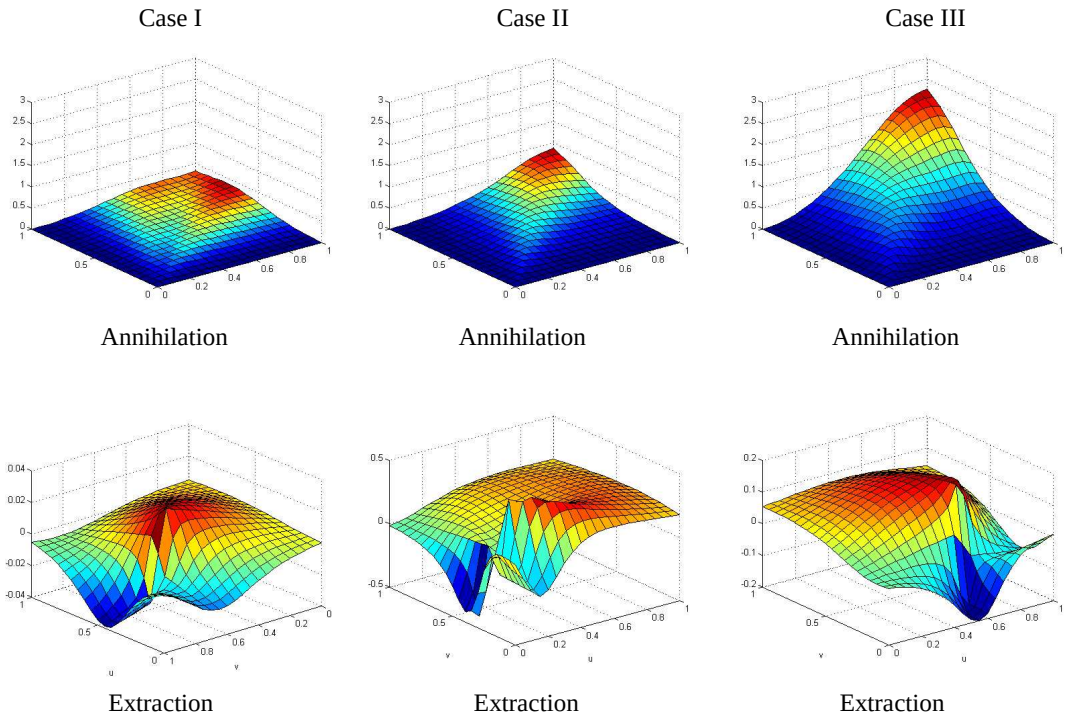


Figure B.5. Kernels of both methods for the three cases.

APPENDIX C

Incomplete LU Preconditioner for FMM implementation

For large scale simulations, possibly with geometrical surface details (e.g. antenna arrays on aircraft), the density of the surface mesh cannot be expected to be uniform. Nevertheless, a nonuniform mesh is well known to produce ill-conditioned MoM matrix equations. Also, different formulations of the same electromagnetic problem are associated with different condition numbers. For example, the CFIE formulation is known to give rise to much better conditioned systems than the EFIE or MFIE. Further, as noted above, the FMM implementation introduces erroneous minima in the solution domain. Use of a preconditioner is therefore essential for robust implementations of iterative solvers.

Although the diagonal preconditioner is simple and leads to significant convergence improvements, it does so for diagonally dominant matrices. Block diagonal preconditioners are more robust but require renumbering of the grid or matrix rearranging so that the dominant matrix terms are clustered around the diagonal. This can easily be done for 2D problems, but is quite difficult, if at all possible, in three dimensions. Alternatively, when the FMM is used to speed up the iterative solution, we have the natural choice of using the near-field portion of the MoM matrix for preconditioning. These near-field elements are the largest in magnitude and constitute the un-approximated portion of the system matrix.

One preconditioning approach is to perform a direct LU decomposition on the un-approximated part of the matrix. However, depending on the sparsity pattern of

the near-field matrix, this may require a significant amount of fill-ins. For large scale simulations, these fill-ins may become a bottleneck in memory utilization. Alternatively, the fill-in requirement of direct LU can be resolved by performing the ILU factorization. The ILU is the same as a direct LU algorithm but avoids fill-ins of elements in the decomposed LU matrices. This also results in less CPU utilization.

The ILU algorithm from [39] has been incorporated into the FMM solver. The pseudo code is repeated below for completeness.

```
for i = 2,...,n, do:
  for k = 1,...,i-1 and for (i,k) in NZ(Z) do:
    compute z_ik = z_ik/z_kk
    for j = k+1,...,n and for (i,j) in NZ(Z) do:
      compute z_ij = z_ij - z_ik z_kj
    end do
  end do
end do
```

Here `NZ(Z)` is the sparsity pattern of the near-field matrix `Z`, and the conventional LU decomposition algorithm is only applied to the non-zero entries of the matrix. Hence, memory utilization is not affected and moreover the sparsity pattern of the stored ILU matrix is identical to that of the original matrix. Thus, further memory savings are attained using ILU decomposition.

