

An Approach to the Implementation of Laplace and a Broadband Helmholtz Fast Multipole Method as an Application Independent Library

Sanjay Velamparambil

Ansys Inc., 1995 N. 57th Court, Suite 100, Boulder, CO-80301, USA
sanjay.velamparambil@ansys.com

Abstract – In this paper, we propose an approach to develop an application independent library of Laplace and Helmholtz fast multipole method (FMM) that can be used in different applications. For this purpose, we consider a generalized problem and a corresponding canonical problem (defined below). In the first main contribution, we show that it is possible to capture the essential characteristics of the canonical summation from sampling the values of certain potentials or signature functions. In the second main contribution, we show that partial derivatives of arbitrary orders acting on the far field can be represented as product of sparse matrices within the library, transparent to the user. Combining the two ideas, we show that once the FMM is configured to compute the canonical summation, the same setup can be used to work with a much wider, general class of problems.

Index Terms – Fast multipole methods, integral equations.

I. INTRODUCTION

In this paper, we describe an approach to develop a *stand-alone, application independent library* of the fast multipole method (FMM) [1–3] to be used in different physical applications and developed by multiple teams with differing backgrounds. In particular, we are concerned with the evaluation of summations of the form:

$$y_m = \sum_{n=1}^N A_{mn} x_n, \quad 1 \leq m \leq M, \quad (1a)$$

with:

$$A_{mn} = \int_{\Omega} g_m(\vec{x}) \mathcal{M}_u \int_{\Omega'} f_n(\vec{x}') \mathcal{L}'_v G(\vec{x}, \vec{x}') d\Omega' d\Omega, \quad (1b)$$

$$u, v = x, y, z; \vec{x}, \vec{x}' \in \mathbb{R}^3,$$

where \mathcal{M}_u and \mathcal{L}_v are linear partial differential operators with constant coefficients and:

$$G(\vec{x}, \vec{x}') = \begin{cases} \frac{1}{\|\vec{x} - \vec{x}'\|} & \text{for the Laplace Equation} \\ \frac{e^{ik\|\vec{x} - \vec{x}'\|}}{\|\vec{x} - \vec{x}'\|} & \text{for the Helmholtz Equation.} \end{cases} \quad (1c)$$

Note that, in general, $\Omega' \neq \Omega$. Furthermore, following the terminology used by the method of moments (MoM) practitioners in electromagnetics, we shall refer to $f_j(\vec{x})$ and $g_i(\vec{x})$ as the *basis* and *testing* functions, respectively, with the understanding that they can be Dirac δ -functions to represent “point” particles.

Such computations can arise from a wide variety of applications, such as electromagnetics, acoustics and elastic wave scattering. Since its development in the 1980s [4, 5] for the Laplace equation to the elegant extension to the Helmholtz and Maxwell equations in the 1990s [2, 3, 6], FMM and its multilevel variants have transformed computational physics, especially, computational electromagnetics. It is beyond the scope of this paper to provide a comprehensive overview of these remarkable developments. Instead, we refer the reader to the recent review by He *et al.* [7].

Although the theory of FMM has been well established and many high quality implementations exist, both in the public domain and in proprietary products, the development of an application independent library is still a challenging problem due to various *practical* constraints such as:

- Different formulations and physics lead to different choices for \mathcal{M}_u and \mathcal{L}_v .
- Different domain and geometry; the integrals could be defined over surfaces or volumetric regions, discretized with surface elements such as flat or curvilinear triangles and/or quadrilaterals and volume elements such as tetrahedra. It is also possible to have mixed formulations involving both surface elements and volume elements. The quality of such a mesh can often be poor, with highly non-uniform elements.
- Different basis and testing procedures in reducing the integral equations to a matrix equation; note that each basis and testing function can be supported by one or more mesh elements.
- Familiarity/background knowledge of the library user. Even though one can expect the users to have a sound understanding of the integral equation and the underlying physics, it is not necessary that they

have a sufficient understanding of FMM to develop the components needed to integrate their code with a library.

We illustrate this with an example. In an implementation of Laplace FMM that uses spherical harmonics, the library user may have to provide the multipole coefficients of basis/testing functions. Although standard implementations of spherical harmonics are available, it may be tricky to ensure consistency between third party implementation of spherical harmonics and the ones used in the FMM implementation.

One of the ways to address these issues is to replace the basis and testing functions with weighted (point) “particles” as shown in Fig. 1. Although this approach simplifies the implementation of the library, it has several drawbacks. First, as illustrated in Fig. 1, replacing mesh elements and basis/testing functions with weighted particles generally leads to *incompatible partitioning* of the geometry, requiring error-prone bookkeeping, complicated data interchange between the application and the library, and messy corrections in the near field. In addition, given that the number of quadrature points is generally larger than the number of basis/testing functions, the matrix as seen by the library tends to be larger as well, potentially affecting the performance.

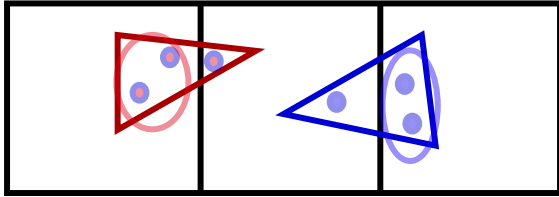


Fig. 1. The two triangles are near to each other, but some of the quadrature points can still be well separated (shown in ellipses) leading to tricky bookkeeping and near field corrections.

Therefore, the objective of this paper is to present an approach that can handle a wide class of problems through the use of good *abstractions* to capture the physics, the formulation and the discretization. The main contributions of this work are:

- Abstractions to characterize the geometry of the elements in a way that allows consistent handling of non-uniform meshes.
- A technique to compute required multipole coefficients of the shape/basis and testing functions from *sampled values* of their corresponding potentials or signature functions which are easy to define and simple to evaluate, without involving any special functions.

- We show that the partial derivatives of the far field, represented by multipole or plane wave expansions, can be represented as products of sparse matrices independent of the basis and testing functions allowing the user to compute these derivatives without having to directly work with multipole expansions.

Finally, we demonstrate that once the library is integrated into an existing code to compute what we call the *canonical problem*, defined in equation (4), the same implementation can be used to evaluate summations in equation (1) without modifying the library and without adding new subroutines to the application code.

II. PRELIMINARIES

A. General class of problems

We assume that the reader is familiar with the basic mathematical formulation and the octree based computational structure of the Laplace and Helmholtz multi-level fast multipole methods (MLFMM). Except when specificity is needed, we shall use the abbreviation MLFMM to refer to either of the two versions.

In the context of MLFMM, we begin by noting that if g_m and f_n in equation (1) belong to well separated boxes, the Green’s function in equation (1) does not involve any singularities. Since:

$$\frac{\partial G}{\partial u} = -\frac{\partial G}{\partial u'}, u = x, y, z, \quad (2)$$

whenever $\vec{x} \neq \vec{x}'$, the differential operator \mathcal{L}'_v appearing on the source point \vec{x}' can be transferred to the observation point \vec{x} when g_m and f_n are *geometrically well separated*. Therefore, for far field computations, it is possible to transform equation (1b) to a linear combination of integrals of the form:

$$A_{mn} = \int_{\Omega} g_m(\vec{x}) \mathcal{D}_{pqr} \int_{\Omega'} f_n(\vec{x}') G(\vec{x}, \vec{x}') d\Omega' d\Omega, \quad (3a)$$

$$\text{where } \mathcal{D}_{pqr} = \frac{\partial^{p+q+r}}{\partial x^p \partial y^q \partial z^r}, \vec{x} \neq \vec{x}', \quad (3b)$$

with p, q and r being non-negative integers.

