SIAM J. Sci. Comput.
Vol. 0, No. 0, pp. 000–000

© 2012 Society for Industrial and Applied Mathematics

FOURIER-BASED FAST MULTIPOLE METHOD FOR THE HELMHOLTZ EQUATION*

CRIS CECKA† AND ERIC DARVE‡

Abstract. The fast multipole method (FMM) has had great success in reducing the computational complexity of solving the boundary integral form of the Helmholtz equation. We present a formulation of the Helmholtz FMM that uses Fourier basis functions rather than spherical harmonics. By modifying the transfer function in the precomputation stage of the FMM, time-critical stages of the algorithm are accelerated by causing the interpolation operators to become straightforward applications of fast Fourier transforms, retaining the diagonality of the transfer function, and providing a simplified error analysis. Using Fourier analysis, constructive algorithms are derived to a priori determine an integration quadrature for a given error tolerance. Sharp error bounds are derived and verified numerically. Various optimizations are considered to reduce the number of quadrature points and reduce the cost of computing the transfer function.

Key words. fast multipole method, fast Fourier transform, Fourier basis, interpolation, anterpolation, Helmholtz, Maxwell, integral equations, boundary element method

AMS subject classifications. 31B10, 42B10, 65D05, 65R20, 65Y20, 65T40, 70F10, 78M15, 78M16

DOI. 10.1137/11085774X

1. Introduction.

Since the development of the fast multipole method (FMM) for the wave equation in [21, 6, 22, 11, 20], the FMM has proven to be a very effective tool for solving scalar acoustic and vector electromagnetic problems. In this paper, we consider the application of the FMM to the scalar Helmholtz equation, although our results can be immediately extended to the vector case as described in [4, 8]. The application of the boundary element method to solve the integral form of the Helmholtz equation results in a dense linear system which can be solved by iterative methods such as GMRES or BCGSTAB. These methods require computing dense matrix-vector products which, using a direct implementation, are performed in $\mathcal{O}(N^2)$ floating-point operations. The FMM uses an approximation of the dense matrix to perform the product in $\mathcal{O}(N \log N)$ operations. This approximation is constructed from close-pair interactions and far-field approximations represented by spherical integrals that are accumulated and distributed through the domain via an octree.

There are a number of difficulties in implementing the FMM, each of which must be carefully considered and optimized to achieve the improved complexity. The most significant complication in the Helmholtz FMM is that the quadrature sampling rate must increase with the size of the box in the octree, requiring interpolation and anterpolation algorithms to transform the data between spherical quadratures of different

*Submitted to the journal's Methods and Algorithms for Scientific Computing section December 5, 2011; accepted for publication (in revised form) September 6, 2012; published electronically DATE. This research was supported by the U.S. Army Research Laboratory, through the Army High Performance Computing Research Center, Cooperative Agreement W911NF-07-0027, the Stanford School of Engineering, and the King Abdullah University of Science and Technology.

http://www.siam.org/journals/sisc/x-x/85774.html

†Institute for Computational and Mathematical Engineering, Stanford University, Stanford, CA 94305 (ccecka@seas.harvard.edu).

‡Institute for Computational and Mathematical Engineering, Mechanical Engineering Department, Stanford University, Stanford, CA 94305 (darve@stanford.edu).

levels of the tree. Local algorithms such as Lagrange interpolation and techniques which sparsify interpolant matrices are fast, but incur significant errors [17, 8]. Spherical harmonic transforms are global interpolation schemes and are exact but require fast versions for efficiency of the FMM. Many of these fast spherical transform algorithms are only approximate, complicated to implement or use, and not always stable [9, 14, 26].

In this paper, we use a multipole expansion which allows the use of two-dimensional fast Fourier transforms (FFT) in the spherical coordinate system (ϕ, θ). The main advantages are twofold: (i) high performance FFT libraries are available on practically all computer platforms, resulting in accurate, robust, and fast interpolation algorithms; (ii) the resulting error analysis is simplified and leads to sharp, a priori error bounds on the FMM. One of the difficulties in using FFTs is that we are forced to use a uniform distribution of points along ϕ and θ in the spherical quadrature. Naively, this leads to a much increased quadrature size for a given accuracy compared to the original spherical harmonics-based FMM. The reason is as follows. The multipole expansion in the high frequency regime is derived from

$$\frac{e^{\imath\kappa|\boldsymbol{r}+\boldsymbol{r}_0|}}{|\boldsymbol{r}+\boldsymbol{r}_0|} = \int_{\phi=0}^{2\pi}\int_{\theta=0}^{\pi} e^{\imath\kappa\hat{\boldsymbol{s}}\cdot\boldsymbol{r}}\, T_{\ell,\boldsymbol{r}_0}(\hat{\boldsymbol{s}})\, \sin(\theta)\,\mathrm{d}\theta\,\mathrm{d}\phi + \varepsilon_G,$$

where $\hat{\boldsymbol{s}} = [\cos(\phi)\sin(\theta), \sin(\phi)\sin(\theta), \cos(\theta)]$ is the spherical unit vector, $T_{\ell,\boldsymbol{r}_0}$ is the *transfer function* to be defined in (2.3), and ε_G is an error term. It is apparent that we are integrating along θ a function which has period 2π. However, the bounds of the integral are 0 to π, over which the function has a discontinuity in its derivative. This implies a slow decay of the Fourier spectrum (essentially $1/\text{freq}^2$) of the integrand. Consequently, using a uniform quadrature in θ and ϕ to numerically compute integrals of this form would require a large number of quadrature points along θ.

We propose to use a variant of the scheme by Sarvas in [24] whereby the integration is extended from 0 to 2π and the integrand modified:

$$\frac{e^{\imath\kappa|\boldsymbol{r}+\boldsymbol{r}_0|}}{|\boldsymbol{r}+\boldsymbol{r}_0|} = \frac{1}{2}\int_{\phi=0}^{2\pi}\int_{\theta=0}^{2\pi} e^{\imath\kappa\hat{\boldsymbol{s}}\cdot\boldsymbol{r}}\, T_{\ell,\boldsymbol{r}_0}(\hat{\boldsymbol{s}})\, |\sin(\theta)|\,\mathrm{d}\theta\,\mathrm{d}\phi + \varepsilon_G. \tag{1.1}$$

We will describe in more details how an efficient scheme can be derived from this equation. The key property is that $e^{\imath\kappa\hat{\boldsymbol{s}}\cdot\boldsymbol{r}}$ is approximately bandlimited in θ, and therefore it is possible to remove the high frequency components of $T_{\ell,\boldsymbol{r}_0}(\hat{\boldsymbol{s}})\,|\sin(\theta)|$ without affecting the accuracy of the approximation. Using this smooth transfer function, which is now bandlimited in Fourier space, the number of quadrature points can be reduced dramatically. We show that the resulting number of quadrature points is reduced by about 40% compared to the original spherical harmonics-based FMM. Consequently, we now have a scheme which requires few quadrature points and enables the use of efficient FFT routines.

The approach in [24] is similar. However, rather than smoothing $T_{\ell,\boldsymbol{r}_0}(\hat{\boldsymbol{s}})\,|\sin(\theta)|$ once during the precomputation phase as we detail in this paper, Sarvas instead incorporates the $|\sin(\theta)|$ factor during the run-time phase of the FMM after the application of the transfer function. Although a detailed analysis is required for accurately assessing the relative cost of the two approaches, the technique in [24] requires approximately 1.5 times more sample points in the time-critical transfer pass of the algorithm, and requires an extra anterpolation step after applying T with about twice more sample points than used for the method in this paper. We also note that the error analysis for the two methods is different and is easier to carry out with the approach in this paper.

We derive a new a priori error analysis which incorporates both effects from truncation of the Gegenbauer series (a problem well analyzed [4]) and the numerical quadrature. Our algorithm to predict the error is very sharp. The sharp bounds allow the method to choose a minimal number of quadrature points to guarantee a prescribed error. By comparison, the conventional approach leads to less accurate estimates, resulting in either lower accuracy than requested or higher computational cost (overestimation of the required approximation order). Although not considered in this paper, our error analysis approach can also be applied to the spherical harmonics-based FMM to yield similarly accurate error bounds. This has practical importance since it allows guaranteeing the error in the calculation while reducing the computational cost.

The novel contributions of this paper can be summarized as follows:

- Development of an efficient Helmholtz multilevel FMM which uses FFTs in the interpolation/anterpolation steps while retaining diagonal transfer and translation functions. The use of FFTs allows leveraging high-performance FFT libraries available for most machines, sequential and parallel.
- An error analysis that accounts for all error in the method and yields constructive algorithms to choose optimal method parameters.
- Details of various optimizations to reduce the computational cost (e.g., use of symmetries in the precomputation of the transfer functions, use of symmetries on the unit sphere for the interpolation/anterpolation steps, optimization of the quadrature points near the poles of the unit sphere).
- Pseudocodes that clarify the method and help with an implementation by the reader.
- Demonstration of the sharpness of the error bound and the asymptotic computational cost.

The paper is organized as follows. In section 2, we introduce the critical parts of the classical Helmholtz FMM including the Gegenbauer series truncation (2.1), the spherical quadrature (2.2), and a short overview of interpolation/anterpolation strategies (2.3). Section 3 details the Fourier basis approach. The transfer function must be modified to lower the computational cost and obtain a competitive scheme, as detailed in section 3.2. Section 3.3 analyzes the integration error to derive an algorithm which determines a quadrature with a prescribed error tolerance. The FFT-based interpolation and anterpolation algorithms are described in section 3.4 and numerical results are given in section 3.5. Table 1 lists the notation used in this paper.

2. The multilevel FMM. The FMM reduces the computational complexity of the matrix-vector multiplication

$$\sigma_i = \sum_{j \neq i} \frac{e^{\imath\kappa|\boldsymbol{x}_i - \boldsymbol{x}_j|}}{|\boldsymbol{x}_i - \boldsymbol{x}_j|}\psi_j \tag{2.1}$$

for $i, j = 1, \ldots, N$ from $\mathcal{O}(N^2)$ to $\mathcal{O}(N \log N)$. This improvement is based on the addition theorem for spherical harmonic functions (Gegenbauer series),

$$\frac{e^{\imath\kappa|\boldsymbol{r}+\boldsymbol{r}_0|}}{|\boldsymbol{r}+\boldsymbol{r}_0|} = \imath\kappa \sum_{n=0}^{\infty} (-1)^n (2n+1) h_n^{(1)}(\kappa\,|\boldsymbol{r}_0|) j_n(\kappa\,|\boldsymbol{r}|) P_n(\hat{\boldsymbol{r}} \cdot \hat{\boldsymbol{r}}_0), \tag{2.2}$$

Table 1
Table of notation.

Notation	Description
κ	wavenumber in $\mathbb{R}$, $2\pi/\lambda$ with wavelength λ
θ	polar angle
ϕ	azimuthal angle
$\boldsymbol{x}$	vector in $\mathbb{R}^3$, $\boldsymbol{x} = \lvert\boldsymbol{x}\rvert\,\hat{\boldsymbol{x}}$
S^2	sphere, $\{\hat{\boldsymbol{s}} \in \mathbb{R}^3 \,:\, \lvert\hat{\boldsymbol{s}}\rvert = 1\}$
a_l	box size at level l. Root: $l = 0$. Highest active level: $l = 2$
j_n	spherical Bessel function of the first kind
y_n	spherical Bessel function of the second kind
$h_n^{(1)}$	spherical Hankel function of the first kind
P_n	Legendre polynomial
$\mathcal{F}_n^x[f]$	nth coefficient of f's Fourier series in x, $f(x) = \sum_n \mathcal{F}_n^x[f]\, e^{\imath n x}$

found in 10.1.45 and 10.1.46 of [1]. The series converges absolutely and uniformly for $|\boldsymbol{r}_0| > |\boldsymbol{r}|$ and has been studied extensively in [3, 7].

Truncating the Gegenbauer series at ℓ and using an integral over the unit sphere, S^2, we obtain

$$\frac{e^{\imath\kappa|\boldsymbol{r}+\boldsymbol{r}_0|}}{|\boldsymbol{r}+\boldsymbol{r}_0|} = \int_{S^2} e^{\imath\kappa\hat{\boldsymbol{s}}\cdot\boldsymbol{r}}\, T_{\ell,\boldsymbol{r}_0}(\hat{\boldsymbol{s}})\,\mathrm{d}S(\hat{\boldsymbol{s}}) + \varepsilon_G,$$

where ε_G is the Gegenbauer series truncation error ($\varepsilon_G \to 0$ as $\ell \to \infty$) and the transfer function, $T_{\ell,\boldsymbol{r}_0}(\hat{\boldsymbol{s}})$, is defined as

$$T_{\ell,\boldsymbol{r}_0}(\hat{\boldsymbol{s}}) = \frac{\imath\kappa}{4\pi}\sum_{n=0}^{\ell} \imath^n(2n+1)h_n^{(1)}(\kappa\,|\boldsymbol{r}_0|)P_n(\hat{\boldsymbol{s}}\cdot\hat{\boldsymbol{r}}_0). \tag{2.3}$$

The reduced computational complexity of the FMM is achieved by constructing a tree of nodes, typically an octree, over the domain of the source and field points. We recall the main steps of the FMM to set some notation. Let $M_\alpha^l(\hat{\boldsymbol{s}})$ be the outgoing field for B_α^l, the box α of the tree in level $l \in [0, L]$ with center $\boldsymbol{c}_\alpha^l$.

