Ever since its introduction in the 1980's, the Fast Multipole Method has been capable of producing very high accuracy for an acceptable cost in two dimensions; in three dimensions, it has been considerably less efficient, except when the accuracy requirements were low. This situation changed somewhat with the appearance of a new version of the FMM in [12], which is highly efficient over a wide range of accuracies. That paper introduced a rather involved mathematical apparatus and described the algorithm in its simplest, non-adaptive form. In this paper, we describe an adaptive version of the scheme of [12], applicable to all distributions of particles that are likely to be encountered in practice. The performance of the algorithm is illustrated with numerical examples.

A Fast Adaptive Multipole Algorithm in Three Dimensions

H. Cheng, L. Greengard, V. Rokhlin
Research Report YALEU/DCS/RR-1158
August 3, 1998

The first and third authors were supported in part by DARPA/AFOSR under Contracts F49620-95-C-0075, F49620-97-1-0011. In addition, the third author was supported in part by ONR under Grant N00014-96-1-0188.
The second author was supported in part by the US Department of Energy under Contract DEFGO288ER25053 and in part by DARPA/AFOSR under Contract F49620-95-C-0075.
Approved for public release: distribution is unlimited.
Keywords: *Laplace Equation, Translation Operators, Fast Multipole Method, Adaptive Algorithms.*

A Fast Adaptive Multipole Algorithm in Three Dimensions

H. Cheng, L. Greengard and V. Rokhlin

August 3, 1998

1 Introduction

In [12], a new version of the Fast Multipole Method (FMM) for the evaluation of potential fields in three dimensions was introduced. The scheme evaluates all pairwise interactions in large ensembles of particles, i.e. expressions of the form

$$\Phi(x_j) = \sum_{\substack{i=1 \\ i \neq j}}^{n} \frac{q_i}{\|x_j - x_i\|} \tag{1}$$

for the gravitational or electrostatic potential and

$$E(x_j) = \sum_{\substack{i=1 \\ i \neq j}}^{n} q_i \cdot \frac{x_j - x_i}{\|x_j - x_i\|^3} \tag{2}$$

for the field, where $x_1, x_2, \cdots, x_n$ are points in $\mathbb{R}^3$, and $q_1, q_2, \cdots, q_n$ are a set of (real) coefficients.

The evaluation of expressions of the form (1) is closely related to a number of important problems in applied mathematics, physics, chemistry, and biology. These include molecular dynamics and quantum-mechanical simulations in chemistry, the evolution of large-scale gravitational systems in astrophysics, capacitance and inductance calculations in electrical engineering, and incompressible fluid dynamics. When certain closely related interactions are considered as well, involving expressions of the form

$$\Phi(x_j) = \sum_{\substack{i=1 \\ i \neq j}}^{n} q_i \cdot \frac{e^{i \cdot k \cdot \|x_j - x_i\|}}{\|x_j - x_i\|}, \tag{3}$$

the list of applications becomes even more extensive.

Ever since its introduction in the 1980's, the FMM has been capable of producing very high accuracy for an acceptable cost in two dimensions; in three dimensions, it has been considerably less efficient, except when the accuracy requirements were low. This situation changed somewhat with the development of a new version of the FMM in [12], which is highly efficient over a wide range of accuracies. That paper introduced a rather involved mathematical apparatus and described the algorithm in its simplest, non-adaptive form.

Needless to say, most charge distributions encountered in applications are highly non-uniform, and to be robust, a procedure for the evaluation of sums of the form (1) or (2) has to be adaptive. In this paper, we introduce such a scheme, applicable to all distributions of particles that are likely to be encountered in practice. An additional improvement introduced in this paper is a "compressed" version of translation operators used by the FMM procedure, which is the principal reason for the improvement of the timings found in Section 7 below over those in [12].