C.1 ILU Performance

To evaluate the performance of the ILU preconditioner, we considered a perfectly electrically conducting (PEC) ogive geometry (depicted in Fig. 2.10 (f)). For this study, the ILU preconditioner was implemented in the matrix systems based on the

EFIE, MFIE, and CFIE formulations. As described above, the FMM near-field matrix is used as a preconditioner in the context of several iterative solver [39]. The size of the matrix system was 480 and refers to a $10'' \times 2'' \times 2''$ ogive with its long axis coincident with the x-axis. All calculations were carried out at 5.91 GHz. This is indeed a very small system and serves the purpose of validating the preconditioning scheme. Also, the ogive was chosen due to its irregular grid at the tips. A uniformly meshed sphere does not serve as a good test example due to its well-conditioned system. The generated matrices are plotted in Fig. C.1.

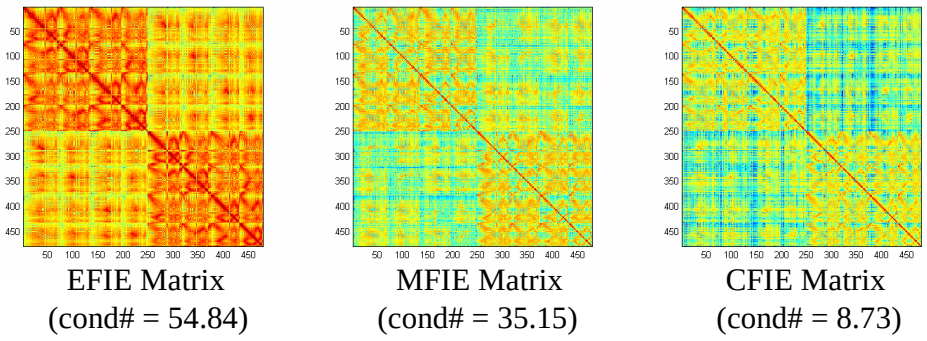


Figure C.1. Matrices generated using EFIE, MFIE, and CFIE (0.2).

For this example problem, the ILU performance is compared with several other preconditioning schemes as shown in Fig. C.2.

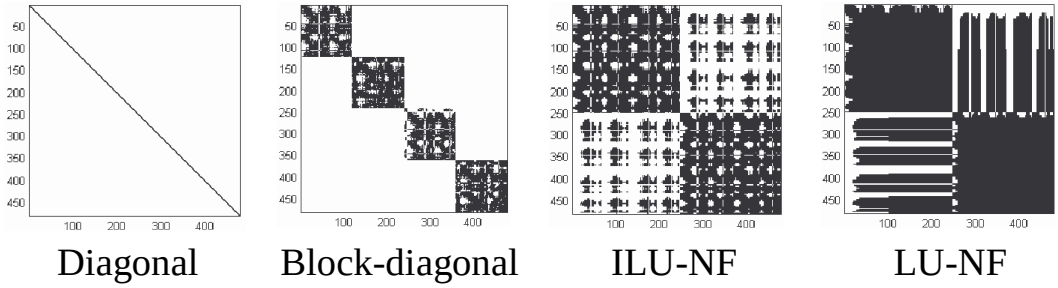


Figure C.2. Preconditioners for FMM.

First, in Fig. C.3, we present the convergence curves for BiCG, QMR, GMRES, and CGS solvers. The restart parameter used for GMRES was 20. Among these solvers, the QMR and the CGS solvers provide good performance at a low computa-

tional cost. Hence, we proceed with these two solvers to compare the ILU with other

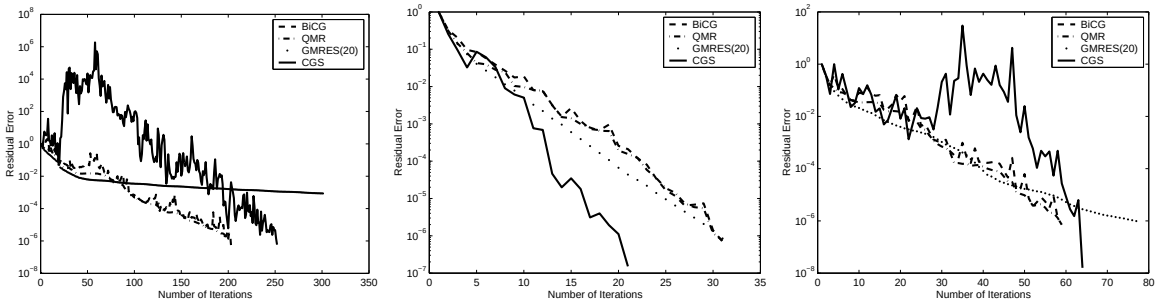


Figure C.3. Performances of different iterative solvers for the ogive problem using EFIE, CFIE(0.2), and MFIE.

preconditioners. Fig. C.4 shows the convergence curves for EFIE, CFIE, and MFIE with several preconditioning methods within the CGS solver. Similar curves for the QMR solver are given in Fig. C.5. We observe from these plots that the complete