One of the key results presented in this work is that the differential operator \mathcal{D}_{pqr} can be *exactly* represented as a product of certain sparse matrices in the calculation of far interactions. Anticipating this result, we define a new summation:

$$y_m = \sum_{n=1}^N P_{mn} x_n, \quad 1 \leq m \leq M, \quad (4a)$$

$$\text{where } P_{mn} = \int_{\Omega} g_m(\vec{x}) \int_{\Omega'} f_n(\vec{x}') G(\vec{x}, \vec{x}') d\Omega' d\Omega, \quad (4b)$$

and refer to it as the *canonical problem*.

B. Some notations and definitions

If $r = \|\vec{x}\|$ and $\hat{\mathbf{x}} = \frac{\vec{x}}{\|\vec{x}\|}$ for $\vec{x} \in \mathbb{R}^3$, we shall often write $\vec{x} = (r, \hat{\mathbf{x}}) = (r, \theta, \phi)$ in spherical coordinates. We shall

denote the unit sphere centered at the origin by $\Omega_0 = \{\vec{r} \in \mathbb{R}^3 : |\vec{r}| = 1\}$ and will typically use the symbol $\hat{\mathbf{s}} = (1, \theta, \phi)$ to denote a vector on Ω_0 .

Following [8, 9], we define the associated Legendre function $P_n^m(x)$ of degree $n > 0$ and order $-n \leq m \leq n$ and by the formula:

$$P_n^m(x) = (-1)^m \frac{(1-x^2)^{m/2}}{2^n n!} \frac{d^{n+m}}{d^{n+m} x} (x^2-1)^n. \quad (5a)$$

Following Messiah [9], we define the spherical harmonic $Y_n^m(\theta, \phi)$ of degree n and order m by the relation:

$$Y_n^m(\theta, \phi) = \left[\frac{2n+1}{4\pi} \right]^{1/2} \left[\frac{(n-m)!}{(n+m)!} \right]^{1/2} P_n^m(\cos \theta) e^{im\phi}. \quad (5b)$$

We have:

$$Y_n^{-m}(\theta, \phi) = (-1)^m [Y_n^m(\theta, \phi)]^*, \quad (5c)$$

$$\int_0^\pi \int_0^{2\pi} Y_n^m(\theta, \phi) Y_{n'}^{m'}(\theta, \phi) \sin \theta d\theta d\phi = \delta_{nn'} \delta_{mm'}, \quad (5d)$$

where $*$ denotes the complex conjugate.

Let $\vec{x} = r\hat{\mathbf{x}} = (r, \theta, \phi) \in \mathbb{R}$. We define *inner and outer* multipole functions for the Laplace equation by the formulae [10]:

$$O_n^m(\vec{x}) = \sqrt{\frac{4\pi}{2n+1}} \frac{(-1)^n i^{-m} Y_n^m(\theta, \phi)}{A_n^m r^{n+1}}, \quad (6a)$$

$$I_n^m(\vec{x}) = \sqrt{\frac{4\pi}{2n+1}} i^m A_n^m r^n Y_n^m(\theta, \phi), \quad (6b)$$

$$\text{where } A_n^m = \frac{(-1)^n}{\sqrt{(n-m)!(n+m)!}}. \quad (6c)$$

Similarly, for the Helmholtz equation, we define the inner and outer functions, denoted by $\tilde{I}_n^m(\vec{x})$ and $\tilde{O}_n^m(\vec{x})$, respectively, by the formulae [10]:

$$\tilde{I}_n^m(\vec{x}) = j_n(kr) Y_n^m(\hat{\mathbf{x}}), \quad (7a)$$

$$\tilde{O}_n^m(\vec{x}) = h_n(kr) Y_n^m(\hat{\mathbf{x}}), \quad (7b)$$

where $j_n(kr)$ and $h_n(kr)$ are, respectively, spherical Bessel and Hankel functions of the first kind and of order n .

III. ABSTRACTIONS FOR THE FAR FIELD COMPUTATIONS

We first develop the abstractions needed to evaluate the canonical summation in equation (4). Assuming the existence of an octree structure, let B_s and B_r be the boxes in which the basis function f_j and the testing func-

tion g_i , respectively, reside.¹ Referring to Fig. 2, let:

$$B_s \subset D_s \stackrel{\text{def}}{=} \{\vec{x} \in \mathbb{R}^3 \mid |\vec{x} - \vec{c}_s| < R_s\}, \quad (8)$$

$$B_r \subset D_r \stackrel{\text{def}}{=} \{\vec{x} \in \mathbb{R}^3 \mid |\vec{x} - \vec{c}_r| < R_r\}. \quad (9)$$

We assume that $\text{supp } f_j \subset D_s$ and $\text{supp } g_i \subset D_r$ and that $\bar{D}_s \cap \bar{D}_r = \emptyset$.

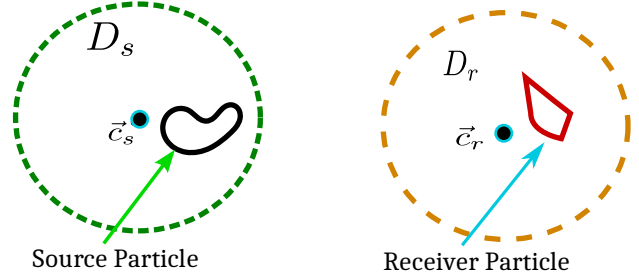


Fig. 2. The basis function, f_j , and the testing function, g_i , (“generalized particles”) are enclosed in spheres D_s , with center at \vec{c}_s and D_r , with center at \vec{c}_r , respectively, under the condition that $D_s \cap D_r = \emptyset$.

To represent the matrix element in equation (4), we first let:

$$\psi_j(\vec{x}) = \int_{\text{supp } f_j} f_j(\vec{x}') G(\vec{x}, \vec{x}') d\vec{x}', \quad (10a)$$

$$P_{ij} = \int_{\text{supp } g_i} g_i(\vec{x}) \psi_j(\vec{x}) d\vec{x}. \quad (10b)$$

For a given accuracy $\varepsilon > 0$, let $N \geq 0$ be given. Let $O_n^m(\vec{x})$ and $I_n^m(\vec{x})$, respectively, denote the outer and inner multipoles of either Laplace or Helmholtz kernel. Then, we recall the following foundational results from the theory of MLFMM [2, 10]:

1. The potential $\psi_j(\vec{x})$ can be approximated by a finite *outer* multipole series valid *outside* D_s :

$$\psi_j(\vec{x}) \approx \sum_{n=0}^N \sum_{m=-n}^n \alpha_n^m O_n^m(\vec{x} - \vec{c}_s), \quad \vec{x} \notin \bar{D}_s. \quad (11a)$$

2. The potential $\psi_j(\vec{x})$ can be approximated as an *inner* multipole series valid *inside* D_r :

$$\psi_j(\vec{x}) \approx \sum_{n=0}^{N'} \sum_{m=-n}^n \beta_n^m I_n^m(\vec{x} - \vec{c}_r), \quad \vec{x} \in D_r. \quad (11b)$$

3. There exists a linear operator, \mathcal{T} , called a translation operator, relating the coefficients $\{\alpha_n^m\}$ and $\{\beta_n^m\}$ by:

$$\beta_n^m = \sum_{p=0}^N \sum_{q=-p}^q \mathcal{T}(n, m; p, q) \alpha_p^q. \quad (11c)$$

¹Note that we are not providing a strict definition for the term “reside.” It is simply a rule that the user provides to determine if a given basis/testing function can be considered as belonging to a given box. For example, in the case of point particles, if the location of the particle falls into a box, it can be considered to be a resident of that box.