Initialization (P2M): The method is initialized by computing the outgoing plane-wave expansions for each cluster contained in a leaf of the tree:

$$M_\alpha^L(\hat{\boldsymbol{s}}) = \sum_{i,\, \boldsymbol{x}_i \in B_\alpha^L} \psi_i\, e^{\imath\kappa\hat{\boldsymbol{s}}\cdot(\boldsymbol{x}_i - \boldsymbol{c}_\alpha^L)}.$$

Upward pass (M2M): These outgoing expansions are then aggregated upward through the tree by accumulating the product of the child cluster expansions with the plane-wave translation function:

$$M_\alpha^{l-1}(\hat{\boldsymbol{s}}) = \sum_{\beta,\, B_\beta^l \subset B_\alpha^{l-1}} M_\beta^l(\hat{\boldsymbol{s}})\, e^{\imath\kappa\hat{\boldsymbol{s}}\cdot(\boldsymbol{c}_\beta^l - \boldsymbol{c}_\alpha^{l-1})}, \qquad l = L,\, L-1,\, \ldots,\, 3.$$

Transfer pass (M2L): Incoming expansions, $I_\alpha^l(\hat{\boldsymbol{s}})$ of box B_α^l, are computed from the outgoing by multiplication with the transfer function:

$$I_\alpha^l(\hat{\boldsymbol{s}}) = \sum_{\beta \in \mathcal{I}(B_\alpha^l)} M_\beta^l(\hat{\boldsymbol{s}})\, T_{\ell,\boldsymbol{c}_\beta^l - \boldsymbol{c}_\alpha^l}(\hat{\boldsymbol{s}}), \qquad l = L,\, L-1,\, \ldots,\, 2,$$

where $\mathcal{I}(B_\alpha^l)$ is the interaction list of box B_α^l, defined as all boxes of level l which are not neighbors of B_α^l, but whose parent is a neighbor of the parent of B_α^l.

Downward pass (L2L): The incoming expansions are then disaggregated downward through the tree to compute the local field $L_\alpha^l(\hat{\boldsymbol{s}})$ of box B_α^l:

$$L_\alpha^{l+1}(\hat{\boldsymbol{s}}) = L_\beta^l(\hat{\boldsymbol{s}})\, e^{\imath\kappa\hat{\boldsymbol{s}}\cdot(\boldsymbol{c}_\beta^l - \boldsymbol{c}_\alpha^{l+1})} + I_\alpha^{l+1}(\hat{\boldsymbol{s}}), \qquad l = 2,\, 3,\, \dots,\, L-1,$$

where $B_\alpha^{l+1} \subset B_\beta^l$.

Field computation (L2P and P2P): At the finest level, the integration over the sphere is finally performed and added to the near-field contribution to determine the field value at the N field points:

$$\sigma_i = \int_{S^2} L_\alpha^L(\hat{\boldsymbol{s}})\, e^{\imath\kappa\hat{\boldsymbol{s}}\cdot(\boldsymbol{c}_\alpha^L - \boldsymbol{x}_i)}\, \mathrm{d}S(\hat{\boldsymbol{s}}) + \sum_{\substack{j,\, j\neq i,\\ \boldsymbol{x}_j\in\mathcal{N}(B_\alpha^L)}} \frac{e^{\imath\kappa|\boldsymbol{x}_i - \boldsymbol{x}_j|}}{|\boldsymbol{x}_i - \boldsymbol{x}_j|}\psi_j, \tag{2.4}$$

where $\boldsymbol{x}_i \in B_\alpha^L$ and $\mathcal{N}(B_\alpha^L)$ is the neighbor list of B_α^L, defined as B_α^L and all neighbor boxes of B_α^L.

2.1. Truncation parameter in the FMM.

The truncation parameter ℓ must be chosen so that the Gegenbauer series (2.2) is converged to a desired accuracy. However, for $n > x$, $j_n(x)$ decreases superexponentially while $h_n^{(1)}(x)$ diverges. The divergence of the Hankel function causes the transfer function to oscillate wildly and become numerically unstable. Even though the expansion converges, roundoff errors will adversely affect the accuracy if ℓ is too large. Thus, while one must choose $\ell > \kappa\,|\boldsymbol{r}|$ so that sufficient convergence of the Gegenbauer series is achieved, it must also be small enough to avoid the divergence of the transfer function. The selection of the truncation parameter ℓ has been studied extensively and a number of procedures for selecting it have been proposed [6, 7].

The excess bandwidth formula (EBF) is derived from the convergence of the plane-wave spectrum as presented in [4]. The EBF chooses ℓ as

$$\ell \approx \kappa\,|\boldsymbol{r}| + C(\kappa\,|\boldsymbol{r}|)^{1/3}. \tag{2.5}$$

An empirically determined common choice is $C = 1.8(d_0)^{2/3}$, where d_0 is the desired number of digits of accuracy. The EBF is one of the most popular choices to select the truncation parameter [13].

The actual Gegenbauer truncation error for a given ℓ can also be approximated. As Carayol and Collino showed in [3], an approximate upper bound of this error for large values of $|\boldsymbol{r}|$ is obtained when $P_n(\hat{\boldsymbol{r}}\cdot\hat{\boldsymbol{r}}_0) = (\pm 1)^n$ so that

$$|\varepsilon_G| \lesssim \kappa \left| \sum_{n=\ell+1}^{\infty} (\mp 1)^n (2n+1) h_n^{(1)}(\kappa\,|\boldsymbol{r}_0|) j_n(\kappa\,|\boldsymbol{r}|) \right|,$$

which they showed can be computed in closed form:

$$= \kappa^2 \frac{|\boldsymbol{r}|\,|\boldsymbol{r}_0|}{|\boldsymbol{r}_0| \pm |\boldsymbol{r}|} \left| h_{\ell+1}^{(1)}(\kappa\,|\boldsymbol{r}_0|) j_\ell(\kappa\,|\boldsymbol{r}|) \pm h_\ell^{(1)}(\kappa\,|\boldsymbol{r}_0|) j_{\ell+1}(\kappa\,|\boldsymbol{r}|) \right|. \tag{2.6}$$

This simplifying assumption is not valid for small $\kappa\,|\boldsymbol{r}|$ when the upper bound is obtained instead by choosing $\hat{\boldsymbol{r}}\cdot\hat{\boldsymbol{r}}_0$ such that the oscillation of $P_n(\hat{\boldsymbol{r}}\cdot\hat{\boldsymbol{r}}_0)$ compensates for the oscillation of $(-1)^n h_n^{(1)}(\kappa |\boldsymbol{r}_0|) j_n(\kappa |\boldsymbol{r}|)$. Using the EBF as an initial guess for ℓ and refining the choice using the above closed form when $|\boldsymbol{r}|$ is sufficiently large is a simple algorithm which yields a nearly optimal value for ℓ (that is the smallest value consistent with the target error). This is the scheme we selected for this paper.

Carayol and Collino in [2] and [3] present an in-depth analysis of the Jacobi–Anger series and the Gegenbauer series. They find the asymptotic formula

$$\ell \approx \kappa |\boldsymbol{r}| - \frac{1}{2} + \left(\frac{1}{2}\right)^{5/3} W^{2/3}\left(\frac{\kappa |\boldsymbol{r}|}{4\varepsilon^6}\left(\frac{1 + |\boldsymbol{r}_0| / |\boldsymbol{r}|}{1 - |\boldsymbol{r}_0| / |\boldsymbol{r}|}\right)^{3/2}\right),$$

where $W(x)$ is the Lambert function defined as the solution to

$$W(x)e^{W(x)} = x, \qquad x > 0.$$

This appears to be near optimal for large box sizes.

The errors introduced by the Gegenbauer series truncation have been investigated in other papers, including [17, 3, 7].

2.2. Spherical quadrature in the FMM. The error analysis is simplified if a scheme is used which exactly integrates spherical harmonics, Y_n^m, up to some degree. The most common choice of quadrature uses uniform sample points in ϕ and Gauss–Legendre sample points in $z(\theta)$. With $N+1$ uniform points in the ϕ-direction and $\frac{N+1}{2}$ Gauss–Legendre points in the θ-direction, all Y_n^m, $-n \le m \le n$, $0 \le n \le N$, are integrated exactly [8, 17].

2.3. Interpolation and anterpolation in the FMM. Since the quadrature performs an L^2 inner product, the required sampling rate depends on the spectral content of the translation operator, $e^{\imath\kappa \hat{\boldsymbol{s}}\cdot\boldsymbol{r}}$, and the transfer operator, $T_{\ell,\boldsymbol{r}_0}(\hat{\boldsymbol{s}})$. The transfer operator has a bandwidth of ℓ, and the translation operator's coefficients in a spherical harmonic expansion decreases superexponentially roughly for $n \gtrsim \kappa |\boldsymbol{r}|$. Since $\kappa |\boldsymbol{r}| < \ell$ and the spherical harmonic coefficients of the transfer function do not decay, the modes at least up to $\kappa |\boldsymbol{r}|$ must be resolved. Therefore, as fields are aggregated in the upward pass (M2M) and $|\boldsymbol{r}|$ becomes larger, a larger quadrature is required to resolve the higher modes.

Similarly, as fields are disaggregated in the downward pass (L2L), $|\boldsymbol{r}|$ becomes smaller and the higher modes of the incoming field make vanishingly small contributions to the integral as a consequence of Parseval's theorem for the L^2 inner product. Thus, as the incoming field is disaggregated down the tree, a smaller quadrature can be used to resolve it. This makes the integration faster and is actually required to achieve an optimal asymptotic running time. See Table 2.

There have been several approaches to performing the interpolation and anterpolation between levels in the FMM. Below, we enumerate a number of options that have previously been studied.

General, local interpolation methods like Lagrange interpolation, Gaussian interpolation, and B-splines are fast and provide for simple error analysis [17].

A spherical harmonic transform maps function values f_k, sampled at (ϕ_k, θ_k), to a new quadrature $(\phi'_{k'}, \theta'_{k'})$, via the linear transformation

$$f_{k'} = \sum_{m,l} Y_l^m(\phi'_{k'}, \theta'_{k'}) \sum_{k \le K} \omega_k \overline{Y_l^m(\phi_k, \theta_k)} f_k = \sum_{k \le K} A_{k'k} f_k. \tag{2.7}$$

This results in an $\mathcal{O}(N \log N)$ or $\mathcal{O}(N^{3/2})$ FMM (see Table 2). Fast spherical transforms (FSTs) have been developed in [9, 14, 26, 23] and applied to the FMM in [5].

TABLE 2

Computational complexity of the Helmholtz FMM in the big-$\mathcal{O}$ sense. Columns 2 *and* 3 *refer to the distribution of particles. We let $N = \mathcal{O}((\kappa a_0)^3)$ for a volume of scatterers and $N = \mathcal{O}((\kappa a_0)^2)$ for a surface of scatterers.* Direct *interpolation refers to a computation of the operator with no acceleration (i.e., several matrix-vector products).* Fast global *refers to fast spherical harmonic transforms or FFTs (this work).* Local *interpolations are based on stencils such as Lagrange interpolation.*

Interpolation	Volume	Surface
Direct	$N \log N$	$N^{3/2}$
Fast global	N	$N \log^2 N$
Local	N	$N \log N$

Using the FST reduces the interpolation and anterpolation procedures to $\mathcal{O}(K \log K)$ [26] (where K is the total number of quadrature points on the unit sphere), which results in an $\mathcal{O}(N)$ or $\mathcal{O}(N \log^2 N)$ FMM. However, the accuracy and stability of these algorithms remain in question.

Approximations of the spherical transform have also been investigated in [15, 8]. The interpolation matrix $A_{k'k}$ in (2.7) can be sparsified in a number of ways to provide an interpolation/anterpolation method that scales as $\mathcal{O}(K)$ with controllable relative error. Many other interpolation schemes exist with varying running times and errors. Rokhlin presents a fast polynomial interpolator based on the FMM in [10]. See also [16].

The asymptotic computational complexity in the big-$\mathcal{O}$ sense is summarized in Table 2.