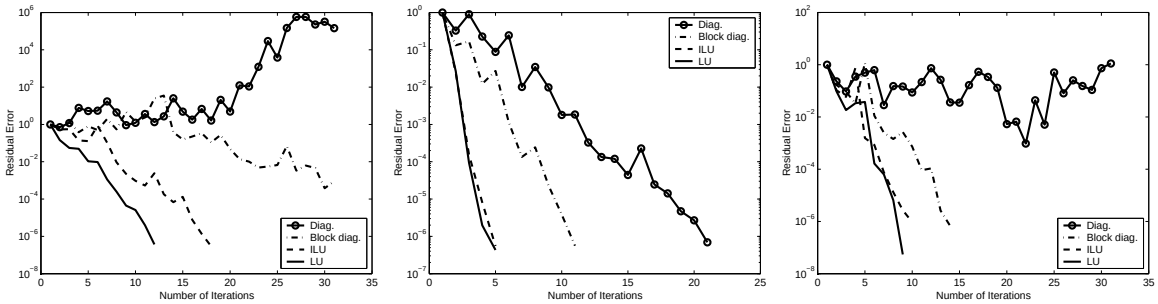


Figure C.4. Performance of CGS solver with different preconditioners for EFIE, CFIE, and MFIE.

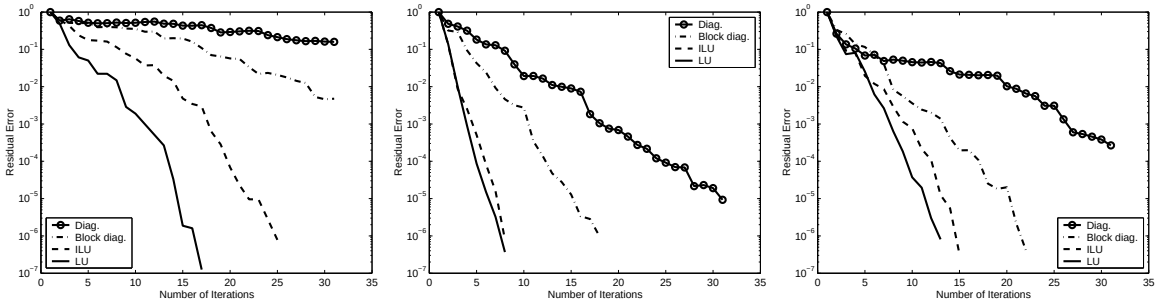


Figure C.5. Performance of QMR solver with different preconditioners for EFIE, CFIE, and MFIE.

LU preconditioner performs the best. However, due to fill-in requirements in the LU process, it requires more memory. ILU eliminates the fill-ins and for the CFIE

solutions its performance is as good as the complete LU preconditioner. Block diagonal preconditioner also reduces the number of iterations required for convergence, however, its performance is not as good as the ILU.

For a different problem geometry (namely the dart geometry shown in Fig. C.6), we plot the ILU performance in Fig. C.6 for a single level FMM solution and a 3-level FMM solution. The matrix size was $4,452$ and for single level FMM 27% of the MoM matrix was retained in the FMM near-field matrix. For the 3-level FMM solution, only 11% of the MoM matrix was retained. ILU preconditioned single level FMM solver converged in 8 CGS iterations and the 3-level FMM solver converged in 22 iterations. Without a preconditioner, 120 iterations were required for a tolerance of 10^{-6}. This example demonstrates that as more terms are retained in the ILU preconditioner, better convergence rates are achieved. Since the near-field matrix is constructed based on the physical proximities of the basis functions and since strong interactions are concentrated between closely located basis and testing functions, the near-field matrix is a very good candidate as a preconditioner.

Table C.1 summarizes the performance of the ILU preconditioner for a larger dart problem. Much like the ogive, the scatterer in this simulation has sharp edges and tips as well as smooth sections. Also, the mesh is quite distorted and non-uniform around these edges. Nevertheless, the performance of the ILU preconditioner is quite impressive. Specifically, ILU improved the convergence of the CFIE ($\alpha = 0.5$) matrix down to $N/10,000$, leading to a solution time of only 5 minutes for a 53,000 unknown system on an 8-processor SGI Origin 2000.

Based on the above performance evaluations, we can conclude that the ILU preconditioner can be used to improve the performance of iterative solvers in FMM implementations without increasing memory utilization for the preconditioner matrix. However, we must note here that as the problem size increases, the $O(N^{3/2})$ memory demand of single level FMM proves to be a bottleneck, and a multilevel