4. To evaluate P_{ij} , we substitute equation (11b) into equation (10b), and obtain:

$$P_{ij} \approx \sum_{n=0}^{\min\{N, N'\}} \sum_{m=-n}^n \beta_n^m \rho_n^m, \quad (11d)$$

$$\rho_n^m \stackrel{\text{def}}{=} \int_{\text{supp } g_i} g_i(\vec{x}) \mathcal{I}_n^m(\vec{x} - \vec{c}_r) d\vec{x}. \quad (11e)$$

Borrowing the terminology from Laplace FMM parlance, we shall refer to α_n^m as the ‘‘Q2M’’ coefficients and ρ_n^m as the ‘‘L2P’’ coefficients.

Expressing the Q2M and L2P coefficients in terms of multipole functions presents a challenge to implementation when multiple teams are involved. For example, a team developing a stand-alone library of FMM, to be used by multiple product teams, may not be deeply familiar with the applications themselves. On the other hand, a product team, although experts in, say, integral equations, may have little or no familiarity with FMM. Given that it is these coefficients that couple the application side (integral equation) with the FMM implementation, it is imperative that they are evaluated correctly and efficiently.

An obvious way to do this is to let the FMM library be aware of the basis and testing functions, and thus let the library evaluate these functions. This has the advantage that if the library decides to change, say, the definitions of inner functions, it will remain transparent to the application. However, this approach has the severe disadvantage that the library is deeply tied to a particular class of basis and testing functions. It is then left with the task of potentially adding a new class of basis and testing functions for every new application. This is inefficient and will eventually make the code unmaintainable.

A second approach is to let the application evaluate the coefficients (α_n^m, ρ_n^m) using formulae similar to those in equation (11d) and then hand-over the results to the FMM kernel. This has the great advantage that the FMM library remains completely decoupled from the application. However, this approach has the disadvantage that the application team must have some familiarity with FMM and multipoles. This may not always be the case.

Therefore, our goal is to evaluate the coefficients α_n^m and ρ_n^m within the library without exposing the functions ($\mathcal{O}_n^m, \mathcal{I}_n^m$) to the application.

We now present an approach that eliminates these difficulties and it is inspired by ideas from [11–14].

A. Evaluation of Q2M and L2P coefficients for the Laplace equation

We begin by noting the basis function f_j is defined by the application and that the application ‘‘knows’’ how to evaluate $\psi_j(\vec{x})$ in equation (10a) at any point $\vec{x} \in \mathbb{R}^3$. Moreover, if $\vec{x} \notin \text{supp } f_j$, the integral can be accurately evaluated using simple quadrature.

Consider a sphere $S \stackrel{\text{def}}{=} \{\vec{x} \in \mathbb{R}^3 \mid |\vec{x} - \vec{c}_s| = R > R_s\}$ so that $S \cap \text{supp } f_j = \emptyset$. Let $\vec{x} \in S$. Then $\vec{x} = \vec{c}_s + R\hat{s}$ where $\hat{s} \in \Omega_0$. Then, substituting the definition of \mathcal{O}_n^m from equation (6a) into equation (11a) and using the orthogonality of $Y_n^m(\hat{s})$ from equation (5d), we obtain the well known relation (see, for example [14, equations 4-5]):

$$\alpha_n^m = i^m \sqrt{\frac{2n+1}{4\pi(n-m)!(n+m)!}} R^{n+1} \int_{\Omega_0} \psi_j(\vec{c}_s + R\hat{s}) Y_n^{m*}(\hat{s}) d\Omega_s. \quad (12)$$

Now assume that the degree of approximation used in the FMM is $p > 0$. Then, the maximum degree of spherical harmonics under the integral that is relevant to the computations is $2p$. Therefore, if we use a quadrature, such as a composite Gauss-Legendre-Trapezoidal rule, that can integrate spherical harmonics of degree $2p$ exactly, we will be able to compute the coefficients exactly as well. Let N_q be the number of quadrature points, $\{(\hat{s}_k, w_k)\}$, be quadrature points and their corresponding weights on Ω_0 , respectively, and let $\vec{x}_k = \vec{c}_s + R\hat{s}_k$, for $1 \leq k \leq N_q$. Then, equation (12) can be approximated as:

$$\tilde{\alpha}_n^m \approx i^m \sqrt{\frac{2n+1}{4\pi(n-m)!(n+m)!}} R^{n+1} \sum_{k=1}^{N_q} w_k \psi_j(\vec{x}_k) Y_n^{m*}(\hat{s}_k). \quad (13)$$

Therefore, to compute the coefficients α_n^m , all the library needs are the values of $\psi_j(\vec{x})$, given by equation (10), at a set points on a specified sphere, which the application can readily supply. As remarked earlier, since the observation points \vec{x}_k are chosen to be far away from the support of f_j , the integrand in (10a) is non-singular and hence $\psi_j(\vec{x}_k)$ can be evaluated using simple quadrature rules.

Using the addition theorem [10]:

$$\frac{1}{\|\vec{x} - \vec{x}'\|} = \sum_{n=0}^{\infty} \sum_{m=-n}^n (-1)^n I_n^{-m}(\vec{x}' - \vec{c}_s) \mathcal{O}_n^m(\vec{x} - \vec{c}_s), \quad (14a)$$

for $\|\vec{x} - \vec{c}_s\| > \|\vec{x}' - \vec{c}_s\|$ in equation (10a) and comparing with equation (11a), it is easy to see that:

$$\alpha_n^m = (-1)^n \int_{\text{supp } f_j} f_j(\vec{x}') I_n^{-m}(\vec{x}' - \vec{c}_s) d\vec{x}'. \quad (14b)$$

Comparing equation (14b) with the definition of ρ_n^m in equation (11d), we see that:

$$\rho_n^m = (-1)^n \alpha_n^{-m}, \quad (15)$$

for the f_j if it were also used as a testing function in a Galerkin MoM. Since:

$$\begin{aligned} & \int_{\Omega} g_m(\vec{x}) \int_{\Omega'} f_n(\vec{x}') G(\vec{x}, \vec{x}') d\Omega' d\Omega \\ &= \int_{\Omega'} f_n(\vec{x}') \int_{\Omega} g_m(\vec{x}) G(\vec{x}, \vec{x}') d\Omega d\Omega', \end{aligned}$$

it follows that we can use exactly the same technique (equations (12) and (13)) for computing ρ_n^m of any testing function g_i by evaluating the corresponding Q2M coefficients and using the relation in equation (15).