3. Fourier-based multilevel FMM. The Fourier-based FMM is based on the identity

$$\int_{S^2} e^{\imath\kappa \hat{\boldsymbol{s}}\cdot\boldsymbol{r}}\, T_{\ell,\boldsymbol{r}_0}(\hat{\boldsymbol{s}})\, \mathrm{d}S(\hat{\boldsymbol{s}}) = \int_0^{2\pi}\int_0^{2\pi} E_{\boldsymbol{r}}(\theta,\phi)\, T^{\mathfrak{s}}_{\ell,\boldsymbol{r}_0}(\theta,\phi)\, \mathrm{d}\phi\, \mathrm{d}\theta, \tag{3.1}$$

where the translation function $E_{\boldsymbol{r}}(\theta,\phi)$ and the modified transfer function $T^{\mathfrak{s}}_{\ell,\boldsymbol{r}_0}(\theta,\phi)$ are defined as

$$E_{\boldsymbol{r}}(\theta,\phi) = e^{\imath\kappa \hat{\boldsymbol{s}}\cdot\boldsymbol{r}}, \qquad T^{\mathfrak{s}}_{\ell,\boldsymbol{r}_0}(\theta,\phi) = \frac{1}{2}\, T_{\ell,\boldsymbol{r}_0}(\hat{\boldsymbol{s}})\, |\sin(\theta)|\,, \tag{3.2}$$

where $\hat{\boldsymbol{s}} = [\cos(\phi)\sin(\theta),\ \sin(\phi)\sin(\theta),\ \cos(\theta)]$ and $T_{\ell,\boldsymbol{r}_0}(\hat{\boldsymbol{s}})$ is the transfer function defined in (2.3). The $\iint_0^{2\pi}$ representation suggests the use of the Fourier functions $\{e^{\imath n\phi}e^{\imath m\theta}\}$, which form an orthonormal basis of $L^2([0,2\pi] \times [0,2\pi])$. The idea of doubling the sphere to use Fourier methods appears in [24, 25].

Using a Fourier basis rather than a spherical harmonics basis allows (i) using two-dimensional uniform quadratures; (ii) FFTs in the interpolation and anterpolation steps; and (iii) spectral analysis in the error estimates. Of these advantages, the most important is that the FFT interpolations and anterpolations are fast and exact. Since there is no interpolation error, the only significant contributions to the final error are the truncation of the Gegenbauer series and the integration error due to the finite quadrature. Thus, the error analysis is simplified, and we will determine in this paper precise bounds on the final error. In fact, our error analysis is fairly general and can be extended to the classical FMM with schemes that exactly integrate spherical harmonics (see direct and fast global methods in section 2.3). The result is a fast, easy to implement, and controllable version of the FMM, which we detail in the following sections.

3.1. Fourier quadratures.

3.1.1. Spherical Fourier. A periodic complex function defined over $[0, 2\pi] \times [0, 2\pi]$ with spherical symmetry,

$$f(\theta, \phi) = f(2\pi - \theta, \pi + \phi), \tag{3.3}$$

has the trigonometric polynomial representation

$$f(\theta, \phi) = \sum_{n=-\infty}^{\infty} \sum_{m=-\infty}^{\infty} \widetilde{f}_{n,m} e^{\imath(n\theta + m\phi)}, \tag{3.4}$$

where the symmetry condition (3.3) is equivalent to

$$\widetilde{f}_{n,m} = (-1)^m \widetilde{f}_{-n,m} \qquad \forall n, m. \tag{3.5}$$

If functions f and g can be represented exactly by trigonometric polynomials of degree $(\widetilde{N}_\theta, \widetilde{N}_\phi)$, then their L^2 inner product can be computed exactly as

$$\begin{aligned} \langle f, g \rangle = \int_0^{2\pi} \int_0^{2\pi} f(\theta, \phi) \overline{g(\theta, \phi)} \, \mathrm{d}\theta \, \mathrm{d}\phi &= 4\pi^2 \sum_{n=-\widetilde{N}_\theta}^{\widetilde{N}_\theta} \sum_{m=-\widetilde{N}_\phi}^{\widetilde{N}_\phi} \widetilde{f}_{n,m} \overline{\widetilde{g}_{n,m}} \\ &= \frac{4\pi^2}{N_\theta N_\phi} \sum_{n=1}^{N_\theta} \sum_{m=1}^{N_\phi} f(\theta_n, \phi_m) \overline{g(\theta_n, \phi_m)}, \end{aligned}$$

where

$$\begin{aligned} N_\theta &= 2\widetilde{N}_\theta + 1, & \theta_n &= \frac{2\pi n}{N_\theta}, \\ N_\phi &= 2\widetilde{N}_\phi + 1, & \phi_m &= \frac{2\pi m}{N_\phi}. \end{aligned}$$

If N_ϕ is made even (by padding the Fourier coefficients with an extra zero), then

$$f(\theta_n, \phi_m) = f(\theta_{N_\theta - n}, \phi_{N_\phi/2 + m}), \tag{3.6}$$

so that only half of the sampled values must be stored.

3.1.2. Integration with Fourier filtering. A key concern in using the two-dimensional Fourier basis rather than the spherical harmonics in (3.1) is the integration weight $\frac{1}{2} |\sin(\theta)|$. Although the transfer function, $T_{\ell, \boldsymbol{r}_0}$, is bandlimited in θ and ϕ, the modified transfer function, $T^{\mathfrak{s}}_{\ell, \boldsymbol{r}_0}$, is not bandlimited in θ due to the integration weight. In this section, we review a common strategy for integrating functions that are not bandlimited against functions that are bandlimited or nearly bandlimited.

Consider the periodic functions

$$f(\theta) = \sum_{n=-\infty}^{\infty} \widetilde{f}_n e^{\imath n\theta}, \qquad g(\theta) = \sum_{m=-\infty}^{\infty} \widetilde{g}_m e^{\imath m\theta}.$$

Note that their exact integral on $[0, 2\pi]$ is

$$\langle f, g \rangle = \int_0^{2\pi} f(\theta) \overline{g(\theta)} \, \mathrm{d}\theta = 2\pi \sum_{n=-\infty}^{\infty} \widetilde{f}_n \overline{\widetilde{g}_n}. \tag{3.7}$$

Computing this integral numerically with a uniform quadrature of size K yields

$$\begin{aligned}\langle f, g\rangle_K = \frac{2\pi}{K}\sum_{k=1}^{K} f\left(\theta_k\right)\overline{g\left(\theta_k\right)} &= \frac{2\pi}{K}\sum_{n=-\infty}^{\infty}\sum_{m=-\infty}^{\infty}\widetilde{f}_n\overline{\widetilde{g}_m}\sum_{k=1}^{K} e^{2\pi\imath(n-m)k/K}\\ &= 2\pi\sum_{n=-\infty}^{\infty}\sum_{m=-\infty}^{\infty}\widetilde{f}_n\overline{\widetilde{g}_{n+mK}}.\end{aligned}$$

So the error is exactly given by

$$|\langle f, g\rangle - \langle f, g\rangle_K| = 2\pi\left|\sum_{n=-\infty}^{\infty}\sum_{m\neq 0}\widetilde{f}_n\overline{\widetilde{g}_{n+mK}}\right| = 2\pi\left|\sum_{n=-\infty}^{\infty}\sum_{m\neq 0}\widetilde{f}_{n+mK}\overline{\widetilde{g}_n}\right|.$$

The error is therefore determined by the asymptotic decay of both $\widetilde{f}_n$ and $\widetilde{g}_n$. For example, if $|\widetilde{f}_n|$ decays as $1/n^{\alpha}$, $\alpha > 1$, then $\sum_n |\widetilde{f}_{nK}\widetilde{g}_0| \sim 1/K^{\alpha}$, which is the typically expected decay of the error on a uniform quadrature as a function of K. The equivalent result holds for $\widetilde{g}_n$. Thus, if the spectrum of either f or g decays slowly, a very large quadrature K is needed.

However, if the spectrum of f decays slowly and the spectrum of g decays quickly, a much smaller quadrature can be used by truncating the Fourier coefficients of f. Let us define a bandlimited version of $f(\theta)$,

$$f_N(\theta) = \sum_{n=-N}^{N}\widetilde{f}_n e^{\imath n\theta}, \tag{3.8}$$

and use a uniform quadrature of K points to compute the integral. Then

$$\langle f_N, g\rangle_K = 2\pi\sum_{n=-N}^{N}\sum_{m=-\infty}^{\infty}\widetilde{f}_n\overline{\widetilde{g}_{n+mK}},$$

which yields the error

$$|\langle f, g\rangle - \langle f_N, g\rangle_K| = 2\pi\left|\sum_{|n|>N}\widetilde{f}_n\overline{\widetilde{g}_n} - \sum_{n=-N}^{N}\sum_{m\neq 0}\widetilde{f}_n\overline{\widetilde{g}_{n+mK}}\right|. \tag{3.9}$$

This error can be made small by requiring only that $\widetilde{g}_n$ decay quickly enough. That is, $|\widetilde{f}_n|$ can decay slowly, provided $|\widetilde{g}_n|$ decays quickly. The first term of (3.9) represents the truncation error in using the bandlimited f_N instead of f. The second term represents the aliasing error resulting from the finite sampling of the function g. As an example, if we assume that $\widetilde{g}_n$ is negligible for $|n| > N$, then it is sufficient to choose $K = 2N + 1$ so that $|n + mK| > N$ for all $-N \leq n \leq N$ and $m \neq 0$ so that $\widetilde{g}_{n+mK}$ is always negligible.

Detailed numerical results will be presented in section 3.5. However, to demonstrate this idea on a simple example, we build a quadrature to calculate

$$\int_0^{2\pi} |\sin(\theta)|\,\cos(64\cos(\theta))\,\mathrm{d}\theta,$$

which is a simple model problem for (see (3.1))

$$\int_0^{2\pi} T_{\ell,\boldsymbol{r}_0}(\hat{\boldsymbol{s}})\, |\sin(\theta)|\; e^{\imath\kappa\hat{\boldsymbol{s}}\cdot\boldsymbol{r}}\, \mathrm{d}\theta$$

when $\boldsymbol{r} = 64\,\hat{\boldsymbol{z}}/\kappa$. Using a uniform quadrature to compute this integral yields slow convergence in $1/N^2$ because the function $f(\theta) = |\sin(\theta)|$ is C^0 (see Figure 3.1). Instead, if we smooth $|\sin(\theta)|$ and remove the high frequency components following (3.8), we obtain a much faster convergence (see Figure 3.1). The number of quadrature points is $K = 2N + 1$. The Fourier spectrum of $g(\theta) = \cos(64\cos(\theta))$ decays rapidly once $|n| \gtrsim 64$ (see Figure 3.2). Convergence should occur once $N \gtrsim 64$, and this is indeed what is observed in Figure 3.1. The exact value of the integral $\sin(64)/16$ is used as the reference solution to calculate the error.

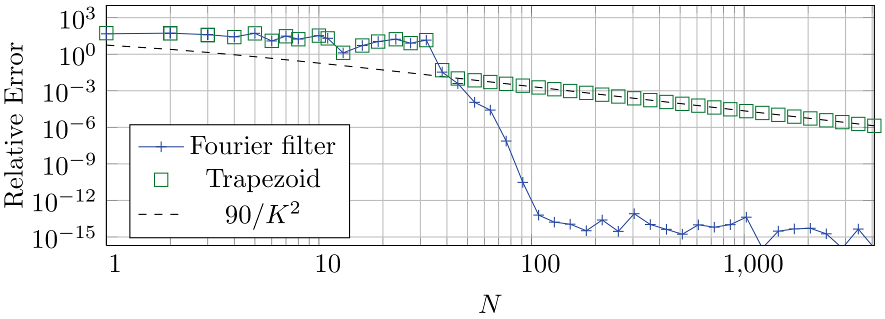

FIG. 3.1. *Relative error in the integral as a function of N. The number of quadrature points is $K = 2N + 1$. The solid line integrates the function using a Fourier filter for $|\sin(\theta)|$ as shown in* (3.9)*. The rapid convergence can be seen after $N \gtrsim 66$. The trapezoid method has a slower convergence in $1/K^2$ because $|\sin(\theta)|$ is only C^0.*

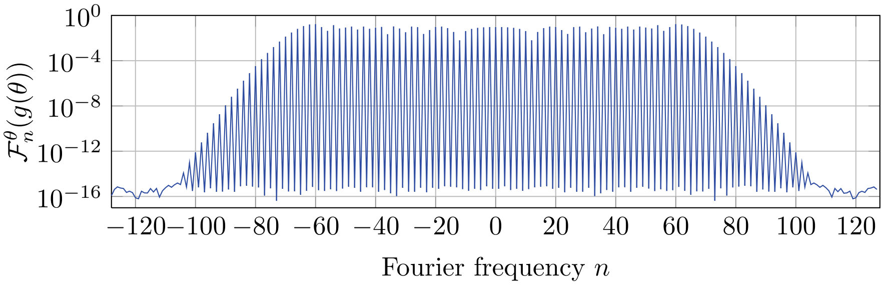

FIG. 3.2. *The spectrum of $g(\theta) = \cos(64\cos(\theta))$ showing a rapid decay of the coefficients for $|n| > 64$. The Fourier spectrum was computed using* 256 *sample points.*

The idea of accurately calculating the integral of a product of two functions by analytically removing high frequencies in one of the two functions can be found in other papers dealing with the FMM for the Helmholtz kernel in the high frequency regime; see, e.g., [8, 7]. In the context of these papers, the smoothing operation (removal of high frequencies) is often termed anterpolation or subsampling. A similar idea is found in Sarvas [24]. The Clenshaw–Curtis quadrature [18] is also based on a similar idea. In McKay Hyde and Bruno [19, Appendix A, pp. 254–257], this idea is used in the more general context of calculating the integral of the product of a discontinuous function with a C^1 piecewise-smooth and periodic function.