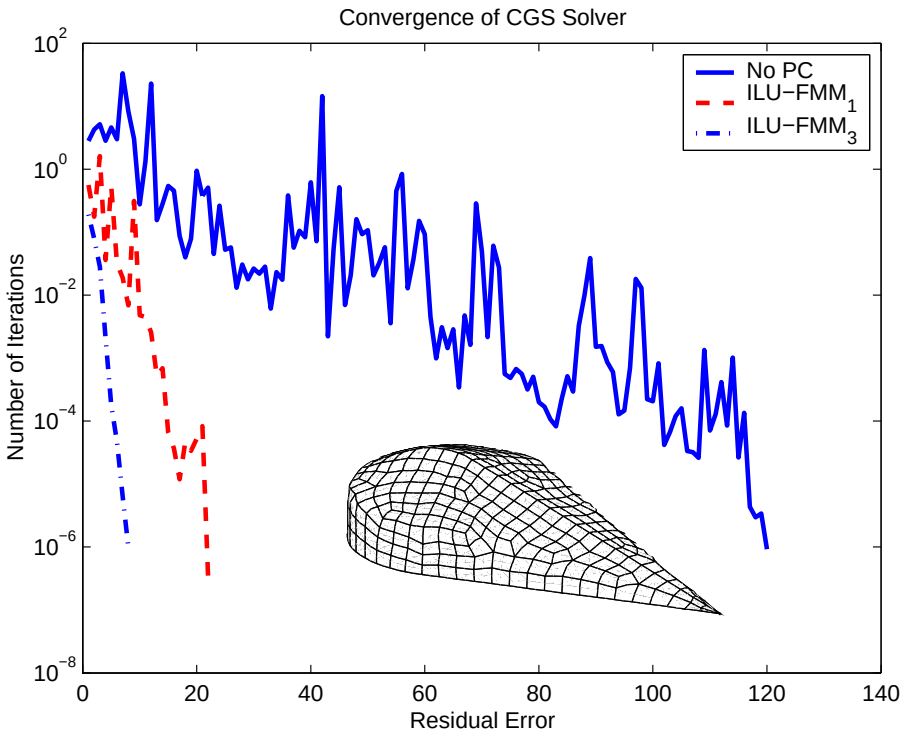


Figure C.6. Performance of CGS solver with CFIE(0.2) for dart geometry.

FMM implementation becomes necessary. Moreover, for a multilevel implementation, the size of the near-field matrix decreases, approaching to an almost diagonal system (with non-zero entries being only of $O(N)$). Hence, most of the non-zero entries in the near-field matrix of a single level FMM method are lost in the multilevel implementation. Consequently, this degrades the performance of the ILU preconditioner for the MLFMM case (as predicted by Fig. C.6). Hence, for the MLFMM implementation, we do not employ any kind of preconditioning.

Number of Unknowns	Matrix Fill Time(min.) (8 proc.)	Precond. LU Time(min.) (1 proc.)	Number of Iter.	Residual Error	Time per Solution(min.) (8 proc.)
53000	77	81	5	0.001	5

Table C.1. Performance of ILU for a large-scale complex target with sharp edges and tips on an 8-processor SGI Origin 2000.

APPENDIX D

MLFMM on Distributed Memory Parallel Computers

The most CPU intensive component in any iterative solution of a dense matrix system is the execution of the matrix-vector product $[Z]\{x\}$ as mentioned earlier. This is usually an $O(N^2)$ operation which can be reduced to $O(N \log N)$ via the MLFMM procedure [8, 44]. Thus, the MLFMM plays the most important role in the solution algorithm, and its parallelization is essential in porting the integral equation solvers to distributed computing platforms.

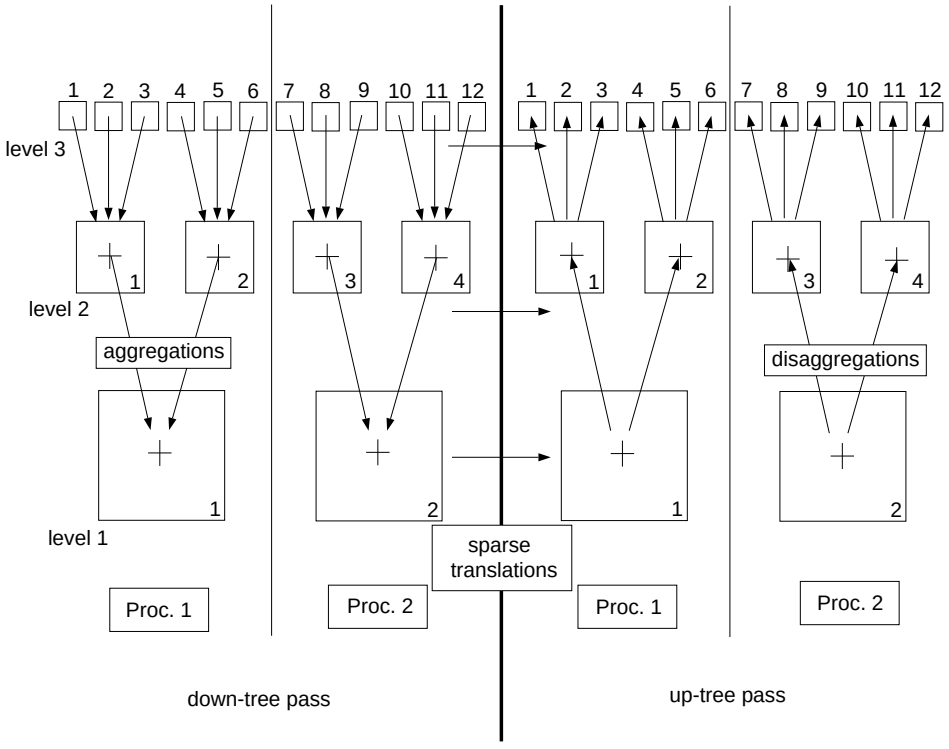


Figure D.1. A hypothetical MLFMM clustering tree.