By this process, we have thus decoupled the application and the library for computing the L2P and Q2M coefficients for the Laplace equation: the application does not have to work with I_n^m functions and the library does not have to know the nature of the basis and testing functions, (f_j, g_i) .

A.1. Comparison of performance

Given a multipole degree $p > 0$ and using a composite Gauss-Trapezoidal rule with $2(p+1)^2$ quadrature points, it is easy to see that the standard approach to computing Q2M (and L2P) coefficients and the potential based approach proposed above have the same asymptotic complexity. However, the proposed method has a larger constant associated with it. We now demonstrate that this additional cost for engineering a simplified interface is negligible in the context of the much larger initialization time (the time for the matrix-vectors are not affected).

To do this, we have taken 30 distinct triangular meshes ranging in size from 200 to about 1.2 million. They were generated by Ansys Q3D Extractor for different test cases and do not share a common geometry. For our purposes, it is sufficient to replace the triangles with point sources at their centroids. The canonical matrix in equation (4) is then constructed with a multipole degree of $p = 5$ which gives roughly 4 digits of accuracy in the matrix vector products. All the computations were done on a single core of a virtual machine with two Intel(R) Xeon(R) 6238R @2.20GHz processors with 182GB of RAM. In Fig. 3, we compare the time for evaluating the

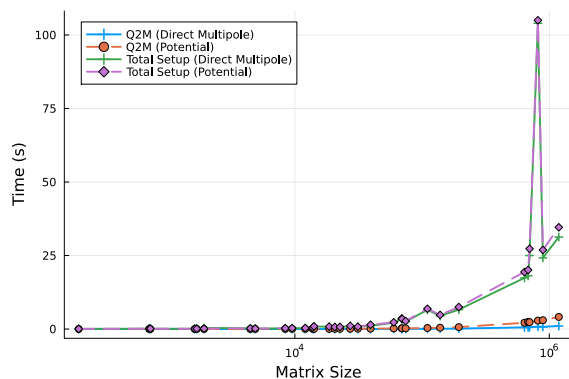


Fig. 3. Comparing the time for evaluating Q2M coefficients using standard (“Direct Multipole”) and using sampled potentials (“Potential”) for various test cases. We have also plotted the total setup time to provide the larger context.

Q2M coefficients using the standard approach and from the potential samples.

It is important to note that the bulk of the initialization/setup time is spent in computing the near matrix and that the near matrix evaluation for concentrated sources gives the least possible time. This is because the commonly used basis functions require more expensive handling of singularities and near singularities. In other words, we are comparing the cost of Q2M evaluations against the best case scenario for the rest of the initialization. It is clear from Fig. 3 that although the proposed method is slightly more expensive, the increase is negligible when compared with the overall setup time. We have also compared the accuracy of the computed matrix vector products and the largest 2–norm error observed was less than 2×10^{-4} , which is comparable to the overall precision of the matrix-vector products corresponding to the degree of multipole approximation $p = 5$.

B. Evaluation of Q2P and L2P coefficients for the Helmholtz equation

In the case of Helmholtz FMM that uses only multipole expansions involving \tilde{O}_n^m and \tilde{I}_n^m , it is clear that an approach analogous to the one described in section A can be readily developed. However, most applications of Helmholtz FMM involves electrically large structures where a diagonalized form is employed [2, 10], often combined with multipole expansions to handle multi-scale structures [15]. For such a mixed-form FMM, it is simpler to compute signature functions [10] rather than work with potentials. We now show how signature functions can be used to decouple the FMM library from the applications.

The *signature function* of the potential $\psi_j(\vec{x})$ of equation (10a), now with the Green’s function for the Helmholtz equation (1), is defined by formula [10]:

$$\tilde{\psi}_j(\hat{\mathbf{s}}) = \lim_{r \rightarrow \infty} k r e^{-ikr} \psi(\vec{c}_s + r\hat{\mathbf{s}}), \quad (16a)$$

which when approximated by equation (11a) with $\mathcal{O}_n^m = \tilde{O}_n^m$, can be shown to be:

$$\tilde{\psi}_j(\hat{\mathbf{s}}) = \sum_{n=0}^{\infty} \sum_{m=-n}^n \alpha_n^m \frac{Y_n^m(\hat{\mathbf{s}})}{i^{n+1}}. \quad (16b)$$

Therefore, given a signature function $\tilde{\psi}(\hat{\mathbf{s}})$, the orthogonality of spherical harmonics implies that we can recover the coefficients of the multipole expansion in equation (11a), α_n^m , using the formula [16]:

$$\alpha_n^m = i^{n+1} \int_{\Omega_0} \tilde{\psi}_j(\hat{\mathbf{s}}) Y_n^{m*}(\hat{\mathbf{s}}) d\Omega_s. \quad (17)$$

It is well known that [2, 10] the signature function of the Green’s function $G(\vec{r}, \vec{r}') = \frac{e^{ik|\vec{r}-\vec{r}'|}}{|\vec{r}-\vec{r}'|}$ for $\vec{r}' \in D_s$ is given by:

$$\tilde{G}(\hat{\mathbf{s}}) = k e^{ik\hat{\mathbf{s}} \cdot (\vec{c}_s - \vec{r}')}$$

Therefore, it follows that signature function of $\psi_j(\vec{x})$ of the canonical problem in equation (10a) is given by:

$$\tilde{\psi}_j(\hat{\mathbf{s}}) = k \int_{\text{supp } f_j} f_j(\vec{r}') e^{ik\hat{\mathbf{s}} \cdot (\vec{c}_s - \vec{r}')} d\vec{r}', \quad (18)$$

and we define the signature of the *basis function* f_j by $\tilde{f}_j(\hat{\mathbf{s}}) = \tilde{\psi}_j(\hat{\mathbf{s}})$.

Using the representation [9, 10]:

$$\tilde{I}_n^m(\vec{x} - \vec{c}) = \frac{1}{4\pi i^n} \int_{\Omega_0} e^{ik(\vec{x} - \vec{c}) \cdot \hat{\mathbf{s}}} Y_n^m(\hat{\mathbf{s}}) d\Omega_s, \quad (19a)$$

it is easy to show that the L2P coefficients:

$$\rho_n^m = \frac{1}{4\pi i^n} \int_{\Omega_0} \tilde{g}_i(\hat{\mathbf{s}}) Y_n^m(\hat{\mathbf{s}}) d\Omega_s \quad (19b)$$

$$\text{where } \tilde{g}_i(\hat{\mathbf{s}}) \stackrel{\text{def}}{=} \int_{\text{supp } g_i} g_i(\vec{x}) e^{ik(\vec{x} - \vec{c}) \cdot \hat{\mathbf{s}}} d\vec{x}, \quad (19c)$$

are defined by the signature function of the *testing function* g_i ².

Hence, in a mixed-form broadband FMM that uses both diagonalized and (possibly scaled) multipole expansions, the user only has to provide the signature functions in equation (18) and equation (19c) for setting up the far field calculations in Helmholtz FMM. Furthermore, these functions are easy to define and easy to calculate.