3.1.3. Fourier interpolation and anterpolation. Fast Fourier interpolation and anterpolation methods are key to creating an efficient Fourier-based FMM by supersampling trigonometric polynomials and truncating spectral content that does not significantly contribute to the final result. A Fourier interpolation pads the Fourier coefficients of a function with zeros and increases the sampling rate in real space. A Fourier anterpolation removes high frequencies of a function and decreases the sampling rate in real space.

To motivate the use of Fourier interpolations and anterpolations in the Helmholtz multilevel FMM, suppose that the spectrum of $f_N(\theta)$ is bounded, that is, $|\widetilde{f}_n| \leq F$, and that $K = 2N+1$; then (3.9) simplifies to

$$|\langle f, g\rangle - \langle f_N, g\rangle_K| \leq 4\pi F \sum_{|n|>N} |\widetilde{g}_n| \,. \tag{3.10}$$

This can be used to find an appropriate truncation parameter N if $\widetilde{g}_n$ is known or can be approximated. The key step to constructing a fast algorithm is to note that the Fourier series of $E_{\boldsymbol{r}}(\theta,\phi)$ in θ decays rapidly, while the Fourier series of $T^{\boldsymbol{s}}_{\ell,\boldsymbol{r}_0}(\theta,\phi)$ in θ decays slowly. This is due to the slow decay of the Fourier series of $|\sin(\theta)|$:

$$\mathcal{F}^\theta_n[|\sin(\theta)|] = \frac{(-1)^n+1}{\pi(1-n^2)} = \begin{cases} \frac{2}{\pi}\frac{1}{1-n^2} & \text{if } n \text{ even,} \\ 0 & \text{if } n \text{ odd.} \end{cases}$$

The spectrum of the plane-wave is given by

$$E_{\boldsymbol{r}}(\theta,\phi) = e^{\imath\kappa|\boldsymbol{r}|\cos(\varphi_{\hat{\boldsymbol{s}},\boldsymbol{r}})} = \sum_{n=-\infty}^{\infty} \imath^n J_n(\kappa\,|\boldsymbol{r}|) e^{\imath n\varphi_{\hat{\boldsymbol{s}},\boldsymbol{r}}}, \tag{3.11}$$

where $\varphi_{\hat{\boldsymbol{s}},\boldsymbol{r}}$ is the angle between $\hat{\boldsymbol{s}}$ and $\boldsymbol{r}$. The functions J_n decay rapidly once $n > \kappa\,|\boldsymbol{r}|$, resulting in an approximately bandlimited function. In the following sections, we use the spectral decay of the translation function $E_{\boldsymbol{r}}(\theta,\phi)$ (which plays the role of g in the previous section) to determine an appropriate Fourier truncation of the modified transfer function, $T^{\boldsymbol{s}}_{\ell,\boldsymbol{r_0}}(\theta,\phi)$ (which plays the role of f).

Equation (3.10) illustrates the need for interpolation and anterpolation as described in section 2.3 but in the context of the Fourier basis. With $\widetilde{g}_n = \mathcal{F}^\theta_n[E_{|\boldsymbol{r}|\hat{\boldsymbol{z}}}] = \imath^n J_n(\kappa\,|\boldsymbol{r}|)$, the truncation N is approximately $N \gtrsim \kappa\,|\boldsymbol{r}| \sim \ell$. During the upward pass, $|\boldsymbol{r}|$ increases and we require more modes in $f = T^{\boldsymbol{s}}_{\ell,\boldsymbol{r}_0}$. During the downward pass, the incoming local field, L^l in section 2, is to be integrated against translation functions of increasingly smaller $|\boldsymbol{r}|$. See (2.4) with $\boldsymbol{r} = \boldsymbol{c}^L_\alpha - \boldsymbol{x}_i$. The high modes of the field do not significantly contribute to the exact integral (3.7), so the field can be safely anterpolated at each downward step.

3.2. Computing the bandlimited modified transfer function (M2L). Select a two-dimensional uniform quadrature with points (θ_n,ϕ_m). Noting that the plane-wave $E_{\boldsymbol{r}}(\theta,\phi)$ and modified transfer function $T^{\boldsymbol{s}}_{\ell,\boldsymbol{r}_0}(\theta,\phi)$ both have spherical symmetry (3.3), the computational and memory cost are reduced by requiring N_ϕ to be even so that only half of the quadrature points need to be computed and stored.

Additionally, in an FMM with a single octree, there are 316 distinct transfer vectors $\boldsymbol{r}_0$ per level. By enforcing symmetries in the quadrature, the number of

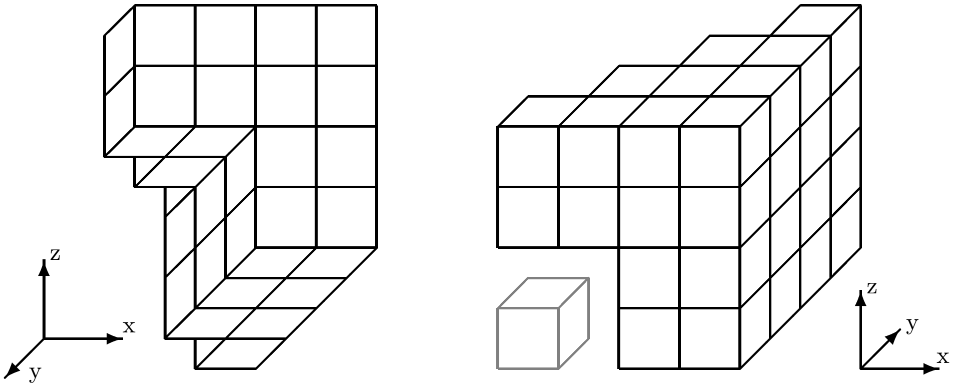

Fig. 3.3. *The center of each box represents one transfer vector* $\boldsymbol{r}_0$ *which must be computed. The left and right panels represent the same set of boxes viewed at two different angles. Due to the symmetries of the quadrature, we need only compute transfer vectors with* $x, y, z \geq 0$ *and* $x \geq y$. *We therefore end up with essentially half of an octant. Specifically,* 34 *transfer vectors are required; they can be reflected into any of the* 316 *needed.*

modified transfer functions that must be precomputed is reduced. Specifically, by requiring N_θ to be a multiple of 2 and N_ϕ to be a multiple of 4, we enforce reflection symmetries in the $z = 0$, $x = 0$, $y = 0$, $x = y$, and $x = -y$ planes. This reduces the number of modified transfer functions that need to be precomputed from 316 per level to 34—saving a factor of 9.3 in memory and costing a negligible permutation of the values of a computed modified transfer function. See Figure 3.3.

Suppose we have chosen a quadrature characterized by (N_θ, N_ϕ). Following section 3.1.2, we need to exactly calculate a bandlimited version of $T^{\mathfrak{s}}_{\ell,\boldsymbol{r}_0}$, called $T^{\mathfrak{s},L}_{\ell,\boldsymbol{r}_0}$, such that

$$\mathcal{F}^\theta_n[T^{\mathfrak{s},L}_{\ell,\boldsymbol{r}_0}(\theta,\phi))] = \begin{cases} \mathcal{F}^\theta_n[T^{\mathfrak{s}}_{\ell,\boldsymbol{r}_0}(\theta,\phi)], & |n| \leq N_\theta/2 - 1, \\ 0 & \text{otherwise.} \end{cases}$$

Since $T_{\ell,\boldsymbol{r}_0}$ is bandlimited in θ with bandwidth $2\ell + 1$, only the frequencies $|m| \leq N_\theta/2 + \ell - 1$ of $|\sin(\theta)|$ contribute to $T^{\mathfrak{s},L}_{\ell,\boldsymbol{r}_0}$. Therefore, the exact bandlimited modified transfer function, $T^{\mathfrak{s},L}_{\ell,\boldsymbol{r}_0}$, can be computed using the pseudocode in Algorithm 1.

Algorithm 1: Pseudocode to compute the bandlimited modified transfer function, $T^{\mathfrak{s},L}$, given ℓ, $\boldsymbol{r}_0$, N_θ, and N_ϕ.

```
$\widetilde{s}_n \leftarrow \mathcal{F}^\theta_n[|\sin(\theta)|] = \frac{(-1)^n+1}{\pi(1-n^2)}$ for all $|n| \leq N_\theta/2 + \ell - 1$;
for $m = 0, \ldots, N_\phi/2 - 1$, do
    $T(\theta_n, \phi_m) \leftarrow \frac{1}{2} T_{\ell,\boldsymbol{r}_0}(\frac{2\pi n}{2\ell+1}, \phi_m)$, $n = 0, \ldots, 2\ell$;
    $\widetilde{T}_n \leftarrow \mathcal{F}^\theta_n[T]$;
    $T^{\mathfrak{s},L}_n \leftarrow \widetilde{s} \otimes \widetilde{T}$ convolution of Fourier series;
    $T^{\mathfrak{s},L}_n \leftarrow$ truncate to frequencies $|n| \leq N_\theta/2 - 1$;
    $T^{\mathfrak{s},L}(\theta_n, \phi_m) \leftarrow$ inverse transform of $T^{\mathfrak{s},L}_n$;
```

Algorithm 1 yields the bandlimited modified transfer function at (θ_n, ϕ_m), $0 \leq n < N_\theta$, $0 \leq m < N_\phi/2$, which can be unwrapped to the remaining points by using the spherical symmetry (3.6). Note that this calculation can also be performed in real space. It is equivalent to making a Fourier interpolation of T_k from $2\ell + 1$ points to $N_\theta + 2\ell - 1$ points, multiplying by a bandlimited $|\sin(\theta)|$, and performing a Fourier anterpolation back to N_θ points, as shown in Figure 3.4.

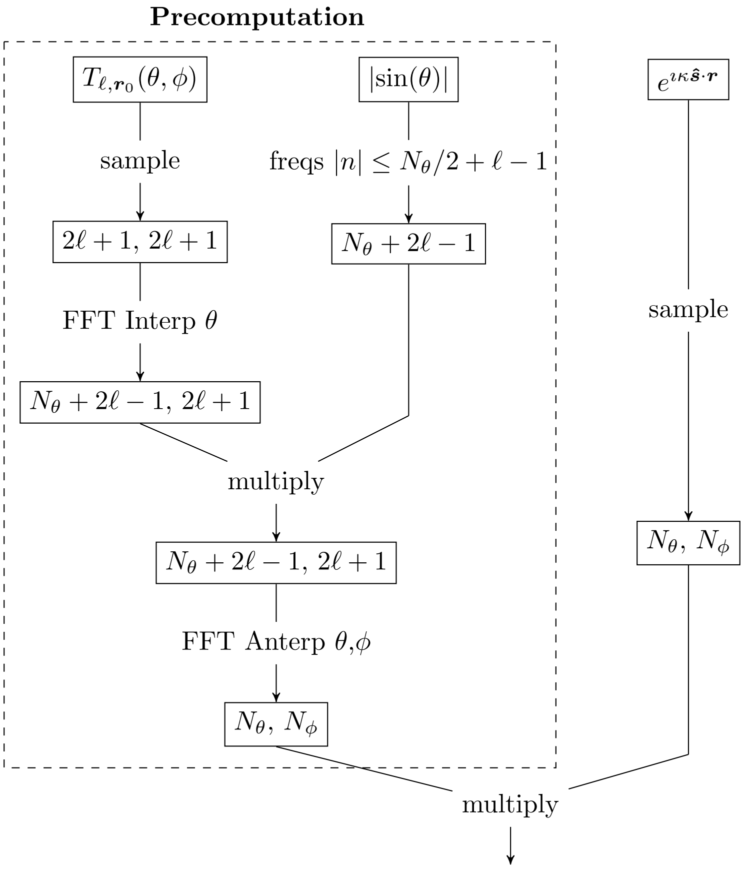
Precomputation
$T_{\ell,\boldsymbol{r}_0}(\theta,\phi)$
$|\sin(\theta)|$
$e^{\imath\kappa\hat{\boldsymbol{s}}\cdot\boldsymbol{r}}$
sample
freqs $|n| \leq N_\theta/2 + \ell - 1$
$2\ell+1,\, 2\ell+1$
$N_\theta + 2\ell - 1$
FFT Interp θ
sample
$N_\theta + 2\ell - 1,\, 2\ell + 1$
multiply
$N_\theta + 2\ell - 1,\, 2\ell + 1$
$N_\theta,\, N_\phi$
FFT Anterp θ,ϕ
$N_\theta,\, N_\phi$
multiply

FIG. 3.4. *Procedure for precomputing the bandlimited modified transfer function and its application to an outgoing field. The boxed numbers give the numbers of quadrature points for θ and ϕ (N_θ and N_ϕ) at each stage.*

Because sampling the transfer function at a single point is an $\mathcal{O}(\ell)$ operation and $N_\phi, N_\theta \in \mathcal{O}(\ell)$, the algorithm as presented is $\mathcal{O}(\ell^3)$. The computation of the transfer function at all sample points can be accelerated to $\mathcal{O}(\ell^2)$, as in [12], by taking advantage of its symmetry about the $\hat{\boldsymbol{r}}_0$-axis and using interpolation algorithms, but at the cost of introducing additional error.