In considering the parallelization of the matrix-vector products in the MLFMM algorithm, we must examine the three basic steps within the algorithm. First, the radiation or signature functions of each group at the $(l+1)^{th}$ level are aggregated to form the radiation functions of the parent groups, i.e. groups at the $(l)^{th}$ level. Secondly, the translation of the radiation functions to cluster centers located in the far-zone of the source cluster at the $(l)^{th}$ level, provided their parents at the $(l-1)^{th}$ level are in the near-zone of each other. The last step is the disaggregation step, where the children clusters at the $(l+1)^{th}$ level are disaggregated to compute the fields within the clusters. These three steps are depicted in Fig. D.1 for a three-level FMM tree. A successful parallel implementation of these three steps requires

- Careful balancing of the computational load among the processors, and
- Minimal communication between the processors.

Load balancing requires the distribution of the MLFMM tree structure on the nodes of the parallel process. For the sake of simplicity, let us consider the tree structure given in Fig. D.1, consisting of 2 main clusters at the coarsest level. If the tree is distributed on 2 nodes of a parallel process, assuming that each cluster at each level has the same number of far-zone clusters, we would achieve a perfectly balanced distribution. For such a case, the aggregation and disaggregation steps can be carried out on separate processors independently, without inter-processor communication. The only required communication will take place at the translation step when the source and target clusters lie on different processors. However, this requires a significant amount of communication between processors, as will be explained below. With vector processing capabilities on computing platforms, the actual work at the translation step on each processor is quite small and comparable to the time used for communication. Consequently, communication among processors becomes a bottleneck for the parallel performance of the MLFMM implementation.

To better explain the situation, let us assume that at the 1^{st} level, clusters 1

and 2 are in the near-zone of each other. Furthermore, let us assume that at the 2^{nd} level, clusters 1 and 2 are in the near-zone of each other (similarly, clusters 2 and 3 and clusters 3 and 4 are in the near-zone of each other) and clusters 3 and 4 lie in the far-zone of cluster 1 and clusters 1 and 2 lie in the far-zone of cluster 4. Likewise, at the 3^{rd} level, clusters 1 to 3 lie in the near-zone of each other and clusters 4 to 12 lie in the far-zone of clusters 1 to 3, and so on. Each processor can independently compute the aggregations at all levels starting from level 3 (we'll call this the down-tree pass). Likewise, once the pertinent data is available, the disaggregations at all levels can be computed on separate processors, independently (up-tree pass). However, inter-processor communication is necessary for the translation operations since, for example, clusters 1 and 3 (at level 2) lie on different processors and the translation operation requires the radiation function of cluster 1 to be translated onto cluster 3. For level 2, the required number of operations for the translation step is $3 \times 2L_2^2$ per processor. Thus the length of data that must be communicated between processors is $3 \times 2L_2^2$ in each direction. For level 3 the situation is slightly different. If we had assumed that clusters 2 and 3 at level 2 were in the near-field of each other, we would need to execute $2 \times 3 \times 3 \times 2L_3^2$ operations to evaluate the translated fields between the children of cluster 2 and 3 (at level 2). This simple example demonstrates the high rate of inter-processor communication needed at the translation step of the MLFMM algorithm. Via aggressive optimization in compiling the MLFMM code on computers with vector processing capabilities, the intra-processor work load can be significantly reduced. Hence, the inter-processor communication speed remains a bottleneck for optimal performance of the MLFMM on distributed memory parallel computers. Below, we present some preliminary performance data and point out the effects of inter-processor communication speed.

As an example, consider the scattering by a sphere. A balanced MLFMM grouping

leads to 5 MLFMM levels each with 8, 56, 272, 1157, and 4517 non-empty cubes. For parallelization, this tree is distributed among processors on the basis of second level clusters, and the performance test was carried out on an IBM SP3 computer having three nodes, each with eight, 375 MHz Power3 processors ($3 \times 8 = 24$ processors). The high-speed switch connecting the three nodes can deliver 500 MB/s bi-directionally to each processor. Both the matrix-fill and the iterative solution time were recorded to evaluate performance. Parallelization of the near-field matrix is straightforward and will not be discussed here.

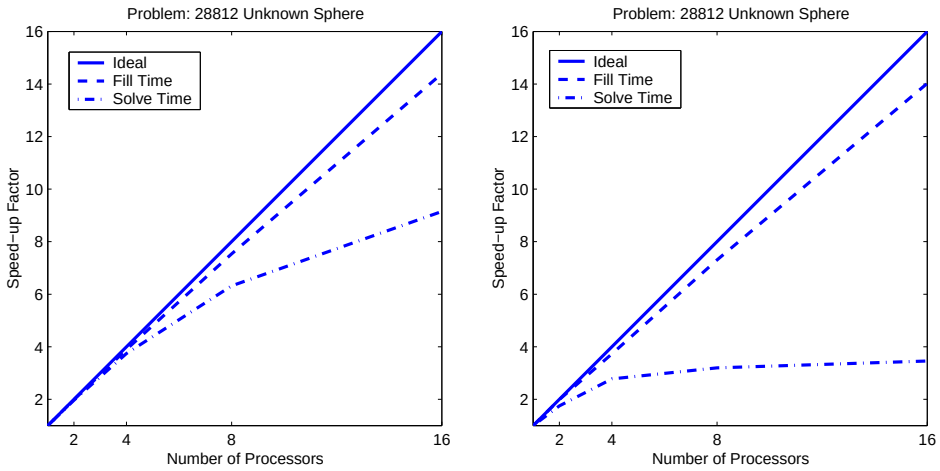


Figure D.2. Performance of parallel MLFMM for a 28,812 unknown sphere, (a) no compiler optimizations, (b) –O4 compiler optimizations.