The paper is organized as follows. In Section 2, we summarize the mathematical and numerical facts to be used in subsequent sections. In Section 3, we review the analytical apparatus to be used in the design of the improved version of the FMM. Section 4 recapitulates the algorithm of [12], to be used as the starting point in the construction of the scheme of this paper. In Section 5, we describe the adaptive version of the FMM and make some comparisons with tree codes. In Section 6, we illustrate the performance of the method with several numerical examples. Finally, Section 7 discusses several possible generalizations. For a review of FMM-type methods and a more thorough discussion of the literature, we refer the reader to [12].

2 Mathematical preliminaries

In this section, we review the analytical tools used in the design of the FMM algorithm. For a detailed discussion, see [15, 14, 21, 9, 12].

We begin by defining the spherical harmonics of degree n and order m according to the formula

$$Y_n^m(\theta,\phi) = \sqrt{\frac{(n-|m|)!}{(n+|m|)!}} \cdot P_n^{|m|}(\cos\theta)e^{im\phi},. \tag{4}$$

Here, the special functions P_n^m are the associated Legendre functions, which can be defined by the Rodrigues' formula

$$P_n^m(x) = (-1)^m(1-x^2)^{m/2}\frac{d^m}{dx^m}P_n(x),$$

where $P_n(x)$ denotes the Legendre polynomial of degree n.

Theorem 2.1 (Multipole Expansion). *Suppose that N charges of strengths $q_1, q_2, \cdots, q_N$ are located at points $X_1, X_2, ..., X_N$ with spherical coordinates $(\rho_1, \alpha_1, \beta_1)$, $(\rho_2, \alpha_2, \beta_2), \cdots,$ $(\rho_N, \alpha_N, \beta_N)$, respectively. Suppose further that the points $X_1, X_2, ..., X_N$ are located inside a sphere of radius a centered at the origin. Then, for any point $X = (r, \theta, \phi) \in \mathbb{R}^3$ with $r > a$, the potential $\Phi(X)$, generated by the charges $q_1, q_2, \cdots, q_N$, is given by the formula*

$$\Phi(X) = \sum_{n=0}^{\infty}\sum_{m=-n}^{n}\frac{M_n^m}{r^{n+1}} \cdot Y_n^m(\theta,\phi)\ , \tag{5}$$

where

$$M_n^m = \sum_{i=1}^{N} q_i \cdot \rho_i^n \cdot Y_n^{-m}(\alpha_i,\beta_i). \tag{6}$$

Furthermore, for any $p \geq 1$,

$$\left|\Phi(X) - \sum_{n=0}^{p} \sum_{m=-n}^{n} \frac{M_n^m}{r^{n+1}} \cdot Y_n^m(\theta, \phi)\right| \left(\frac{\sum_{i=1}^{N} |q_i|}{r-a}\right) \left(\frac{a}{r}\right)^{p+1} . \tag{7}$$

The preceding theorem describes an efficient representation of the far field due to a collection of sources. Within the FMM, it is also useful to be able to describe the field locally when the charges themselves are far away.

Theorem 2.2 (Local Expansion) *Suppose that* N *charges of strengths* $q_1, q_2, \cdots, q_N$ *are located at the points* $X_1, X_2, \cdots, X_N$ *in* $\mathbb{R}^3$ *with spherical coordinates* $(\rho_1, \alpha_1, \beta_1)$, $(\rho_2, \alpha_2, \beta_2), \cdots,$ $(\rho_N, \alpha_N, \beta_N)$ *respectively. Suppose further that all the points* $X_1, X_2, \cdots, X_N$ *are located outside the sphere* S_a *of radius* a *centered at the origin. Then, for any point* $X \in S_a$ *with coordinates* (r, θ, ϕ), *the potential* $\Phi(X)$ *generated by the charges* $q_1, q_2, \cdots, q_N$ *is described by the local expansion:*

$$\Phi(X) = \sum_{j=0}^{\infty} \sum_{k=-j}^{j} L_j^k \cdot Y_j^k(\theta, \phi) \cdot r^j , \tag{8}$$

where

$$L_j^k = \sum_{l=1}^{N} q_l \cdot \frac{Y_j^{-k}(\alpha_l, \beta_l)}{\rho_l^{j+1}} , \tag{9}$$

with A_n^m *defined by (14). Furthermore, for any* $p \geq 1$,

$$\left|\Phi(X) - \sum_{j=0}^{p} \sum_{k=-j}^{j} L_j^k \cdot Y_j^k(\theta, \phi) \cdot r^{j+1}\right| \leq \left(\frac{\sum_{i=1}^{N} |q_i|}{a-r}\right) \left(\frac{r}{a}\right)^{p+1} . \tag{10}$$