3.3. Choice of quadrature. The quadrature parameters can be constructively computed by determining the maximum error they incur. The error in computing the desired integral using the bandlimited modified transfer function with a finite uniform quadrature is

$$|\varepsilon_I| = \left| \int_0^{2\pi} \int_0^{2\pi} E_{\boldsymbol{r}}(\theta,\phi)\, T^{\mathfrak{s}}_{\ell,\boldsymbol{r}_0}(\theta,\phi)\, \mathrm{d}\phi\, \mathrm{d}\theta - \frac{4\pi^2}{N_\theta N_\phi} \sum_{n=1}^{N_\theta} \sum_{m=1}^{N_\phi} E_{\boldsymbol{r}}(\theta_n,\phi_m)\, T^{\mathfrak{s},L}_{\ell,\boldsymbol{r}_0}(\theta_n,\phi_m) \right|, \tag{3.12}$$

where $T^{\mathfrak{s},L}_{\ell,\boldsymbol{r}_0}(\theta_n,\phi_m)$ is the bandlimited modified transfer function described in section 3.2.

3.3.1. Choosing $\boldsymbol{N_\theta}$. The worst case for ε_I in terms of N_θ occurs when $\boldsymbol{r}$ and $\boldsymbol{r}_0$ are aligned with the z-axis. This causes all spectral information to be contained in the θ-direction and makes ε_I a function of N_θ only. Integrating ϕ out of (3.12) yields

$$\left|\varepsilon_I^\theta\right| = 2\pi \left| \int_0^{2\pi} E_{\pm|r|\hat{\boldsymbol{z}}}(\theta,0)\, T^{\mathfrak{s}}_{\ell,|\boldsymbol{r}_0|\hat{\boldsymbol{z}}}(\theta,0)\, \mathrm{d}\theta - \frac{2\pi}{N_\theta} \sum_{n=1}^{N_\theta} E_{\pm|r|\hat{\boldsymbol{z}}}(\theta_n,0)\, T^{\mathfrak{s},L}_{\ell,|\boldsymbol{r}_0|\hat{\boldsymbol{z}}}(\theta_n,0) \right|,$$

which is equivalent to the one-dimensional case considered in section 3.1. Retrieving the Fourier coefficients of the plane-wave in the case $\hat{\boldsymbol{r}} = \hat{\boldsymbol{z}}$ from (3.11) and numerically computing exactly the low frequencies of the modified transfer function as described in section 3.2, then (3.9) leads to

$$\left|\varepsilon_I^\theta\right| = 4\pi^2 \left| \sum_{|n|\geq N_\theta/2} \widetilde{T}^{\mathfrak{s}}_n \imath^n J_{\mp n}(\kappa\,|\boldsymbol{r}|) - \sum_{|n|<N_\theta/2} \sum_{m\neq 0} \widetilde{T}^{\mathfrak{s}}_n \imath^{n+mN_\theta} J_{\mp(n+mN_\theta)}(\kappa\,|\boldsymbol{r}|) \right|,$$

where $\widetilde{T}^{\mathfrak{s}}_n = \mathcal{F}^\theta_n[T^{\mathfrak{s}}_{\ell,|\boldsymbol{r}_0|\hat{\boldsymbol{z}}}(\theta,0)]$. Applying the triangle inequality yields the simplification

$$\left|\varepsilon_I^\theta\right| \leq 4\pi^2 \sum_{|n|\geq N_\theta/2} \left|\widetilde{T}^{\mathfrak{s}}_n\right| |J_n(\kappa\,|\boldsymbol{r}|)| + \sum_{|n|<N_\theta/2} \left|\widetilde{T}^{\mathfrak{s}}_n\right| \sum_{m\neq 0} |J_{n+mN_\theta}(\kappa\,|\boldsymbol{r}|)| \tag{3.13}$$

and is valid for all directions $\boldsymbol{r}$. When $\kappa\,|\boldsymbol{r}| \geq 1$, the Bessel functions decay super-exponentially when $n \gtrsim \kappa\,|\boldsymbol{r}|$. Due to this very fast decay, we find it is sufficient to keep only the lowest order Bessel function terms:

$$\left|\varepsilon_I^\theta\right| \lesssim 4\pi^2 \sum_{n=-\infty}^{\infty} \left|\widetilde{T}^{\mathfrak{s}}_n\right| \left|J_{M(N_\theta,n)}(\kappa\,|\boldsymbol{r}|)\right|, \tag{3.14}$$

where

$$M(N_\theta,n) = \begin{cases} N_\theta - |n|\,, & |n| < N_\theta/2, \\ |n|\,, & |n| \geq N_\theta/2. \end{cases} \tag{3.15}$$

Thus, (3.14) keeps the first sum in (3.13), while, from the second sum, only Bessel functions of order up to $\pm(N_\theta - 1)$ are kept. Since the argument is of the order $\kappa\,|r| \sim \ell \sim N_\theta/2$, the neglected terms are well within the region of exponential decay for the Bessel functions. Equation (3.14) can be then used to accurately search for a value N_θ via Algorithm 2.

Since N_θ^{max} in Algorithm 2 is typically only a small constant larger than $2\ell+1$, the algorithm as presented is dominated by the computation of the $\mathcal{O}(\ell)$ modified transfer function values and requires $\mathcal{O}(\ell^2)$ operations. Important optimizations include using more advanced searching methods (such as bisection), applying the symmetries $\widetilde{E}^*_m = \widetilde{E}^*_{-m}$ and $\widetilde{T}_m = \widetilde{T}_{-m}$, and taking advantage of the very fast decay of J_n to neglect very small terms in the dot product.

Algorithm 2: Pseudocode to compute N_θ given ℓ, $|\boldsymbol{r}_0|$, and $|\boldsymbol{r}|$.

```
Choose $N_\theta^{max}$ sufficiently larger than $2\ell + 1$;
$T_n \leftarrow T^{\mathfrak{s},L}_{\ell,|\boldsymbol{r}_0|\hat{\boldsymbol{z}}}(\frac{2\pi n}{N_\theta^{max}}, 0)$, $n = 0, \ldots, N_\theta^{max} - 1$;
$\widetilde{T}_n \leftarrow \big|\mathcal{F}^\theta_n[T]\big|$;
$\widetilde{E}_n \leftarrow |J_n(\kappa\,|\boldsymbol{r}|)|$;
for $N_\theta$ from $2\ell$ to $N_\theta^{max}$ by 2 do
    $\widetilde{E}^*_n \leftarrow \widetilde{E}_{M(N_\theta,n)}$;
    if $\widetilde{E}^* \cdot \widetilde{T} < \varepsilon/4\pi^2$ then
        return $N_\theta$
```

3.3.2. Choosing N_ϕ. After determining an appropriate N_θ, letting N_ϕ be a function of θ allows one to reduce the number of quadrature points without affecting the error. The worst case for the integration error in terms of N_ϕ occurs when $\boldsymbol{r}$ and $\boldsymbol{r}_0$ are in the xy-plane. Without loss of generality, suppose $\hat{\boldsymbol{r}} = \hat{\boldsymbol{x}}$. Then, for a fixed θ_n, the plane-wave (3.11) can be expressed as

$$E_{\boldsymbol{r}}(\theta_n, \phi) = e^{\imath\kappa\hat{\boldsymbol{s}}(\theta_n,\phi)\cdot\boldsymbol{r}} = \sum_{m=-\infty}^{\infty} i^m J_m(\kappa\,|\boldsymbol{r}|\sin(\theta_n))\, e^{\imath m\phi}. \tag{3.16}$$

Since $J_m(\kappa\,|\boldsymbol{r}|\sin(\theta_n))$ decays exponentially when $m \gtrsim \kappa\,|\boldsymbol{r}|\sin(\theta_n)$, the series can be truncated at $N_\phi(\theta_n) \sim \kappa\,|\boldsymbol{r}|\sin(\theta_n)$ without incurring any appreciable error, provided that the exact Fourier coefficients of $T^{\mathfrak{s},L}_{\ell,\boldsymbol{r}_0}(\theta,\phi)$ in ϕ are available. This is the case since $T^{\mathfrak{s},L}_{\ell,\boldsymbol{r}_0}$ is bandlimited in ϕ. Additionally, letting N_ϕ be a function of θ requires a final step in the computation of the modified transfer function. Section 3.2 computed the transfer function on an $(N_\theta/2+1) \times N_\phi$ grid. With $N_\phi \to N_\phi(\theta_n)$, the data computed for each θ_n must be Fourier anterpolated from length N_ϕ to length $N_\phi(\theta_n)$.

Estimates of $N_\phi(\theta_n)$ can be developed by determining when $J_m(\kappa\,|\boldsymbol{r}|\sin(\theta_n))$ becomes exponentially small, as in the computation of the EBF in [4]. However, we find that the EBF-generated quadrature typically overestimates the sampling rate. To accurately compute $N_\phi(\theta_n)$, a similar procedure to that in section 3.3.1 is applied. After determining the appropriate N_θ, the $T^{\mathfrak{s},L}_{\ell,\boldsymbol{r}_0}$ can be computed. For a given θ, the error is

$$\big|\varepsilon^\phi_I\big| = \left|\int_0^{2\pi} E_{\pm|\boldsymbol{r}|\hat{\boldsymbol{x}}}(\theta,\phi)\, T^{\mathfrak{s},L}_{\ell,\pm|\boldsymbol{r}_0|\hat{\boldsymbol{x}}}(\theta,\phi)\,\mathrm{d}\phi - \frac{2\pi}{N_\phi(\theta)}\sum_{m=1}^{N_\phi(\theta)} E_{\pm|\boldsymbol{r}|\hat{\boldsymbol{x}}}(\theta,\phi_m)\, T^{\mathfrak{s},LL}_{\ell,\pm|\boldsymbol{r}_0|\hat{\boldsymbol{x}}}(\theta,\phi_m)\right|,$$

where $T^{\mathfrak{s},LL}$ is the bandlimited modified transfer function with both the θ-frequencies and ϕ-frequencies truncated for the quadrature. That is, if

$$T^{\mathfrak{s}}_{\ell,\boldsymbol{r}_0}(\theta,\phi) = \frac{1}{2}T_{\ell,\boldsymbol{r}_0}(\theta,\phi)\,|\sin(\theta)| = \sum_{n=-\infty}^{\infty}\sum_{m=-\ell}^{\ell} \widetilde{T}^{\mathfrak{s}}_{n,m} e^{n\theta+m\phi},$$

then

$$T^{\mathfrak{s},L}_{\ell,\boldsymbol{r}_0}(\theta,\phi) = \sum_{n=-N_\theta/2+1}^{N_\theta/2-1}\sum_{m=-\ell}^{\ell} \widetilde{T}^{\mathfrak{s}}_{n,m} e^{n\theta+m\phi}$$

and

$$T^{\mathfrak{s},LL}_{\ell,\boldsymbol{r}_0}(\theta,\phi) = \sum_{n=-N_\theta/2+1}^{N_\theta/2-1} \; \sum_{m=-N_\phi(\theta_n)/2+1}^{N_\phi(\theta_n)/2-1} \widetilde{T}^{\mathfrak{s}}_{n,m} e^{n\theta+m\phi}.$$

We again apply the result of section 3.1 by retrieving the Fourier coefficients of the plane-wave in the case $\hat{\boldsymbol{r}} = \hat{\boldsymbol{x}}$ from (3.16) and computing exactly the low frequencies of the modified transfer function. Equation (3.9) leads to

$$\left|\varepsilon_I^{\phi}\right| \lesssim 2\pi \sum_{m=-\ell}^{\ell} \left|\widetilde{T}^{\mathfrak{s},L}_m(\theta)\right| \left|J_{M(N_\phi(\theta),m)}(\kappa\,|\boldsymbol{r}|\sin(\theta))\right| \tag{3.17}$$

Equation (3.17) can then be used to search for a value of $N_\phi(\theta_n)$ via Algorithm 3.