Specifically, we show the parallel performance for a 1 m radius sphere at 0.75 GHz. This problem resulted in $28,812$ unknowns and as seen in Fig. D.2, severe performance deterioration is observed due to inter-processor communication. For the non-optimized code, better parallelization performance is observed since the evaluation of the translations has a significant computational burden at this step. However, for the optimized code (–O4 option), the inter-processor communication dominates after the 4^{th} processor is added, leading to saturation. We note here that, for a serial run (single processor), the matrix fill time demonstrated a speed-up factor of 3.7, whereas the solution time was 8.2 times faster after compiler optimizations. In light

of the above observations, we conclude that even for a very well balanced problem, inter-processor communication remains the bottleneck for the parallel performance of MLFMM on distributed memory computers. Consequently, future efforts must focus on significant re-organization of the MLFMM algorithm for massively distributed platforms. One alternative is to distribute the k-space samples among the processors rather than distributing the basis functions themselves. Since the translation operation is a diagonal operation, meaning every outgoing plane-wave amplitude is translated over the testing domain to form the incoming plane-wave amplitude independent of the rest of the plane-wave directions. This will eliminate any communication between processors at the translation stage, however, a new interpolation scheme is necessary since not all of the plane-wave directions will be available on each processor at a given level.

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] J.-M. Jin, J. L. Volakis, and V. V. Liepa, "A moment method solution of a volume-surface integral equation using isoparametric elements and point matching (TE scattering)", *IEEE Trans. Microwave Theory Tech.*, vol. 37, no. 10, pp. 1641–1645, Oct. 1989.

[2] S. M. Rao, D. R. Wilton, and A. W. Glisson, "Electromagnetic scattering by surfaces of arbitrary shape", *IEEE Trans. Antennas Propagat.*, vol. 30, no. 3, pp. 409–418, May 1982.

[3] B. M. Kolundzija and B. D. Popovic, "Entire-domain Galerkin method for analysis of metallic antennas and scatterers", *IEE Proc.-H*, vol. 140, no. 1, pp. 1–10, 1993.

[4] J. M. Song and W. C. Chew, "Moment method solutions using parametric geometry", *Journal of Electromagnetic Waves and Applications*, vol. 9, no. 1/2, pp. 71–83, 1995.

[5] L. Valle, F. Rivas, M. F. Catedra, "Combining the moment method with geometrical modeling by nurbs surfaces and bezier patches", *IEEE Trans. Antennas Propagat.*, vol. 42, no. 3, pp. 373–381, March 1994.

[6] K. Sertel and L. Gürel, "A comparison of surface modeling techniques", *IEEE AP-S Conference Digest*, vol. 3, pp. 1834–1837, 1997.

[7] S. Wandzura, "Electric current basis functions for curved surfaces", *Electromagnetics*, vol. 12, no. 1, pp. 77–91, Feb./March 1992.

[8] R. Coifman, V. Rokhlin, and S. Wandzura, "The fast multipole method for the wave equation: A pedestrian prescription", *IEEE Antennas and Propagat. Mag.*,

vol. 35, no. 3, pp. 7–12, June 1993.

[9] R. L. Wagner and C. W. Chew, "A ray-propagation fast multipole algorithm", *Microwave Opt. Tech. Lett.*, vol. 7, no. 10, pp. 435–438, July 1994.

[10] R. J. Burkholder and D. H. Kwon, "High frequency asymptotic acceleration of the fast multipole method", *Radio Sci.*, vol. 31, no. 5, pp. 1199–1206, Sep./Oct. 1996.

[11] C. C. Lu and W. C. Chew, "Fast far-field approximation for calculating the rcs of large objects", *Microwave Opt. Tech. Lett.*, vol. 8, no. 5, pp. 238–241, April 1995.

[12] L. Rossi, P. J. Cullen, and C. Brennan, "Implementation of a multilevel fast far-field algorithm for solving electric-field integral equations", *IEE Proc. Microwaves, Antennas and Propagation*, vol. 147, no. 1, pp. 19–24, Feb. 2000.

[13] J. M. Song, C. C. Lu, and W. C. Chew, "Multilevel fast multipole algorithm for electromagnetic scattering by large complex objects", *IEEE Trans. Antennas Propagat.*, vol. 45, no. 10, pp. 1488–1493, Oct. 1997.

[14] T. J. Peters and J. L. Volakis, "Application of the conjugate gradient FFT method to scattering from thin planar material plates", *IEEE Trans. Antennas Propagat.*, vol. 36, no. 4, pp. 518–526, April 1988.

[15] N. N. Bojarski, "k–space formulation of the electromagnetic scattering problem", *Technical Report AFAL–TR–71–5, U. S. Air Force*, March 1971.

[16] A. P. M. Zwanborn and P. M. van den Berg, "A weak form of the conjugate gradient fft method for plate problems", *IEEE Trans. Antennas Propagat.*, vol. 39, no. 2, pp. 224–228, Feb. 1991.