IV. COMPUTATION OF THE DERIVATIVES OF THE FAR FIELD

We claimed in the introduction that it is possible to represent the derivatives of far fields as product of sparse matrices. In this section, we shall demonstrate this claim. The approach relies on the following observation for the Helmholtz equation (A similar argument can be made for the Laplace equation by setting $k = 0$ and replacing \tilde{I}_n^m with I_n^m .)

Consider the potential field $\psi_j(\vec{x})$ due to sources in D_s having an approximation of the form in equation (11b):

$$\psi_j(\vec{x}) = \sum_{n=0}^N \sum_{m=-n}^n \beta_n^m \tilde{I}_n^m(\vec{x} - \vec{c}_r), \quad \vec{x} \in D_r. \quad (20a)$$

Since:

$$\nabla^2 \frac{\partial \psi_j}{\partial u} + k^2 \frac{\partial \psi_j}{\partial u} = \frac{\partial}{\partial u} \{ \nabla^2 \psi_j + k^2 \psi_j \} = 0, \quad u = x, y, z, \quad (20b)$$

it follows that $\frac{\partial \psi_j}{\partial u}$ also must have an expansion of the form:

$$\frac{\partial \psi_j(\vec{x})}{\partial u} = \sum_{n=0}^{N'} \sum_{m=-n}^n \delta_n^m \tilde{I}_n^m(\vec{x} - \vec{c}_r), \quad \vec{x} \in D_r, \quad (20c)$$

and that the coefficients $\{\beta_n^m\}$ and $\{\delta_n^m\}$ must be related through a linear operator T_u so that:

$$\psi_j \mapsto \frac{\partial \psi_j}{\partial u} \implies \{\beta_n^m\} \mapsto \{\delta_n^m\} = T_u(\{\beta_n^m\}).$$

²Compare with equation (18) for subtle differences.

Thus, for non-negative integers $p, q, r \geq 0$, an inductive reasoning shows that:

$$\psi_j \mapsto \frac{\partial^{p+q+r} \psi_j}{\partial x^p \partial y^q \partial z^r} \implies \{\beta_n^m\} \mapsto \{\delta_n^m\} = T_z^r T_y^q T_x^p(\{\beta_n^m\}).$$

Therefore:

$$\int_{\text{supp } g_i} g_i(\vec{x}) \frac{\partial^{p+q+r} \psi_j}{\partial x^p \partial y^q \partial z^r} d\vec{x} = \sum_{n=0}^{N'} \sum_{m=-n}^n \delta_n^m \cdot \int_{\text{supp } g_i} g_i(\vec{x}) \tilde{I}_n^m(\vec{x} - \vec{c}_r) d\vec{x} \quad (21a)$$

$$= \sum_{n=0}^{N'} \sum_{m=-n}^n \delta_n^m \rho_n^m, \quad (21b)$$

where ρ_n^m are the same receiving coefficients defined in equation (11d). Thus, we see that partial derivatives of arbitrary order manifest as product of linear operators that can be applied to the local expansion coefficients β_n^m transparent to the application/user.

So far, we have not said anything about the nature of T_u . In the next three sections, we shall demonstrate that the finite dimensional representation of T_u is a sparse matrix for the Laplace and low frequency Helmholtz FMM that use explicit multipole expansions and a diagonal matrix for the high frequency Helmholtz FMM that uses signature functions.

A. Derivatives for the Laplace kernel

Let $\vec{x} = (r, \theta, \phi) \in \mathbb{R}^3$. We begin with the integral representation of the inner function $I_n^m(\vec{x})$ derived in [10, equation 2.33]:

$$I_n^m(\vec{x}) = \frac{(-1)^{n+m}}{n!} \frac{1}{2\pi} \int_{-\pi}^{\pi} (z + ix \cos \alpha + iy \sin \alpha)^n e^{im\alpha} d\alpha. \quad (22)$$

Without the loss of generality, consider the case of $\frac{\partial I_n^m(\vec{x})}{\partial x}$. From Appendix A, we have:

$$\frac{\partial I_n^m(\vec{x})}{\partial x} = \frac{i}{2} (I_{n-1}^{m+1}(\vec{x}) + I_{n-1}^{m-1}(\vec{x})). \quad (23)$$

Substituting equation (23) into equation (20a), we get:

$$\begin{aligned} \frac{\partial \psi_j(\vec{x})}{\partial x} &= \sum_{n=0}^N \sum_{m=-n}^n \beta_n^m \frac{\partial I_n^m(\vec{x} - \vec{c}_r)}{\partial x} \\ &= \sum_{n=0}^N \sum_{m=-n}^n \beta_n^m \cdot \frac{i}{2} (I_{n-1}^{m+1}(\vec{x} - \vec{c}_r) + I_{n-1}^{m-1}(\vec{x} - \vec{c}_r)). \end{aligned} \quad (24a)$$

Since $I_n^m = 0$ if $n < 0$ or $|m| > n$, the above equation can be rewritten as:

$$\frac{\partial \psi_j(\vec{x})}{\partial x} = \sum_{n=0}^{N-1} \sum_{m=-n}^n \frac{i}{2} (\beta_{n+1}^{m-1} + \beta_{n+1}^{m+1}) I_n^m(\vec{x} - \vec{c}_r). \quad (24b)$$

Identifying:

$$\delta_n^m = \frac{i}{2}(\beta_{n+1}^{m-1} + \beta_{n+1}^{m+1}),$$

and $N' = N - 1$ in equation (20c), we see that the matrix representation of T_x has *at most two non-zero entries per row*. It is clear from Appendix A that a similar reasoning can be made for T_y and T_z as well.

B. Derivatives for the Helmholtz kernel with multipoles

Without the loss of generality, we consider the partial derivative with respect to x . Then, it can be shown that $\frac{\partial \tilde{\rho}_n^m}{\partial x}$ can be expressed as a four-term recurrence of the form (see Appendix B for the derivation and for the explicit expressions):

$$\frac{\partial \tilde{\rho}_n^m(\vec{x})}{\partial x} = \sum_{\substack{v=|n-1| \\ v \neq n}}^{n+1} \sum_{p=-1,1} T_v^{m+p} \tilde{\rho}_v^{m+p}(\vec{x}). \quad (25a)$$

A reasoning similar to that made in the previous section shows that each row of the matrix representation of T_x has at most four entries, once again showing that the derivative can be represented as a sparse matrix. Furthermore, a closer look at equation (25a) shows that if equation (20a) has degree N , then equation (20c) will have degree $N' = N + 1$. However, if the degree of the ‘‘L2P’’ coefficients ρ_n^m is truncated to N , then the summation in equation (21b) will be truncated to degree N and we can set $N' = N$.