Algorithm 3: Pseudocode to compute each $N_\phi(\theta_n)$ given ℓ, $|\boldsymbol{r}_0|$, $|\boldsymbol{r}|$, and N_θ.

```
Choose $N_\phi^{max}$ sufficiently larger than $2\ell+1$;
for $n = 0, \dots, N_\theta/2$, do
    $T_m \leftarrow T^{\mathfrak{s},L}_{\ell,|\boldsymbol{r}_0|\hat{\boldsymbol{x}}}(\theta_n, \frac{2\pi m}{2\ell+1})$, $m = 0, \dots, 2\ell$;
    $\widetilde{T}_m \leftarrow \left|\mathcal{F}^{\phi}_m[T]\right|$;
    $\widetilde{E}_m \leftarrow |J_m(\kappa\,|\boldsymbol{r}|\sin(\theta_n))|$;
    for $N_\phi(\theta_n)$ from 4 to $N_\phi^{max}$ by 4 do
        $\widetilde{E}^*_m \leftarrow \widetilde{E}_{M(N_\phi(\theta_n),m)}$;
        if $\widetilde{E}^* \cdot \widetilde{T} < \varepsilon/4\pi^2$ then
            Save $N_\phi(\theta_n)$
```

Since N_ϕ^{max} is only a small constant larger than $2\ell+1$, the algorithm as presented is dominated by the computation of the modified transfer function and requires $\mathcal{O}(\ell^3)$ operations. Optimizations similar to those presented in section 3.3.1 can be applied. Using the EBF as an initial guess in the search for $N_\phi(\theta_n)$ further improves the searching speed. Additionally, only half of the $N_\phi(\theta_n)$'s may be computed due to symmetry about the $z = 0$ plane.

3.3.3. Choosing $|\boldsymbol{r}|$ and $|\boldsymbol{r}_0|$. The previous algorithms require representative values of $|\boldsymbol{r}|$ and $|\boldsymbol{r}_0|$ for each level of the tree. The worst-case transfer vectors, $\boldsymbol{r}_0$, are those with smallest length. If a_l is the box size at level l, then $|\boldsymbol{r}_0| = 2a_l$ is the smallest transfer vector length in the common one buffer-box case.

The worst-case value of $|\boldsymbol{r}|$ is the largest. For a box of size a_l, $|\boldsymbol{r}| \leq a_l\sqrt{3}$. However, using $|\boldsymbol{r}| = a_l\sqrt{3}$ in the previous methods is often too conservative. This case only occurs when two points are located in the exact corners of the boxes—a very rare case. See Figure 3.5. Instead, we let $|\boldsymbol{r}| = \alpha a_l\sqrt{3}$ for some $\alpha \in [0,1]$. A high α guarantees an upper bound on the error generated by the quadrature, but the points which actually generate this error become increasingly rare. A lower value of α will yield a smaller quadrature, but more points may fall outside the radius $|\boldsymbol{r}|$ for which the upper bound on the error is guaranteed.

3.3.4. Number of quadrature points. Recall from section 2.2 that the typical approach in the FMM is to use $N+1$ uniform points in the ϕ-direction and $\frac{N+1}{2}$

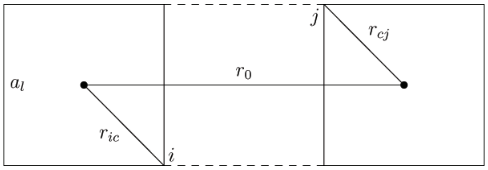

Fig. 3.5. *The worst case $\boldsymbol{r}$ and $\boldsymbol{r}_0$, projected from the three-dimensional box. Here, $|\boldsymbol{r}_0| = 2a_l$ and i and j are on the opposite corners of the box so that $|\boldsymbol{r}| = |\boldsymbol{r}_{ic}| + |\boldsymbol{r}_{cj}| = a_l\sqrt{3}$.*

Gauss–Legendre points in the θ-direction.

$$K_{sh} = 2(\ell + 1)^2 \sim 2\ell^2$$

quadrature points.

For a given Gegenbauer series truncation ℓ, the total number of quadrature points required in the Fourier based FMM is approximately

$$K_{fb} \approx \frac{N_\theta}{2} \frac{1}{\pi} \int_0^\pi N_\phi(\theta) \, \mathrm{d}\theta.$$

Noting that $N_\theta \sim \ell$ and $N_\phi(\theta) \sim 2\ell \sin(\theta)$ and keeping only the leading term in ℓ, we are left with

$$K_{fb} \sim \frac{4}{\pi} \ell^2 \approx 1.3 \, \ell^2.$$

Thus, the method presented in this paper uses approximately 0.64 times the number of quadrature points in the standard FMM. However, it is possible that the same N_ϕ optimization can be applied to the standard FMM for the same reasons it was applied in section 3.3.2 to reduce the standard quadrature to a comparable size.

3.4. Interpolation and anterpolation (M2M and L2L). The Fourier-based FMM directly uses FFTs in the interpolation and anterpolation step. This makes the time-critical upward pass (M2M) and downward pass (L2L) efficient and easy to implement while retaining the exactness of global methods.

Characterize a quadrature by an array of length $N_\theta/2 + 1$,

$$K = [N_\phi(\theta_0), N_\phi(\theta_1), \ldots, N_\phi(\theta_{N_\theta/2-1}), N_\phi(\theta_{N_\theta/2})],$$

noting that $N_\phi(\theta_n) = N_\phi(\theta_{N_\theta/2+n})$. The data $F(\theta_n, \phi_m)$ sampled on a quadrature K is transformed to another quadrature K' by performing a sequence of Fourier interpolations and anterpolations. Let

$$\mathcal{N}_\phi = \max\Big[\max_{0 \le n \le N_\theta/2} N_\phi(\theta_n), \max_{0 \le n \le N'_\theta/2} N'_\phi(\theta_n) \Big].$$

Then the following steps, as illustrated in Figure 3.6, perform an exact interpolation/anterpolation using only FFTs:

1. For each θ_n, $0 \le n \le N_\theta/2$, Fourier interpolate the data $[F(\theta_n, \phi_m)]_{m=0}^{N_\phi(\theta_n)-1}$ from length $N_\phi(\theta_n)$ to $\mathcal{N}_\phi$.
2. For each ϕ_m, $0 \le m < \mathcal{N}_\phi/2$, apply symmetry (3.6) to construct the θ-periodic sequences $[F(\theta_n, \phi_m)]_{n=0}^{N_\theta-1}$.

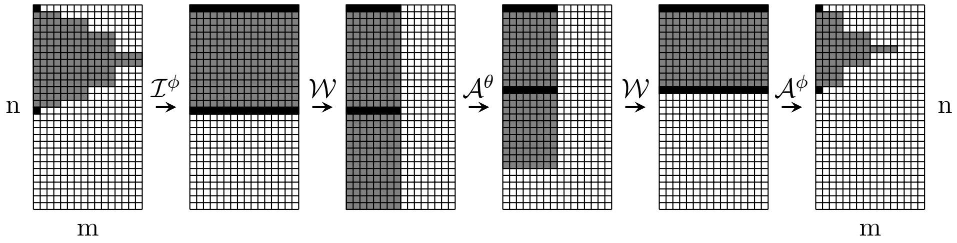

FIG. 3.6. *The data profile at each step in an anterpolation from a large quadrature K with $N_\theta = 30$ to a smaller quadrature K' with $N'_\theta = 24$. The data corresponding to a pole has been darkened for clarity.*

3. For each ϕ_m, $0 \le m < \mathcal{N}_\phi/2$, Fourier interpolate the data $[F(\theta_n, \phi_m)]_{n=0}^{N_\theta - 1}$ from length N_θ to N'_θ.
4. For each θ_n, $0 \le n \le N'_\theta/2$, apply symmetry (3.6) to construct the ϕ-periodic sequences $[F(\theta_n, \phi_m)]_{m=0}^{\mathcal{N}_\phi - 1}$.
5. For each θ_n, $0 \le n \le N'_\theta/2$, Fourier anterpolate the data $[F(\theta_n, \phi_m)]_{m=0}^{\mathcal{N}_\phi - 1}$ from length $\mathcal{N}_\phi$ to $N'_\phi(\theta_n)$.

A slightly more efficient algorithm uses the symmetry (3.5) rather than (3.6).

3.5. Numerical results. In the following sections, we will use the total measured error, ε_T, defined as

$$\varepsilon_T = \frac{e^{\imath\kappa|\boldsymbol{r}+\boldsymbol{r}_0|}}{|\boldsymbol{r}+\boldsymbol{r}_0|} - \sum_{n=1}^{N_\theta} \frac{4\pi^2}{N_\theta N_\phi(\theta_n)} \sum_{m=1}^{N_\phi(\theta_n)} E_{\boldsymbol{r}}(\theta_n, \phi_m)\, T^{\mathfrak{s},L}_{\ell,\boldsymbol{r}_0}(\theta_n, \phi_m). \tag{3.18}$$

The total Gegenbauer truncation error, ε_G, is

$$\varepsilon_G = \frac{e^{\imath\kappa|\boldsymbol{r}+\boldsymbol{r}_0|}}{|\boldsymbol{r}+\boldsymbol{r}_0|} - \imath\kappa \sum_{n=0}^{\ell} (-1)^n (2n+1) h_n^{(1)}(\kappa\,|\boldsymbol{r}_0|) j_n(\kappa\,|\boldsymbol{r}|) P_n(\hat{\boldsymbol{r}} \cdot \hat{\boldsymbol{r}}_0). \tag{3.19}$$

The total integration error, ε_I, is

$$\varepsilon_I = \varepsilon_T - \varepsilon_G. \tag{3.20}$$

The error ε_G is controlled primarily by the choice of ℓ, and ε_I is controlled primarily by the choice of N_θ and N_ϕ's. In the following sections we will show that using Algorithms 2 and 3 allows for accurate control of ε_I. To prevent roundoff errors from accumulating in the algorithm due to the divergence of the transfer function, we allow ε_G to grow when κa is small (e.g., ℓ is chosen small enough to avoid significant roundoff errors). In the following sections, we show that the total error $\varepsilon_T = \varepsilon_G + \varepsilon_I$ in the worst case can be well controlled when $\kappa a_l \gtrsim 1$, and the algorithm uses a smaller quadrature than with simpler estimates of the quadrature parameters.

3.5.1. Single-level error. In the first test case, shown in Figure 3.7, we examine the choice of N_θ by using the worst case $\boldsymbol{r}_0 = |\boldsymbol{r}_0|\,\hat{\boldsymbol{z}}$. Here, the optimal Gegenbauer truncation ℓ is obtained as explained in section 2.1 (see (2.6)). The quadrature for computing the integral (3.1) was constructed as described in section 3.3. With box size $a = 1$, the quadrature and Gegenbauer truncation are constructed with $|\boldsymbol{r}| = 0.8a\sqrt{3}$,

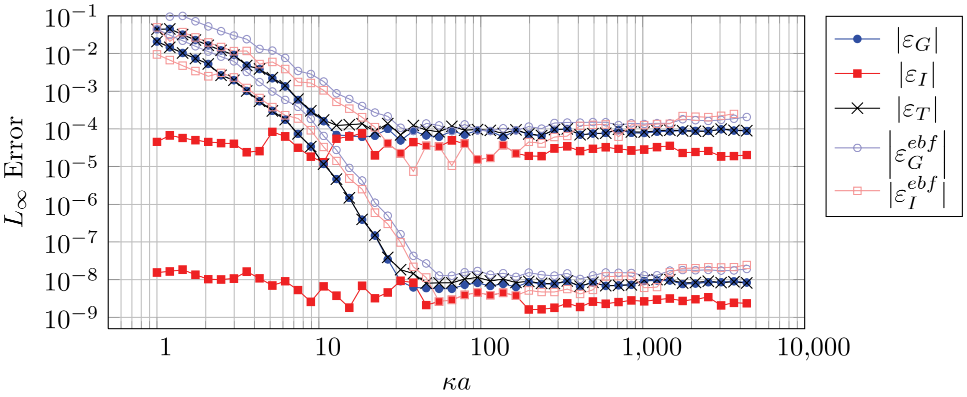

Fig. 3.7. *Results with* $\boldsymbol{r}_0 \sim \hat{\boldsymbol{z}}$, *target accuracy* $\varepsilon = 10^{-4}$ *and* $\varepsilon = 10^{-8}$, *the Gegenbauer truncation chosen as in section* 2.1, N_θ *chosen as in section* 3.3.1, *and box size* $a = 1$. *The* ε_G^{ebf} *and* ε_I^{ebf} *curves correspond to using the EBF to choose* ℓ *and* $N_\theta = 2\ell + 1$, *respectively; this choice leads to a drift in the error as the frequency increases. In contrast, using our error estimate and scheme to choose the parameters, the error follows closely the target error.*

$|\boldsymbol{r}_0| = 2a$, and target error $\varepsilon = 10^{-4}, 10^{-8}$. The plotted errors represent the maximum found over many directions $\hat{\boldsymbol{r}}$, verifying that the worst case is $\boldsymbol{r} \sim \pm\hat{\boldsymbol{z}}$.