2.1 Translation Operators

The FMM relies on the ability to translate multipole and local expansions. The relevant translation operators are described in the next three theorems [11, 9].

Theorem 2.3 (Translation of a Multipole Expansion) *Suppose that* N *charges of strengths* $q_1, q_2, \cdots, q_N$ *are located inside the sphere* D *of radius* a *centered at* $X_0 = (\rho, \alpha, \beta)$. *Suppose further that for any point* $X = (r, \theta, \phi) \in \mathbb{R}^3 \setminus D$, *the potential due to these charges is given by the multipole expansion*

$$\Phi(X) = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} \frac{O_n^m}{r'^{n+1}} \cdot Y_n^m(\theta', \phi') , \tag{11}$$

where (r', θ', ϕ') *are the spherical coordinates of the vector* $X - X_0$.

Then, for any point $X = (r, \theta, \phi)$ outside a sphere D_1 of radius $(a + \rho)$ centered at the origin,

$$\Phi(X) = \sum_{j=0}^{\infty} \sum_{k=-j}^{j} \frac{M_j^k}{r^{j+1}} \cdot Y_j^k(\theta, \phi) \ , \tag{12}$$

where

$$M_j^k = \sum_{n=0}^{j} \sum_{m=-n}^{n} \frac{O_{j-n}^{k-m} \cdot i^{|k|-|m|-|k-m|} \cdot A_n^m \cdot A_{j-n}^{k-m} \cdot \rho^n \cdot Y_n^{-m}(\alpha, \beta)}{A_j^k} \ , \tag{13}$$

with A_n^m defined by the formula

$$A_n^m = \frac{(-1)^n}{\sqrt{(n-m)! \cdot (n+m)!}} \ . \tag{14}$$

Furthermore, for any $p \geq 1$,

$$\left| \Phi(X) - \sum_{j=0}^{p} \sum_{k=-j}^{j} \frac{M_j^k}{r^{j+1}} \cdot Y_j^k(\theta, \phi) \right| \leq \left(\frac{\sum_{i=1}^{N} |q_i|}{r - (a + \rho)} \right) \left(\frac{a + \rho}{r} \right)^{p+1} \ . \tag{15}$$

Definition 2.1 Formula (13) defines a linear operator converting the multipole expansion coefficients $\{O_j^k\}$ into the multipole expansion coefficients $\{M_j^k\}$. This linear mapping will be denoted by $\mathcal{T}_{MM}$.

Theorem 2.4 (Conversion of a Multipole Expansion to a Local Expansion) *Suppose that N charges of strengths $q_1, q_2, \cdots, q_N$ are located inside the sphere D_{X_0} of radius a centered at the point $X_0 = (\rho, \alpha, \beta)$, and that $\rho > (c + 1)a$ for some $c > 1$. Then the corresponding multipole expansion (11) converges inside the sphere D_0 of radius a centered at the origin. Furthermore, for any point $X \in D_0$ with coordinates (r, θ, ϕ), the potential due to the charges $q_1, q_2, \cdots, q_N$ is described by the local expansion:*

$$\Phi(X) = \sum_{j=0}^{\infty} \sum_{k=-j}^{j} L_j^k \cdot Y_j^k(\theta, \phi) \cdot r^j \ , \tag{16}$$