[17] C. Y. Shen, K. J. Glover, M. I. Sancer, and A. D. Varvatsis, "The discrete fourier transform method of solving differential–integral equations in scattering theory", *IEEE Trans. Antennas Propagat.*, vol. 37, no. 8, pp. 1032–1041, Aug. 1989.

[18] J. L. Volakis, A. Chatterjee and L. Kempel, *Finite Element Methods for Electromagnetics*, New York: IEEE Press, 1998.

[19] G. E. Antilla and N. G. Alexopoulos, "Scattering from complex 3D geometries by a curvilinear hybrid finite element-integral equation approach", *J. Opt. Soc. Am. A.*, vol. 11, no. 4, pp. 1445–1457, 1994.

[20] T. Özdemir, and J. L. Volakis, "Triangular prisms for edge-based vector finite elements analysis of conformal antennas", *IEEE Trans. Antennas Propagat.*, vol. 45, no. 5, pp. 788–797, May 1997.

[21] T. F. Eibert, J. L. Volakis, D. R. Wilton, and D. R. Jackson, "Hybrid FE/BI modeling of 3D doubly periodic structures utilizing triangular prismatic elements and a MPIE formulation accelerated by the Ewald transformation", *IEEE Trans. Antennas Propagat.*, vol. 47, no. 5, pp. 843–850, Mar. 1999.

[22] A. J. Poggio, *Integral Equation Solutions of Three Dimensional Scattering Problems in Computer Techniques for Electromagnetics*, Oxford, U.K.: Permagon, 1973.

[23] D. E. Livesay and K. Chen, "Electromagnetic fields induced inside arbitrarily shaped biological bodies", *IEEE Trans. Microwave Theory Tech.*, vol. 22, no. 12, pp. 1273–1280, Dec. 1974.

[24] D. H. Schaubert, D. R. Wilton, and A. W. Glisson, "A tetrahedral modeling method for electromagnetic scattering by arbitrarily shaped inhomogeneous dielectric bodies", *IEEE Trans. Antennas Propagat.*, vol. 32, no. 1, pp. 77–85, Jan. 1984.

[25] X. Yuan, "Three-dimensional electromagnetic scattering from inhomogeneous objects by the hybrid moment and finite element method", *IEEE Trans. Microwave Theory Tech.*, vol. 38, no. 8, pp. 1053–1058, Aug. 1990.

[26] T. Eibert and V. Hansen, "Calculation of unbounded field problems in free space by a 3-D FEM/BEM-hybrid approach", *Journal of Electromagnetic Waves and*

Applications, vol. 10, no. 1, pp. 61–78, 1996.

[27] R. F. Harrington, *Time-Harmonic Electromagnetic Fields*, New York: McGraw-Hill, 1961.

[28] C. T. Tai, "A note on the integral equations for the scattering of a plane wave by an electromagnetically permeable body", *Electromagnetics*, vol. 5, no. 1, pp. 79–88, 1985.

[29] J. H. Richmond, "Scattering by a dielectric cylinder of arbitrary cross section shape", *IEEE Trans. Antennas Propagat.*, vol. 13, no. 3, pp. 334–341, May 1965.

[30] A. F. Peterson, "Analysis of heterogeneous electromagnetic scatterers: research progress of the past decade", *Proc. IEEE*, vol. 79, no. 10, pp. 1431–1441, Oct. 1991.

[31] R. D. Graglia, P. L. E. Uslenghi, and R. S. Zich, "Moment method with isoparametric elements for three-dimensional anisotropic scatterers", *Proc. IEEE*, vol. 77, no. 5, pp. 750–760, May 1989.

[32] J. L. Volakis, "Alternative field representations and integral equations for modeling inhomogeneous dielectrics", *IEEE Trans. Microwave Theory Tech.*, vol. 40, no. 3, pp. 604–608, March 1992.

[33] A. F. Peterson and D. R. Wilton, *Finite Element Software in Microwave Engineering, Ch. 5*, Edited by Itoh, Pelosi, and Silvester, New York: Wiley, 1996.

[34] D. F. Rogers and J. A. Adams, *Mathematical Elements for Computer Graphics*, New York: McGraw Hill, 1990.

[35] G. Farin, *Curves and Surfaces for Computer Aided Geometric Design*, San Diego: Academic Press, 1997.

[36] R. D. Graglia, D. R. Wilton, and A. F. Peterson, "Higher order interpolatory vector bases for computational electromagnetics", *IEEE Trans. Antennas Propagat.*, vol. 45, no. 3, pp. 329–342, March 1997.

[37] A. W. Glisson and D. R. Wilton, "Simple and efficient numerical methods

for problems of electromagnetic radiation and scattering from surfaces", *IEEE Trans. Antennas Propagat.*, vol. 28, no. 5, pp. 593–603, Sep. 1980.

[38] E. Jorgensen, J. L. Volakis, P. Meincke, and O. Breinbjerg, "Higher-order hierarchical Legendre basis functions for iterative integral equation solvers with curvilinear surface modeling", *IEEE International Symposium on Antennas and Propagation, San Antonio, TX., Digest*, pp. 618–621, 2002.

[39] Y. Saad, *Iterative Methods for Sparse Linear Systems*, Boston: PWS Pub. Co., 1996.

[40] K. Sertel and J. L. Volakis, "Incomplete LU preconditioner for FMM implementation", *Microwave Opt. Tech. Lett.*, vol. 26, no. 4, pp. 265–267, when 2000.