C. Derivatives for the Helmholtz kernel in diagonalized form

A potential field $\psi_j(\vec{x})$ having an approximation of the form in (20a), can also be represented as [2, 10]:

$$\psi_j(\vec{x}) = \frac{i}{4\pi} \int_{\Omega_0} \tilde{\psi}_j(\hat{s}) \cdot e^{ik(\vec{x}-\vec{c}_r) \cdot \hat{s}} d\Omega_s, \quad (26a)$$

$$\text{where } \tilde{\psi}_j(\hat{s}) \stackrel{\text{def}}{=} \sum_{n=0}^N \sum_{m=-n}^n \beta_n^m \frac{Y_n^m(\hat{s})}{i^{n+1}}, \quad (26b)$$

so that:

$$\begin{aligned} P_{ij} &= \int_{\text{supp } g_i} g_i(\vec{x}) \psi_j(\vec{x}) d\vec{x} \\ &= \frac{i}{4\pi} \int_{\Omega_0} \tilde{g}_i(\hat{s}) \cdot \tilde{\psi}_j(\hat{s}) d\Omega_s, \end{aligned} \quad (26c)$$

$$\text{where } \tilde{g}_i(\hat{s}) = \int_{\text{supp } g_i} g_i(\vec{x}) e^{ik(\vec{x}-\vec{c}_r) \cdot \hat{s}} d\vec{x}, \quad (26d)$$

and we recognize $\tilde{g}_i(\hat{s})$ as the signature function defined in equation (19c). Since:

$$\frac{\partial \psi_j(\vec{x})}{\partial x} = \frac{i}{4\pi} \int_{\Omega_0} (iks_x) \cdot \tilde{\psi}_j(\hat{s}) \cdot e^{ik(\vec{x}-\vec{c}_r) \cdot \hat{s}} d\Omega_s, \quad (26e)$$

from equation (26a), it follows that:

$$\begin{aligned} &\int_{\text{supp } g_i} g_i(\vec{x}) \frac{\partial \psi_j(\vec{x})}{\partial x} d\vec{x} \\ &= \frac{i}{4\pi} \int_{\Omega_0} \tilde{g}_i(\hat{s}) \cdot (iks_x) \cdot \tilde{\psi}_j(\hat{s}) d\Omega_s, \end{aligned} \quad (26f)$$

demonstrating that the partial differential operator appears as a *diagonal* modification to the incoming signature function $\tilde{\psi}_j(\hat{s})$.

Note that this result has been known from the very early days of FMM and has been effectively used for computing with derivatives [17]. The novelty in our approach is that we are using it to transfer the *responsibility of computing the derivative* to the library instead of requiring the application do it.

V. A PSEUDO-EXAMPLE FOR A LIBRARY INTERFACE

We now discuss how the results from sections III and IV can be used to develop an application independent library of Laplace and Helmholtz FMM. Our objective is not to describe a concrete implementation, but to discuss the essential functionality.

To that end, we divide the library functionality into two parts: the matrix build/initialization functions and matrix apply (‘‘matrix-vector’’ product) functions. During the initialization, we expect the user to provide what are called ‘‘call-back’’ functions. The library, in turn, repeatedly calls these functions to get the information needed to complete matrix initialization. After the matrix is built, the library provides the functionality to evaluate matrix-vector products, potentially involving different combinations of derivatives.

A. Matrix initialization

Building the matrix consists of three phases:

- The tree construction phase
- The evaluation of the near interactions
- The evaluation of the far field representations (‘‘Q2M’’ and ‘‘L2P’’ coefficients).

A.1. Constructing the tree

Rationale:

In general, each source/receiver can be assigned a ‘‘center’’ and a spatial extent. The former can be captured as a array of three floating point numbers. Given that the mesh used in the discretization can be extremely non-uniform in practical cases, it is often necessary to take the spatial extent into account to ensure sufficient accuracy. This can lead to non-uniform trees with internal nodes also ‘‘hosting’’ sources and/or receivers. The following listing shows an example of an interface.

```
1 function get_center(particle_index, center []) ←
;
2 function get_bounding_box(particle_index, ←
xyzMin [], xyzMax []);
```

A.2. Computing near interactions with multiple matrices

Rationale:

With the ability to compute arbitrary derivatives of the far field, it becomes necessary to handle multiple

matrices as well. As a simple example, consider the following summations:

$$f(\vec{x}_i) = \sum_{j=1}^N a_j G(\vec{x}_i, \vec{x}_j), \quad (27a)$$

$$\vec{F}(\vec{x}_i) = \sum_{j=1}^N a_j \nabla G(\vec{x}_i, \vec{x}_j). \quad (27b)$$

Although potential and the derivatives of the far field can be computed from a single FMM representation, the four components of the near matrix must be stored simultaneously. In other words, the library needs to allow for the possibility of multiple matrix entries resulting from each pair-wise interaction. If we assume that every pair-wise near interaction results in the same number of matrix entries (although there are exceptions, this usually the case), the following interface would be adequate for most purposes.

```
function get_matrix_element(receiver_index, ←
    source_index, elements[]);
```

In the above example, $\dim(\text{elements})=4$ and a user can return the Green's function and its gradient as:

$$\text{elements} = \left[G, \frac{\partial G}{\partial x}, \frac{\partial G}{\partial y}, \frac{\partial G}{\partial z} \right].$$

A.3. Computing Q2M and L2P coefficients

Rationale:

In section III, we have shown that the required Q2M and L2P coefficients can be evaluated from sampled potential values or signature functions. For the Laplace equation, this can be implemented in the following way. Given a node with sources (or receivers) in it, we can construct a sphere with center, \vec{c}_s at the center of the box and a sufficiently large radius, $R \gg d$ where d is the box length, and introduce a Gauss-Trapezoidal rule as discussed in section A. Then, a function of the following nature can be invoked repeatedly to evaluate the potential, as defined by equation (10a) at each integration point $\vec{x}_k = \vec{c}_s + R\hat{s}_k$ in equation (13).

```
function P(xk) = get_potential(particle_index ←
    , xk[]);
```

For the Helmholtz equation, we expect the user to return the signature function, defined by equation (18).

```
function F(sk) = get_signature(particle_index ←
    , cs[], sk[]);
```

Note that in this case, we need both the center of the box \vec{c}_s and the direction vector \hat{s}_k passed to the application.

B. Evaluating matrix-vector products

In order to offer maximum flexibility, we split the evaluation of the matrix-vector products (MVP) into three steps:

- `multipole_pass`: performs all the source-to-multipole (Q2M), outer-to-outer (M2M, aggregation), outer-to-inner (M2L) and inner-to-inner (L2L) translations in the standard FMM flow. This is a pre-requisite for calling the far field evaluation step below.
- `far_axpy`: evaluates (the integral of) the potential (or its derivative) for every testing function using the results in Section IV.
- `near_axpy`: evaluates the contribution due the near interaction matrix selected by the user (if there are multiple matrices involved).

The library can offer a functionality similar to the following one to accomplish these tasks:

```
1 /* takes the input vector X and performs M2M, ←
   * M2L and L2L passes of FMM
2 * Must be called before calling far_axpy.
3 */
4 function multipole_pass(input_x);
5
6 /* near_matrix_index lets the user select the ←
   * correct near matrix */
7 function near_axpy(near_matrix_index, input_x ←
   , output_y);
8
9 /* the input array deriv tells the library to ←
   * apply the desired derivatives
10 * to the far field coefficients before ←
   * evaluating the potential */
11 function far_axpy(deriv[], output_y);
```

where the array `deriv` specifies the orders of the partial derivatives. For example:

$$\text{deriv}[1,2,2] \implies \frac{\partial^2}{\partial z^2} \frac{\partial^2}{\partial y^2} \frac{\partial}{\partial x}.$$

The motivation for this is best illustrated by an example. Towards this, consider the evaluation of the two sums in equation (27). A pseudo-code is given below.

```
1 /* Let X denote the input vector and P, dPx, ←
   * dPy and dPz, respectively denote the ←
   * potential and
2 * the three components of the gradient
3 */
4
5 /* do M2M, M2L and L2L */
6 multipole_pass(X);
7
8 /* compute P = A * X for potential; matrix ←
   * index 0 ==> Green's function */
9 near_axpy(0, X, P);
```



```

10  deriv = [0,0,0]; /* no partial derivatives ←
    */
11  far_axpy(deriv,P);
12
13  /* compute dPx = dAx * X for partial ←
    derivative with respect to x */
14  near_axpy(1, X,dPx); /* matrix index 1 ==>←
    partial derivative with respect to x ←
    */
15  deriv = [1, 0, 0]; /* partial derivative←
    with respect to x */
16  far_axpy(deriv,dPx);

```

Computation of the partial derivatives with respect to y and z can be done analogously. Note that the `multipole_pass` is usually the most dominant part during a matrix-vector evaluation and it is invoked only once.