where

$$L_j^k = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} \frac{O_n^m \cdot i^{|k-m|-|k|-|m|} \cdot A_n^m \cdot A_j^k \cdot Y_{j+n}^{m-k}(\alpha, \beta)}{(-1)^n A_{j+n}^{m-k} \cdot \rho^{j+n+1}} \ , \tag{17}$$

with A_n^m defined by (14). Furthermore, for any $p \geq 1$,

$$\left| \Phi(X) - \sum_{j=0}^{p} \sum_{k=-j}^{j} L_j^k \cdot Y_j^k(\theta, \phi) \cdot r^{j+1} \right| \leq \left(\frac{\sum_{i=1}^{N} |q_i|}{ca - a} \right) \left(\frac{1}{c} \right)^{p+1} \ . \tag{18}$$

Definition 2.2 Formula (17) defines a linear operator converting the multipole expansion coefficients $\{O_j^k\}$ into the local expansion coefficients $\{L_j^k\}$. This linear mapping will be denoted by $\mathcal{T}_{ML}$.

Theorem 2.5 (Translation of a Local Expansion) *Suppose that X_0, X are a pair of points in $\mathbb{R}^3$ with spherical coordinates $(\rho, \alpha, \beta), (r, \theta, \phi)$ respectively, and (r', θ', ϕ') are the spherical coordinates of the vector $X - X_0$ and p is a natural number. Let X_0 be the center of a pth-order local expansion with p finite, its expression at the point X is given by the formula*

$$\Phi(X) = \sum_{n=0}^{p} \sum_{m=-n}^{n} O_n^m \cdot Y_n^m(\theta', \phi') \cdot r'^n \ . \tag{19}$$

Then

$$\Phi(X) = \sum_{j=0}^{p} \sum_{k=-j}^{j} L_j^k \cdot Y_j^k(\theta, \phi) \cdot r^j \ , \tag{20}$$

everywhere in $\mathbb{R}^3$, with

$$L_j^k = \sum_{n=j}^{p} \sum_{m=-n}^{n} \frac{O_n^m \cdot i^{|m|-|m-k|-|k|} \cdot A_{n-j}^{m-k} \cdot A_j^k \cdot Y_{n-j}^{m-k}(\alpha, \beta) \cdot \rho^{n-j}}{(-1)^{n+j} \cdot A_n^m}, \tag{21}$$

and A_n^m are defined by (14).

Definition 2.3 Formula (21) defines a linear operator converting the local expansion coefficients $\{O_n^m\}$ into the local expansion coefficients $\{L_n^m\}$. This linear mapping will be denoted by $\mathcal{T}_{LL}$.

Remark 2.1 The matrices representing the linear operators $\mathcal{T}_{MM}$, $\mathcal{T}_{ML}$, and $\mathcal{T}_{LL}$ are dense, so that applying them to truncated expansions with $O(p^2)$ coefficients costs $O(p^4)$ operations. This is one of principal reasons for the relatively high CPU time requirements of most existing FMM implementations in three dimensions. Section 3 of this paper provides tools for the rapid application of the operators $\mathcal{T}_{MM}, \mathcal{T}_{ML}, \mathcal{T}_{LL}$ to arbitrary vectors, improving the efficiency of FMM algorithms significantly.

2.2 Rotation Operators

In this subsection, we introduce operators which transform multipole and local expansions under rotations of the coordinate system. These operators will play a role in Section 3. The basic results are contained in the next two theorems, whose proofs can be found in [3], together with formulae for the evaluation of the coefficients $R_n^{m,m'}$ in (22), (23).