[41] R. F. Harrington, *Field Computation by Moment Methods*, New York: Macmillan, 1968.

[42] S. S. Bindiganavale and J. L. Volakis, "Comparison of three FMM techniques for solving hybrid FE/BI systems", *IEEE Antennas and Propagat. Mag.*, vol. 39, no. 4, pp. 47–60, August 1997.

[43] N. Lu and J.–M. Jin, "Application of fast multipole method to finite–element boundary–integral solution of scattering problems", *IEEE Trans. Antennas Propagat.*, vol. 44, no. 6, pp. 781–786, June 1996.

[44] W. C. Chew, J. Jin, C. Lu, E. Michielssen, and J. M. Song, "Fast solution methods in electromagnetics", *IEEE Trans. Antennas Propagat.*, vol. 45, no. 3, pp. 533–543, Mar. 1997.

[45] X. Q. Sheng, J.-M. Jin, J. M. Song, C. C. Lu, and W. C. Chew, "On the formulation of hybrid finite-element and boundary-integral methods for 3-D scattering", *IEEE Trans. Antennas Propagat.*, vol. 46, no. 3, pp. 303–311, Mar. 1998.

[46] E. Bleszynski, M. Bleszynski, and T. Jaroszewicz, "AIM: Adaptive integral method for solving large-scale electromagnetic scattering and radiation problems", *Radio Sci.*, vol. 31, no. 5, pp. 1225–1251, Sep./Oct. 1996.

[47] H. T. Anastassiu, M. Smelyanskiy, S. Bindiganavale, and J. L. Volakis, "Scattering from relatively flat surfaces using the adaptive integral method", *Radio Sci.*, vol. 33, no. 1, pp. 7–16, Jan./Feb. 1998.

[48] S. S. Bindiganavale and J. L. Volakis, "Scattering and radiation from planar structures containing small features using the adaptive integral method (AIM)", *IEEE Trans. Antennas Propagat.*, vol. 46, no. 12, pp. 1867–1878, Dec. 1998.

[49] M. Abramowitz and I. A. Stegun, *Handbook of Mathematical Functions*, National Bureau of Standards, 1972.

[50] P. Griffits and I. D. Hill, Ed., *Applied Statistics Algorithms*, Chichester: Ellis Horwood, 1985.

[51] R. J.-Chien and B. K. Alpert, "A fast spherical filter with uniform resolution", *J. Comp. Phys.*, vol. 136, pp. 580–584, 1997.

[52] N. Yarvin and V. Rokhlin, "A generalized one-dimensional fast multipole method with application to filtering of spherical harmonics", *J. Comp. Phys.*, vol. 147, pp. 594–609, 1998.

[53] E. Darve, "The fast multipole method: numerical implementation", *J. Comp. Phys.*, vol. 160, no. 1, pp. 195–240, May 2000.

[54] J.-M. Jin, J. L. Volakis, and J. D. Collins, "A finite-element-boundary-integral method for scattering and radiation by two- and three-dimensional structures", *IEEE Antennas and Propagat. Mag.*, vol. 33, no. 3, pp. 22–32, June 1991.

[55] J. L. Volakis, T. Ozdemir, and J. Gong, "Hybrid finite-element methodologies for antennas and scattering", *IEEE Trans. Antennas Propagat.*, vol. 45, no. 3, pp. 493–507, March 1997.

[56] W. C. Chew, J.-M. Jin, E. Michielssen, and J. Song, ed., *Fast and Efficient Algorithms in Computational Electromagnetics*, Boston: Artech House, 2001.

[57] J. L. Volakis, T. F. Eibert, and K. Sertel, "Fast integral methods for conformal antenna and array modeling in conjunction with hybrid finite element formula-

tions", *Radio Sci.*, vol. 35, no. 2, pp. 537–546, March/April 2000.

[58] X. Q. Sheng, J.-M. Jin, J. M. Song, W. C. Chew, and C. C. Lu, "Solution of combined-field integral equation using multilevel fast multipole algorithm for scattering by homogeneous bodies", *IEEE Trans. Antennas Propagat.*, vol. 46, no. 11, pp. 1718–1726, Nov. 1998.

[59] R. W. Kindt, K. Sertel, E. Topsakal, and J. L. Volakis, "Array decomposition method for the accurate analysis of finite arrays", To appear in IEEE Trans. Antennas Propagat., 2003.

[60] E. Arvas, A. Rahhal–Arabi, A. Sadigh, and S. M. Rao, "Scattering from multiple conducting and dielectric bodies of arbitrary shape", *IEEE Antennas and Propagat. Mag.*, vol. 33, no. 2, pp. 29–36, April 1991.

[61] C. C. Lu, "Indoor radio wave propagation modeling using multilevel fast multipole algorithm", *Microwave Opt. Tech. Lett.*, vol. 29, no. 3, pp. 168–175, May 2001.

[62] W. C. Chew, *Waves and Fields in Inhomogeneous Media*, New York: Piscataway, IEEE Press, 1995.

[63] K. Sertel and J. L. Volakis, "Method of moments solution of volume integral equations using parametric geometry", *Radio Sci.*, vol. 37, no. 1, pp. 1–7, 2002.