VI. SUMMARY AND CONCLUSIONS

In this paper, we have proposed an approach to developing a stand-alone, application independent library of Laplace and Helmholtz FMM. Towards this end, we have demonstrated a technique to capture the essential characteristics of the problem needed for setting up the FMM – the basis and testing functions, mesh etc. – using samples of either potentials or the signature functions of a *canonical problem*. Furthermore, we have shown that it is possible to transfer the responsibility of computing the partial derivatives of the far field to the library, instead of burdening the user to implement complicated integrals. This is accomplished by representing partial derivatives of arbitrary orders as product of certain sparse matrices. We have also outlined a pseudo interface that illustrates how the methods can be used.

Within Ansys Inc., we have developed a flexible library, called `Q3Dfastmvpack` incorporating these ideas and it is currently being used by multiple Ansys products.

ACKNOWLEDGEMENTS

I would like to thank my colleagues Andy Mathis, Dalian Zheng, Eric Bracken and Indranil Chowdhury for the many discussions. I would also like to thank Werner Thiel for supporting this work.

APPENDICES

A. Partial derivatives of I_n^m

Let $h(\alpha; \vec{x}) = (z + ix \cos \alpha + iy \sin \alpha)$ in equation (22). Consider:

$$\frac{\partial I_n^m(\vec{x})}{\partial x} = \frac{(-1)^{n+m}}{(n-1)!} \frac{i}{2\pi} \int_{-\pi}^{\pi} h(\alpha; \vec{x})^{n-1} e^{im\alpha} \cos \alpha d\alpha$$

$$\begin{aligned} &= \frac{(-1)^{n+m}}{(n-1)!} \frac{i}{2\pi} \int_{-\pi}^{\pi} h(\alpha; \vec{x})^{n-1} e^{im\alpha} \frac{e^{i\alpha} + e^{-i\alpha}}{2} d\alpha \\ &= \frac{i}{2} \frac{(-1)^{n+m}}{(n-1)!} \frac{1}{2\pi} \left\{ \int_{-\pi}^{\pi} h(\alpha; \vec{x})^{n-1} e^{i(m+1)\alpha} d\alpha \right. \\ &\quad \left. + \int_{-\pi}^{\pi} h(\alpha; \vec{x})^{n-1} e^{i(m-1)\alpha} d\alpha \right\} \\ &= \frac{i}{2} \frac{(-1)^{n+m}}{(n-1)!} \frac{1}{2\pi} \int_{-\pi}^{\pi} h(\alpha; \vec{x})^{n-1} e^{i(m+1)\alpha} d\alpha \\ &\quad + \frac{i}{2} \frac{(-1)^{n+m}}{(n-1)!} \frac{1}{2\pi} \int_{-\pi}^{\pi} h(\alpha; \vec{x})^{n-1} e^{i(m-1)\alpha} d\alpha \\ &= \frac{i}{2} (I_{n-1}^{m+1}(\vec{x}) + I_{n-1}^{m-1}(\vec{x})), \end{aligned}$$

where we have used the facts $n-1+m-1 = n+m$ and that $(-1)^{n+m-2} = (-1)^{n+m}$. The following results can be similarly derived:

$$\frac{\partial I_n^m(\vec{x})}{\partial y} = \frac{1}{2} (I_{n-1}^{m+1}(\vec{x}) - I_{n-1}^{m-1}(\vec{x})),$$

$$\frac{\partial I_n^m(\vec{x})}{\partial z} = -I_{n-1}^m(\vec{x}).$$

B. Partial derivatives of $\tilde{I}_n^m(\vec{x})$

We have, from equation (19a):

$$\begin{aligned} &\frac{\partial^{p+q+r}}{\partial x^p \partial y^q \partial z^r} I_n^m(\vec{x}) \\ &= \frac{(ik)^{p+q+r}}{4\pi i^n} \int_{\Omega_0} s_x^p s_y^q s_z^r e^{ik\vec{x}\cdot\hat{s}} Y_n^m(\hat{s}) d\Omega_s, \end{aligned} \quad (28)$$

where $\hat{s} = (s_x, s_y, s_z) = (\sin \theta \cos \phi, \sin \theta \sin \phi, \cos \theta)$. Noting that:

$$\sin \theta e^{i\phi} = (-) \sqrt{\frac{8\pi}{3}} Y_1^1(\hat{s}), \quad (29)$$

from the definition of spherical harmonics $Y_n^m(\theta, \phi)$ in equation (5b), and using equation (5c), we get:

$$\sin \theta \cos \phi = (-) \sqrt{\frac{2\pi}{3}} [Y_1^1(\hat{s}) - Y_1^{-1}(\hat{s})] \quad (30a)$$

$$\sin \theta \sin \phi = i \sqrt{\frac{2\pi}{3}} [Y_1^1(\hat{s}) + Y_1^{-1}(\hat{s})] \quad (30b)$$

$$\cos \theta = \sqrt{\frac{4\pi}{3}} Y_1^0(\hat{s}). \quad (30c)$$

Let:

$$\mathcal{V}_p(\vec{x}) = \frac{1}{4\pi i^n} \int_{\Omega_0} Y_1^p(\hat{s}) e^{ik\vec{x}\cdot\hat{s}} Y_n^m(\hat{s}) d\Omega_s, p = -1, 0, 1. \quad (31)$$

Then, substituting equation (30) into equation (28) and using equation (31), we get:

$$\frac{\partial I_n^m(\vec{x})}{\partial x} = (-) ik \sqrt{\frac{2\pi}{3}} (\mathcal{V}_1(\vec{x}) - \mathcal{V}_{-1}(\vec{x})) \quad (32a)$$

$$\frac{\partial I_n^m(\vec{x})}{\partial y} = (-) k \sqrt{\frac{2\pi}{3}} (\mathcal{V}_1(\vec{x}) + \mathcal{V}_{-1}(\vec{x})) \quad (32b)$$

$$\frac{\partial I_n^m(\vec{x})}{\partial z} = ik \sqrt{\frac{4\pi}{3}} \mathcal{V}_0(\vec{x}). \quad (32c)$$