Theorem 2.6 (Rotation of Multipole Expansions) *Suppose that* (e_1, e_2, e_3) *are the three standard orthonormal basis vectors in* $\mathbb{R}^3$*, given by the formulae*

$$\begin{aligned} e_1 &= (1,0,0), \\ e_2 &= (0,1,0), \\ e_3 &= (0,0,1), \end{aligned}$$

and $(\omega_1, \omega_2, \omega_3)$ *are three other orthonormal verctors in* $\mathbb{R}^3$*, forming another basis.*

Suppose further that a harmonic function $\Phi : \mathbb{R}^3 \setminus \{0\} \mapsto \mathbb{R}$ *is defined by the formula*

$$\Phi(X) = \sum_{n=0}^{p} \sum_{m=-n}^{n} \frac{M_n^m}{r^{n+1}} \cdot Y_n^m(\theta, \phi),$$

with (r, θ, ϕ) *the spherical coordinates of the point* $X \in \mathbb{R}^3$ *associated with the basis* (e_1, e_2, e_3)*. Then, there exist coefficients* $R_n^{m,m'}$ *with* $n = 0, 1, \cdots, p$*,* $m = -n, ..., n$*,* $m' = -n, ..., n$*, such that for any* $X \in \mathbb{R}^3$*,*

$$\Phi(X) = \sum_{n=0}^{p} \sum_{m'=-n}^{n} \frac{\tilde{M}_n^{m'}}{r^{n+1}} \cdot Y_n^{m'}(\theta', \phi'),$$

where (r, θ', ϕ') *are spherical coordinates of* X *in the system of coordinates associated with the basis* $(\omega_1, \omega_2, \omega_3)$*, and*

$$\tilde{M}_n^{m'} = \sum_{m=-n}^{n} R_n^{m,m'} \cdot M_n^m, \tag{22}$$

for all $n = 0, 1, ..., p$*,* $m' = -n, ..., n$*.*

Theorem 2.7 (Rotation of Local Expansions) *Under the conditions of Theorem 2.6, suppose that a harmonic function* $\Phi : \mathbb{R}^3 \mapsto \mathbb{R}$ *is defined by the formula*

$$\Phi(X) = \sum_{n=0}^{p} \sum_{m=-n}^{n} L_n^m \cdot r^{n+1} \cdot Y_n^m(\theta, \phi),$$

where (r, θ, ϕ) *are the spherical coordinates of the point* $X \in \mathbb{R}^3$ *associated with the basis* (e_1, e_2, e_3)*. Then for any* $X \in \mathbb{R}^3$*,*

$$\Phi(X) = \sum_{n=0}^{p} \sum_{m'=-n}^{n} \tilde{L}_n^{m'} \cdot r^{n+1} \cdot Y_n^{m'}(\theta', \phi'),$$

where (r, θ', ϕ') *are spherical coordinates of* X *in the system of coordinates associated with the basis* $(\omega_1, \omega_2, \omega_3)$*, and*

$$\tilde{L}_n^{m'} = \sum_{m=-n}^{n} R_n^{m,m'} \cdot L_n^m, \tag{23}$$

for all $n = 0, 1, ..., p$*,* $m' = -n, ..., n$*. Furthermore, the coefficients* $R_n^{m,m'}$ *are the same as in (22).*

Definition 2.4 Given a rotation $\Omega : \mathbb{R}^3 \mapsto \mathbb{R}^3$, formulae (22), (23) define operators converting the multipole coefficients $\{M_n^m\}$ into the multipole coefficients $\{\tilde{M}_n^m\}$ and the local coefficients $\{L_n^m\}$ into the local coefficients $\{\tilde{L}_n^m\}$, respectively. These two operators are identical, and will be denoted by $\mathcal{R}(\Omega)$.

Remark 2.2 An inspection of formulae (22), (23) shows immediately that the numerical evaluation of the operator $\mathcal{R}(\Omega)$ requires $O(p^3)$ operations.

2.3 Exponential representation

The new generation of FMMs is based on a combination of multipole expansions and exponential or "plane wave" expansions. Given a source point $P = (x_0, y_0, z_0)$ and a target location $Q = (x, y, z)$, with $z > z_0$ and $r = \|P - Q\|$, we begin with the formula [16]

$$\frac{1}{r} = \frac{1}{2\pi} \int_0^\infty e^{-\lambda(z-z_0)} \int_0^{2\pi} e^{i\lambda((x-x_0)\cos\alpha + (y-y_0)\sin\alpha)} d\alpha \, d\lambda. \tag{24}$$

We will construct approximations to the integral in (24) via appropriately chosen quadrature formulae. These quadratures are investigated in detail in [23]; in the following lemma, we simply state the result for three special cases, corresponding to three-digit, six-digit and nine-digit accuracy.