We note that the target error ε is accurately achieved for all frequencies (except in the low frequency breakdown because of roundoff errors). The actual error is within a factor of 2 or less of the target error. The increase in error for small box sizes corresponds to the low frequency breakdown when the transfer function has very large amplitude. At small sizes, in order to avoid roundoff errors, we limit the size of ℓ. This leads to a growth of ε_G, while ε_I remains small.

We also show a comparison with a simple heuristic for choosing N_θ. Based on the truncation of the Gegenbauer series, we expect $N_\theta = 2\ell+1$ (rounded up to the nearest multiple of 2) to be a reasonable guess. The error resulting from this choice is shown by the curve labeled ε_I^{ebf}. This shows that our scheme produces a more accurate estimate of this parameter and, in particular, with our scheme, we no longer see the drift in error seen with ε_I^{ebf}. Similarly, we observe an improvement over ε_G^{ebf}, which shows the Gegenbauer error from choosing ℓ using the more standard EBF formula.

The second test case, shown in Figure 3.8, shows the accuracy resulting from our choice of N_ϕ using the worst case $\boldsymbol{r}_0 = |\boldsymbol{r}_0|\,\hat{\boldsymbol{x}}$. Again, the direct computation was used to find the optimal Gegenbauer truncation ℓ and the quadrature was constructed following section 3.3. With box size $a = 1$, the quadrature and Gegenbauer truncation are constructed with $|\boldsymbol{r}| = 0.8a\sqrt{3}$, $|\boldsymbol{r}_0| = 2a$, and target error $\varepsilon = 10^{-4}$ and 10^{-8}. The plotted errors represent the maximum found over many directions $\hat{\boldsymbol{r}}$, verifying that the worst case is $\boldsymbol{r} \sim \pm\hat{\boldsymbol{x}}$. We can see that the target error is achieved even more accurately than in the $\hat{\boldsymbol{z}}$ case. This is due to the number of N_ϕ that are chosen—one for each θ_n. The $N_\phi(\theta_n)$ that yields the most inaccurate result dominates the others, bringing the integration error very close to the target. The comparison heuristics ε_I^{ebf} uses the constant $N_\phi = 2\ell + 1$ (rounded up to the nearest multiple of 4) and ε_G^{ebf} chooses ℓ with the EBF formula. Again, the target error should be relaxed in the low frequency breakdown regime.

Figure 3.9 shows the ratio of the number of points in the quadrature used in Figure 3.8 to the number of quadrature points that would be used in a typical spherical

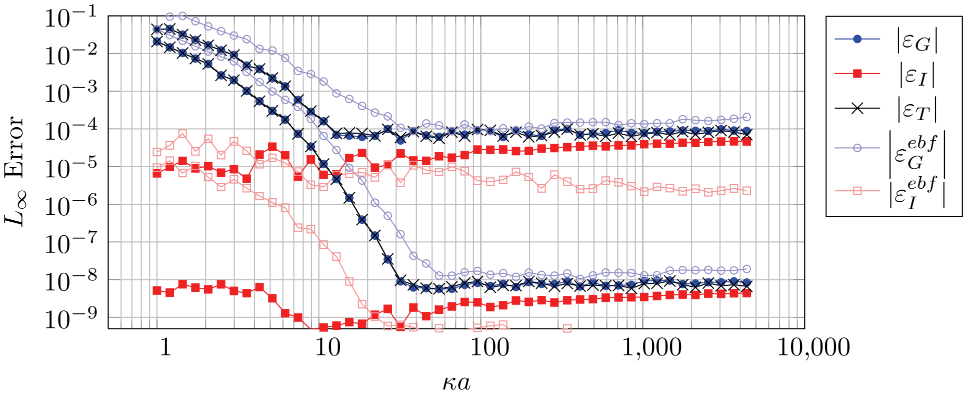

FIG. 3.8. *Results with $\boldsymbol{r}_0 \sim \hat{\boldsymbol{x}}$, target accuracy $\varepsilon = 10^{-4}$ and $\varepsilon = 10^{-8}$, the Gegenbauer truncation chosen as in section* 2.1*, N_θ and N_ϕ chosen as in sections* 3.3.1 *and* 3.3.2*, and box size $a = 1$. The ε_G^{ebf} and ε_I^{ebf} curves correspond to using the EBF to choose ℓ and $N_\phi(\theta_n) = 2\ell + 1$, respectively; this choice greatly overestimates the size of the quadrature needed in the ϕ-direction, especially near the poles $\theta = \{0, \pi\}$. In contrast, using our error estimate and scheme to choose the parameters, the error follows closely the target error.*

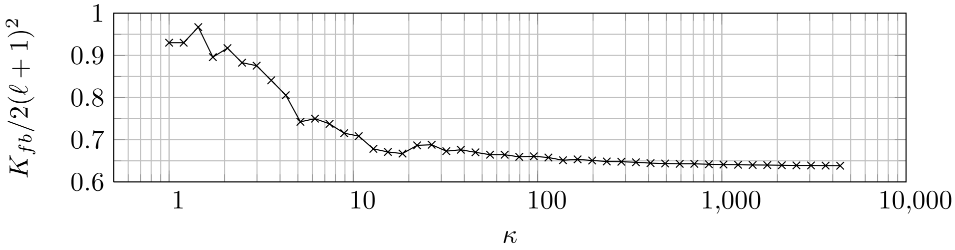

FIG. 3.9. *The ratio of the number of quadrature points required in the Fourier-based FMM and what would be used in a typical spherical harmonics-based FMM for the same ℓ. The curve asymptotes close to $2/\pi \approx 0.64$, as expected from section* 3.3.4.

harmonics-based FMM using the same Gegenbauer truncation ℓ chosen by the direct calculation. The procedures presented in this paper result in a quadrature which is substantially smaller than what would typically be used. Notably, the analysis in section 3.3.4 is supported.

Together, these results demonstrate that by choosing ℓ and the quadrature as presented in this paper, the error is controlled and the quadrature size is chosen nearly optimally. The accurate error bounds derived in section 3.3 means that we can provide a sharp bound of the total final error of the method and optimize the running time of the method for that prescribed error. A reduction in the quadrature size improves memory usage and suggests an improved running time over similar algorithms.

3.5.2. Multilevel error. The previous section verified the error bounds derived for a single level. In this section, we show that the method performs as expected in the multilevel case as well. We considered two points distributed as in Figure 3.5, at opposite corners of the box. The transfer pass (M2L) is always done at the highest level in the tree (level 2). Then the number of levels is increased, thereby adding

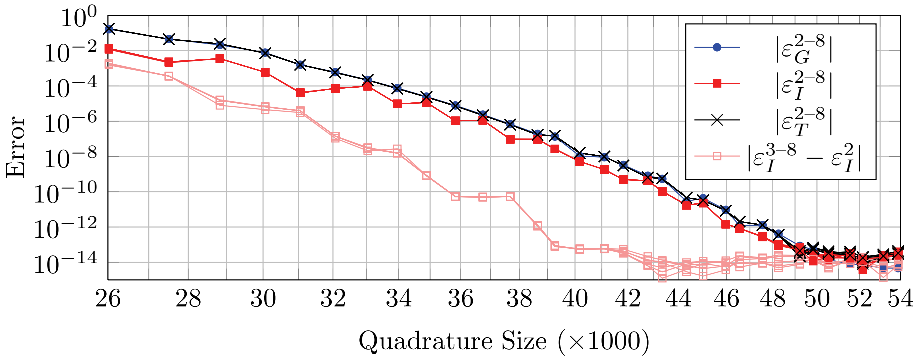

Fig. 3.10. *Log-log plot of multilevel convergence with* $\kappa = 100$*,* $\alpha = 0.8$*, and highest active level box size* $a_2 = 1$*. For each curve, we have* 7 *cases with* $L = 2, \ldots, 8$*. This is denoted by* 2–8 *in the legend of the figure. The* 7 *cases are nearly indistinguishable except when the error is of order* 10^{-12} *and below. The convergence of the numerical quadrature for* (θ, ϕ) *("×" line) should be compared with the convergence of the toy problem in Figure* 3.1 *("•" line, "Fourier filter"). For this calculation, convergence is obtained by simultaneously increasing the order* ℓ *and the number of quadrature points.*

additional translation steps to the calculation. This test therefore checks that the Fourier-based interpolation and anterpolation procedure (section 3.4) does not affect the accuracy of the calculation.

Figure 3.10 shows the convergence of the method as the target error is adjusted, thereby changing ℓ and the quadrature size appropriately. We show ε_G^L (Gegenbauer truncation), ε_T^L (total error), and ε_I^L (interpolation error obtained as the difference between ε_G^L and ε_T^L) for $L = 2, \ldots, 8$ (L is the total number of levels). The quadrature size is the total number of quadrature points at the highest active level, with box size $a_2 = 1$. Note that for $L = 2$ there is only one active level and no translation step. On the same plot, we show the discrepancy between ε_I^L ($L = 3, \ldots, 8$) and ε_I^2 which corresponds to the error due to the upward and downward passes. We expect this error to be much smaller than the target error, since the target error accounts for the M2L operation, and therefore the quadrature is overestimated when considering only the translation functions. This is confirmed by Figure 3.10. The curve ε_G^L is exactly the same for all L, while ε_T^L has small variations with L due to the sampling of the translation function in the upward and downward passes.

We note that the method converges superexponentially, following the rate of decay of the Jacobi–Anger series (3.11).

To further show that the numerical integral is converging to the Gegenbauer series value, Figure 3.11 sets the Gegenbauer error to a constant and uses the target error to increase the quadrature size only. The target error for the Gegenbauer series is set at 10^{-4}, 10^{-6}, 10^{-8}, and 10^{-10}. We observe that as the quadrature size increases, the total error becomes exactly equal to the Gegenbauer series truncation error. For this case, we used two active levels for the multilevel FMM, therefore including the FFT interpolation and anterpolation stages. This validates numerically our theoretical analysis and shows that the error caused by our anterpolation strategy and smoothing of the transfer function $T^s_{\ell,\boldsymbol{r}_0}(\theta, \phi)$ can be effectively controlled and behaves as expected.

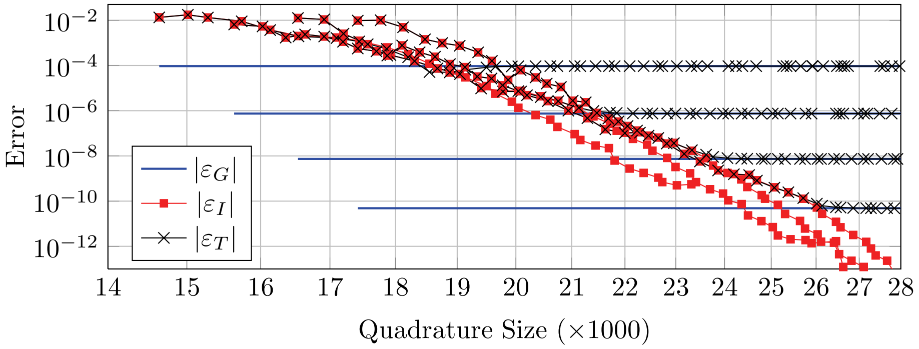

Fig. 3.11. *Log-log plot of two-level convergence to the truncated Gegenbauer series value with* $\kappa = 100$*,* $\alpha = 1/\sqrt{3}$*, and highest active level box size* $a_2 = 1$*. In the four cases, we fix the target error for the Gegenbauer series (*ℓ *is constant) and observe the convergence as the quadrature in* (θ, ϕ) *is increased. The target error for the Gegenbauer series is set at* 10^{-4}*,* 10^{-6}*,* 10^{-8}*, and* 10^{-10}*. The total error* $|\varepsilon_T|$ *converges to the error in the Gegenbauer series* $|\varepsilon_G|$ *once the quadrature is large enough. This demonstrates that the numerical quadrature can be made accurate close to machine accuracy. As in Figure* 3.10*, the convergence of the numerical quadrature for* (θ, ϕ) *("■" lines) should be compared with the convergence of the toy problem in Figure* 3.1 *(blue curve "Fourier filter").*

3.5.3. Speed. As discussed in section 3.4, the Fourier-based FMM uses only FFTs in the upward pass and downward pass to perform the interpolations and anterpolations. FFTs make these steps easier to implement and very efficient. In addition, section 3.3.4 and Figure 3.9 suggest that the quadrature used in this paper is significantly smaller than what is typically used for a spherical harmonic basis.

All times reported are from repeated runs on a 3.2GHz Intel Xeon W3670 processor with 12MB Cache and 24GB of 1333MHz RAM.