Now, substituting the following well known addition theorem [9, 10]:

$$e^{ik\vec{x}\cdot\hat{s}} = \sum_{n'=0}^{\infty} \sum_{m'=-n'}^{n'} 4\pi i^{n'} j_{n'}(kx) Y_{n'}^{m'*}(\hat{\mathbf{x}}) Y_{n'}^{m'}(\hat{s}),$$

into equation (31), we can express:

$$\mathcal{V}_p(\vec{x}) = \frac{1}{4\pi i^n} \sum_{v=0}^{\infty} \sum_{\mu=-v}^v 4\pi i^v (-1)^\mu j_v(k\|\vec{x}\|) Y_v^{-\mu}(\hat{\mathbf{x}}) \int_{\Omega_0} Y_1^p(\hat{s}) Y_v^\mu(\hat{s}) Y_n(\hat{s}) d\Omega_s \quad (33a)$$

$$= \sum_{v=0}^{\infty} \sum_{\mu=-v}^v i^{v-n} (-1)^\mu j_v(k\|\vec{x}\|) Y_v^{-\mu}(\hat{\mathbf{x}}) \gamma \begin{pmatrix} v & 1 & n \\ \mu & p & m \end{pmatrix}, \quad (33b)$$

where γ is the Gaunt coefficient defined as

$$\gamma \begin{pmatrix} n_1 & n_2 & n_3 \\ m_1 & m_2 & m_3 \end{pmatrix} \stackrel{\text{def}}{=} \int_{\Omega_0} Y_{n_1}^{m_1}(\hat{s}) Y_{n_2}^{m_2}(\hat{s}) Y_{n_3}^{m_3}(\hat{s}) d\Omega_s, \quad (33c)$$

for integers $n_i \geq 0$ and $-n_i \leq m_i \leq n_i$ for $i = 1, 2, 3$. Using the properties of Gaunt coefficients [9], for $\gamma \neq 0$, we have to have $|n-1| \leq v \leq n+1$, $v+n+1$ even and $\mu+m+p=0$. Thus, $v = |n-1|$ and $v = n+1$ and therefore:

$$\mathcal{V}_p(\vec{x}) = \sum_{\substack{v=|n-1| \\ v \neq n}}^{n+1} i^{v-n} (-1)^{m+p} j_v(k\|\vec{x}\|) Y_v^{m+p}(\hat{\mathbf{x}}) \gamma \begin{pmatrix} v & 1 & n \\ -(m+p) & p & m \end{pmatrix} \quad (33d)$$

$$= (-1)^{m+p} \sum_{\substack{v=|n-1| \\ v \neq n}}^{n+1} i^{v-n}.$$

$$\gamma \begin{pmatrix} v & 1 & n \\ -(m+p) & p & m \end{pmatrix} I_v^{m+p}(\vec{x}), \quad (33e)$$

showing that the right hand sides of equations (32a)-(32c) have at most four non-zero terms.

REFERENCES

- [1] L. Greengard, "The rapid evaluation of potential fields in particle systems," Ph.D. thesis, Yale University, 1987.
- [2] V. Rokhlin, "Diagonal forms of translation operators for the Helmholtz equation in three dimensions," *Applied and Comp. Harmonic Analysis*, vol. 1, pp. 82-93, 1993.
- [3] W. C. Chew, J.-M. Jin, E. Michielssen, and J. M. Song, *Fast and Efficient Algorithms in Computational Electromagnetics*, chap. 9, Boston: Artech House, 2001.
- [4] V. Rokhlin, "Rapid solution of integral equations of classical potential theory," *Journal Of Computational Physics*, vol. 60, pp. 187-207, 1985.
- [5] J. Carrier, L. Greengard, and V. Rokhlin, "A fast adaptive multipole algorithm for particle simulations," *SIAM Journal on Scientific and Statistical Computing*, vol. 9, no. 4, pp. 669-686, July 1988.
- [6] V. Rokhlin, "Rapid solution of integral equations of scattering theory in two dimensions," *Journal Of Computational Physics*, vol. 86, pp. 414-439, 1990.
- [7] W.-J. He, X.-W. Huang, M.-L. Yang, and X.-Q. Sheng, "Massively parallel multilevel fast multipole algorithm for extremely large-scale electromagnetic simulations: A review," *Progress In Electromagnetics Research*, vol. 173, pp. 37-52, 2022.
- [8] F. W. J. Olver, D. W. Lozier, R. F. Boisvert, and C. W. Clark, *NIST Handbook of Mathematical Functions*, Cambridge: Cambridge University Press, Cambridge, 2010.
- [9] A. Messiah, *Quantum Mechanics*, New York: Dover Publications, Inc., 2014.
- [10] M. A. Epton and B. Dembart, "Multipole translation theory for the three dimensional Laplace and Helmholtz equations," *SIAM J. of Scientific Computing*, vol. 16, no. 4, pp. 865-897, July 1995.
- [11] C. R. Anderson, "An implementation of the fast multipole method without multipoles," *SIAM Journal of Scientific and Statistical Computing*, vol. 13, no. 4, pp. 923-947, July 1992.
- [12] T. Eibert, "A diagonalized multilevel fast multipole method with spherical harmonics expansion of the k-space Integrals," *IEEE Transactions on Antennas and Propagation*, vol. 53, no. 2, pp. 814-817, Feb. 2005.
- [13] B. Dembart and E. Yip, "A 3D Fast multipole method for electromagnetics with multiple levels," *11th Annual Review of Progress in Applied Computational Electromagnetics, Monterey, CA*, vol. 1, pp. 621-628, Mar. 1995.
- [14] J. Phillips and J. White, "A precorrected-FFT method for electrostatic analysis of complicated 3-D structures," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 16, no. 10, pp. 1059-1072, 1997.
- [15] J.-S. Zhao and W. C. Chew, "Integral equation solution of Maxwell's equations from zero frequency to microwave frequencies," *IEEE Transactions on Antennas and Propagation*, vol. 48, no. 10, pp. 1635-1645, Oct. 2000.
- [16] B. Dembart and E. Yip, "The accuracy of fast multipole methods for Maxwell's equations," *IEEE Computational Science and Engineering*, vol. 5, no. 3, pp. 48-56, 1998.
- [17] J. M. Song and W. C. Chew, "Multilevel fast multipole algorithm for solving combined field integral equations of electromagnetic scattering," *Micro. Opt. Tech. Lett.*, vol. 10, no. 1, pp. 14-19, Sep. 1995.



Sanjay Velamparambil I obtained a Ph.D. in Engineering from the Indian Institute of Science, Bangalore, in 1997. My doctoral work was in the then nascent field of fast multipole methods (FMM). Subsequently, working with Dr. Weng Chew at the University of Illinois at Urbana-

Champaign, I pioneered the parallelization of Helmholtz FMM on distributed memory computers.

I was with Ansoft Corporation (currently Ansys Inc.) from 2002-2005 working on fast integral equation methods for signal integrity applications. In 2007, I joined Acceleware Corporation, Calgary, and worked on linear algebraic algorithms for GPU computing. In 2010, I joined Apache Design Solutions, which was later acquired by Ansys. I have been working on fast integral equation methods for several Ansys products such as Ansys Q3D Extractor[®] and SIwave[™] since 2010.

I enjoy long bicycle rides, especially mountainous routes, running, and generally love the outdoors. I am married and we are fortunate enough to live in the foothills of the majestic Rocky Mountains in Northern Colorado.