Lemma 2.8 ([23, 12]) *Suppose that $X_0 = (x_0, y_0, z_0)$, $X = (x, y, z)$ are a pair of points in $\mathbb{R}^3$, and that $r = ||X - X_0||$. Suppose further that the coordinates $(x - x_0, y - y_0, z - z_0)$ of the vector $X - X_0$ satisfy the conditions*

$$1 \le z - z_0 \le 4, \quad 0 \le \sqrt{(x-x_0)^2 + (y-y_0)^2} \le 4\sqrt{2}. \tag{25}$$

Then

$$\left| \frac{1}{r} - \sum_{k=1}^{8} \frac{w_k^3}{M_k^3} \sum_{j=1}^{M_k^3} e^{-\lambda_k^3 \cdot [(z-z_0) - i(x-x_0)\cdot\cos(\alpha_{j,k}^3) - (y-y_0)\cdot\sin(\alpha_{j,k}^3)]} \right| < 1.6 \times 10^{-3}, \tag{26}$$

$$\left| \frac{1}{r} - \sum_{k=1}^{17} \frac{w_k^6}{M_k^6} \sum_{j=1}^{M_k^6} e^{-\lambda_k^6 \cdot [(z-z_0) - i(x-x_0)\cdot\cos(\alpha_{j,k}^6) - (y-y_0)\cdot\sin(\alpha_{j,k}^6)]} \right| < 1.3 \times 10^{-6}, \tag{27}$$

$$\left| \frac{1}{r} - \sum_{k=1}^{26} \frac{w_k^9}{M_k^9} \sum_{j=1}^{M_k^9} e^{-\lambda_k^9 \cdot [(z-z_0) - i(x-x_0)\cdot\cos(\alpha_{j,k}^9) - (y-y_0)\cdot\sin(\alpha_{j,k}^9)]} \right| < 1.1 \times 10^{-9}, \tag{28}$$

where $\alpha_{j,k}^3 = 2\pi j / M_k^3$, $\alpha_{j,k}^6 = 2\pi j / M_k^6$, $\alpha_{j,k}^9 = 2\pi j / M_k^9$. The weights $\{w_l^3, l = 1, ..., 8\}$, $\{w_l^6, l = 1, ..., 17\}$, $\{w_l^9, l = 1, ..., 26\}$, the nodes $\{\lambda_l^3, l = 1, ..., 8\}$, $\{\lambda_l^6, l = 1, ..., 17\}$, $\{\lambda_l^9, l = 1, ..., 26\}$ and the integer arrays $\{M_k^3, k = 1, ..., 8\}$, $\{M_k^6, k = 1, ..., 17\}$, $\{M_k^9, k = 1, ..., 26\}$ are given in Tables 13, 14, 15 of the Appendix, respectively.

Remark 2.3 The conditions (25) in the preceding Lemma appear to be rather special. They are, however, related to the geometric refinement of space introduced by the FMM and their use will become clear in the next section.

Remark 2.4 When the desired precision is clear from the context, we will simplify the notation used in Lemma 2.8, writing each of the expressions (26), (27), (28) in the form

$$\left| \frac{1}{r} - \sum_{k=1}^{s(\varepsilon)} \frac{w_k}{M_k} \sum_{j=1}^{M_k} e^{-\lambda_k \cdot (z-z_0)} \cdot e^{i\lambda_k \cdot [(x-x_0)\cdot \cos(\alpha_{j,k}) + (y-y_0)\cdot \sin(\alpha_{j,k})]} \right| < \varepsilon, \tag{29}$$