First, we illustrate the efficient use of FFTs in the interpolation and anterpolation stages. Here, we compare the FFT interpolation method of section 3.4 against the seminaive exact spherical harmonic interpolation method, which consists of a forward FFT in ϕ, a dense matrix-matrix product on the θ angles, and a backward FFT in ϕ. This spherical harmonics method is analyzed and accelerated in [15, 8]. Figure 3.12 confirms the expected asymptotic running times of each method.

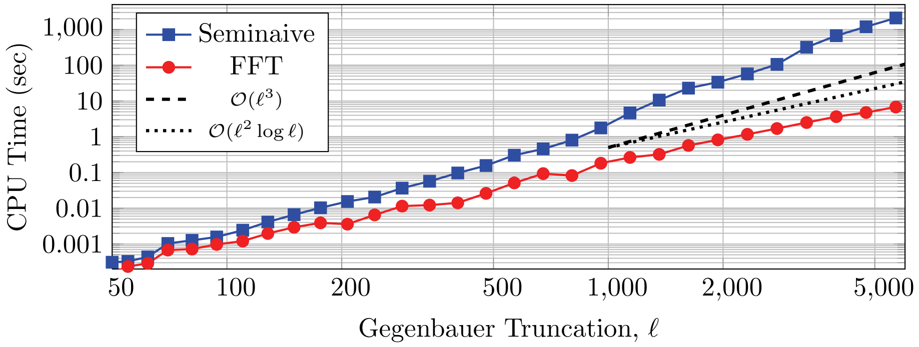

Fig. 3.12. *Comparison of the FFT interpolation scheme of section* 3.4 *(complexity* $\mathcal{O}(\ell^2 \log \ell)$*) with the seminaive* $\mathcal{O}(\ell^3)$ *spherical harmonic transform described in* [15]*. Caching effects can be seen in the seminaive scheme for* $\ell > 1000$*.*

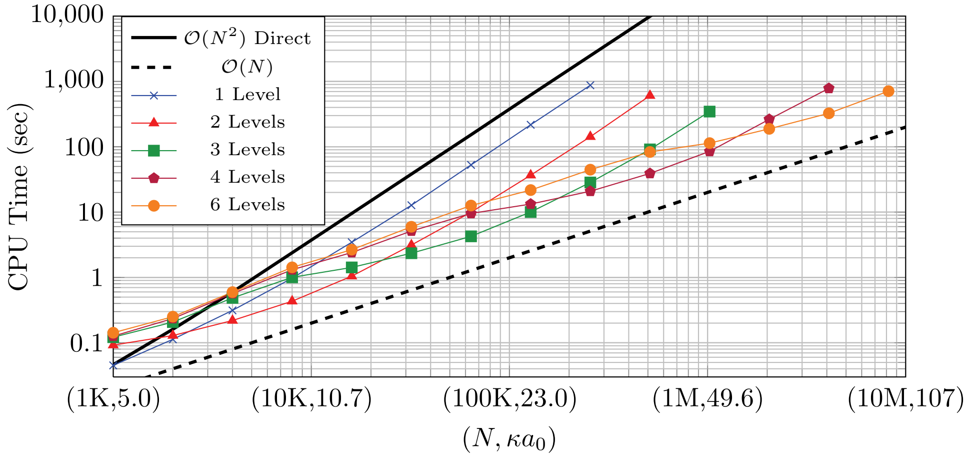

Fig. 3.13. *Average running times of the Fourier-based FMM for a constant number of points per wavelength distributed uniformly in a three-dimensional box. For the volumetric particle distribution, $\kappa \sim N^{1/3}$. By selecting the optimal number of levels, the complexity is $\mathcal{O}(N)$. The test code is not optimized for memory usage and spills into virtual memory for $N > 2M$ with 6 levels.*

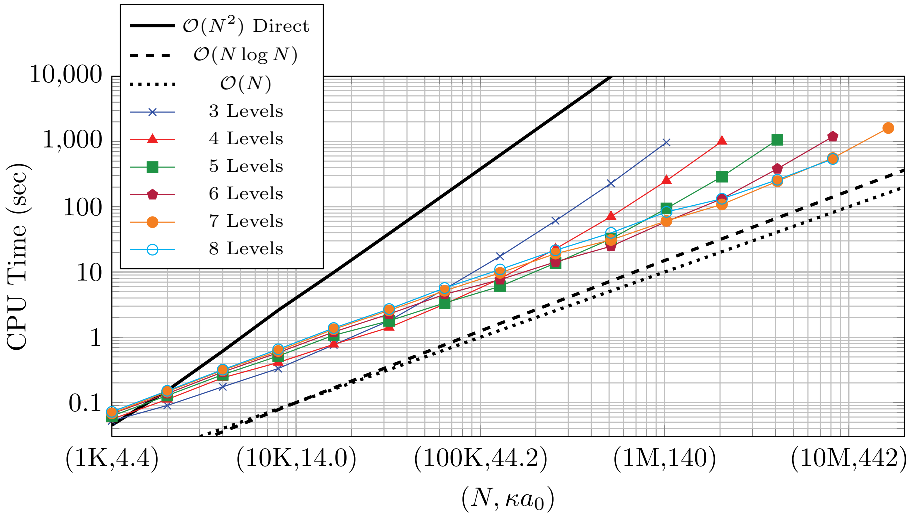

Fig. 3.14. *Average running times of the Fourier-based FMM for constant number of points per wavelength distributed uniformly on the surface of a sphere. For the surface particle distribution, $\kappa \sim N^{1/2}$. By selecting the optimal number of levels, the asymptotic complexity is $\mathcal{O}(N \log^2 N)$.*

To show that the optimal asymptotic running time is achieved, Figures 3.13 and 3.14 show the recorded running times of the Fourier-based FMM and the direct matrix-vector product. The target error is set to 10^{-4} with $\alpha = 1.0$ and was verified for the total relative error in every case the direct product was feasibly computable.

In Figure 3.13 (resp., Figure 3.14), the points are uniformly distributed in a unit cube (on a unit sphere) and the wavenumber κ is scaled with $N^{1/3}$ ($N^{1/2}$) to provide a nearly constant density of points per wavelength as N is varied. As expected, by choosing the correct number of levels the running time is asymptotically $\mathcal{O}(N)$

in the volumetric case. The surface case is more relevant to applications such as the boundary element method, and we find it scales between $\mathcal{O}(N)$ and $\mathcal{O}(N \log N)$ rather than the projected $\mathcal{O}(N \log^2 N)$. The M2M and L2L stages are the only stages that are theoretically $\mathcal{O}(N \log^2 N)$ and, in part because of our use of FFTs, do not represent a sufficiently important portion of the total running time at these scales. Because our code cannot run on parallel clusters, we were not able to scale the problem size beyond what is shown in Figures 3.13 and Figure 3.14.

4. Conclusion. We have proposed using the Fourier basis $e^{\imath p\phi} e^{\imath q\theta}$ in the spherical variables ϕ and θ to represent the far-field approximation in the FMM. By approximating the Helmholtz kernel with (1.1) and using a uniform quadrature we can take advantage of very fast, exact, and well-known FFT interpolation/anterpolation methods. By exploiting symmetries and a scheme to reduce the number of points in the ϕ-direction, the total number of uniform quadrature points required is smaller than the number of Gauss–Legendre quadrature points typically used with spherical harmonics. This is realized by removing the high frequency components of the modified transfer function, $T^{\mathbf{s}}_{\ell,\boldsymbol{r}_0}(\theta, \phi)$, during the precomputation phase which do not significantly contribute to the final integration.

The Fourier-based FMM approach has a number of advantages. Since the interpolation and anterpolation algorithms are exact, the error analysis is simplified; we establish a sharp upper bound for the error. The key parameters are the Gegenbauer truncation parameter ℓ and the quadrature size, in particular the sampling rate in the θ-direction. The truncation error ε_G has been extensively studied by other authors and is well understood. The integration error ε_I accounts for the bandlimited approximation of the modified transfer function and the finite sampling of the plane-waves. This error can be accounted for a priori during the precomputation stage. Numerical tests have confirmed that this error analysis is quite sharp. Constructive algorithms to find nearly optimal parameters were proposed. Since highly efficient FFT algorithms are available in virtually every computing environment, the time-critical interpolation stages of the algorithm are much easier to implement efficiently.

REFERENCES

[1] M. Abramowitz and I. A. Stegun, *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*, Dover, New York, 1964.

[2] Q. Carayol and F. Collino, *Error estimates in the fast multipole method for scattering problems. Part* 1 : *Truncation of the Jacobi-Anger series*, M2AN Math. Model. Numer. Anal., 38 (2004), pp. 371–394.

[3] Q. Carayol and F. Collino, *Error estimates in the fast multipole method for scattering problems. Part* 2 : *Truncation of the Gegenbauer series*, M2AN Math. Model. Numer. Anal., 39 (2004), pp. 183–221.

[4] W. Chew, E. Michielssen, J. M. Song, and J. M. Jin, eds., *Fast and Efficient Algorithms in Computational Electromagnetics*, Artech House, Norwood, MA, 2001.

[5] I. Chowdhury and V. Jandhyala, *Integration and interpolation based on fast spherical transforms for the multilevel fast multipole method*, Microwave Optical Technol. Lett., 48 (2006), pp. 1961–1964.

[6] R. Coifman, V. Rokhlin, and S. Wandzura, *The fast multipole method for the wave equation: A pedestrian prescription*, IEEE Trans. Antennas and Propagation, 35 (1993), pp. 7–12.

[7] E. Darve, *The fast multipole method* I: *Error analysis and asymptotic complexity*, SIAM J. Numer. Anal., 38 (2000), pp. 98–128.

[8] E. Darve, *The fast multipole method: Numerical implementation*, J. Comput. Phys., 160 (2000), pp. 195–240.

[9] J. R. Driscoll and D. Healy, *Computing Fourier transforms and convolutions on the* 2*-sphere*, Adv. in Appl. Math., 15 (1994), pp. 202–250.

[10] A. Dutt, M. Gu, and V. Rokhlin, *Fast algorithms for polynomial interpolation, integration, and differentiation*, SIAM J. Numer. Anal., 33 (1996), pp. 1689–1711.

[11] N. Engheta, W. D. Murphy, V. Rokhlin, and M. S. Vassiliou, *The fast multipole method (FMM) for electromagnetic scattering problems*, IEEE Trans. Antennas and Propagation, 40 (1992), pp. 634–641.

[12] O. Ergul and L. Gurel, *Optimal interpolation of translation operator in multilevel fast multipole algorithm*, IEEE Trans. Antennas and Propagation, 54 (2006), pp. 3822–3826.

[13] M. L. Hastriter, S. Ohnuki, and W. C. Chew, *Error control of the translation operator in* 3*D MLFMA*, Microwave Optical Technol. Lett., 37 (2003), pp. 184–188.

[14] D. Healy, D. Rockmore, P. Kostelec, and S. Moore, *FFTs for the* 2*-sphere: Improvements and variations*, J. Fourier Anal. Appl., 9 (2003), pp. 341–385.

[15] R. Jakob-Chien and B. K. Alpert, *A fast spherical filter with uniform resolution*, J. Comput. Phys., 136 (1997), pp. 580–584.

[16] J. Knab, *Interpolation of bandlimited functions using the approximate prolate series (corresp.)*, IEEE Trans. Inform. Theory, 25 (1979), pp. 717–720.

[17] S. Koc, J. Song, and W. C. Chew, *Error analysis for the numerical evaluation of the diagonal forms of the scalar spherical addition theorem*, SIAM J. Numer. Anal., 36 (1999), pp. 906–921.

[18] R. Kussmaul, *Clenshaw-Curtis quadrature with a weighting function*, Computing, 9 (1972), pp. 159–164.

[19] E. McKay Hyde and O. Bruno, *A fast higher-order solver for scattering by penetrable bodies in three dimension*, J. Comput. Phys., 202 (2005), pp. 236–261.

[20] J. Rahola, *Diagonal forms of the translation operators in the fast multipole algorithm for scattering problems*, BIT, 36 (1996), pp. 333–358.

[21] V. Rokhlin, *Rapid solution of integral equations of scattering theory in two dimensions*, J. Comput. Phys., 86 (1990), pp. 414–439.

[22] V. Rokhlin, *Diagonal forms of translation operators for the Helmholtz equation in three dimensions*, Appl. Comput. Harmon. Anal., 1 (1993), pp. 82–93.

[23] V. Rokhlin and M. Tygert, *Fast algorithms for spherical harmonic expansions*, SIAM J. Sci. Comput., 27 (2006), pp. 1903–1928.

[24] J. Sarvas, *Performing interpolation and anterpolation entirely by fast Fourier transform in the* 3*-D multilevel fast multipole algorithm*, SIAM J. Numer. Anal., 41 (2003), pp. 2180–2196.

[25] N. Sneeuw and R. Bun, *Global spherical harmonic computation by two-dimensional Fourier methods*, J. Geodesy, 70 (1996), pp. 224–232.

[26] R. Suda and M. Takami, *A fast spherical harmonics transform algorithm*, Math. Comp., 71 (2001), pp. 703–715.