where the integers $s(\varepsilon)$ and the triplets $\{M_k, w_k, \lambda_k |\ k = 1, \ldots, \varepsilon\}$ all depend on ε, and $\alpha_{j,k} = 2\pi j / M_k$. The total number of exponential basis functions used in (29) will be denoted by

$$S_{exp} = \sum_{k=1}^{s(\varepsilon)} M_k. \tag{30}$$

3 Data Structures and Fast Translation Operators

In order to develop a fast algorithm, we first define the computational domain to be the smallest cube in $\mathbf{R}^3$ containing all sources. We then build a hierarchy of boxes, refining the computational domain into smaller and smaller regions. At refinement level 0, we have a single box corresponding to the entire computational domain. Refinement level $l+1$ is obtained recursively from level l by the subdivision of each box into eight cubic boxes of equal size. In the nonadaptive case, this recursive process is halted after roughly $\log_8 N$ levels, where N is the total number of sources under consideration.

Definition 3.1 A box c is said to be a *child* of box b, if box c is obtained by a single subdivision of box b. Box b is said to be the *parent* of box c.

Definition 3.2 Two boxes are said to be *colleagues* if they are at the same refinement level and share a boundary point. (A box is considered to be a colleague of itself.) The set of colleagues of a box b will be denoted by $Coll(b)$.

Definition 3.3 Two boxes are said to be *well separated* if they are at the same refinement level and are not colleagues.

Definition 3.4 With each box b is associated an *interaction list*, consisting of the children of the colleagues of b's parent which are well separated from box b (Figure 1).

Note that a box can have up to 27 colleagues and that its interaction list contains up to 189 boxes. Figure 1 depicts the colleagues and interaction list of a box in a two-dimensional setting.

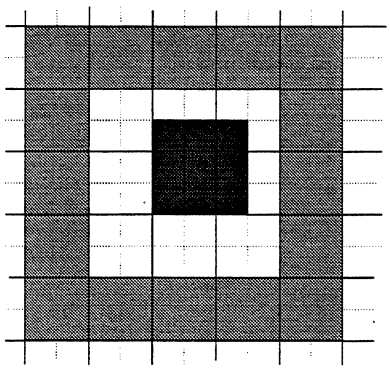

Figure 1: The colleagues of a (two-dimensional) box b are darkly shaded, while its interaction list is indicated in white. In three dimensions, a box b has up to 27 colleagues and its interaction list contains up to 189 boxes.

The interaction list for each box will be further subdivided into six lists, associated with the six coordinate directions $(+z, -z, +y, -y, +x, -x)$ in the three dimensional coordinate system. We will refer to the $+z$ direction as *up*, the $-z$ direction as *down*, the $+y$ direction as *north*, the $-y$ direction as *south*, the $+x$ direction as *east*, and the $-x$ direction as *west*.

Definition 3.5 (Directional lists)

The *Uplist* for a box b consists of those elements of the interaction list which lie *above* b and are separated by at least one box in the $+z$-direction (Fig. 2).

The *Downlist* for a box b consists of those elements of the interaction list which lie *below* b and are separated by at least one box in the $-z$-direction.

The *Northlist* for a box b consists of those elements of the interaction list which lie *north* of b, are separated by at least one box in the $+y$-direction, and are not contained in the Up or Down lists.

The *Southlist* for a box b consists of those elements of the interaction list which lie *south* of b, are separated by at least one box in the $-y$-direction, and are not contained in the Up or Down lists.

The *Eastlist* for a box b consists of those elements of the interaction list which lie *east* of b, are separated by at least one box in the $+x$-direction, and are not contained in the Up, Down, North, or South lists.

The *Westlist* for a box b consists of those elements of the interaction list which lie west of b, are separated by at least one box in the $-x$-direction, and are not contained in the *Up*, *Down*, *North*, or *South* lists.

For any box b, we will denote the number of elements in its *Uplist* by $N(Uplist(b))$, and adopt a similar convention for each of the remain five lists.

Remark 3.1 It is easy to verify that the original interaction list is equal to the union of the

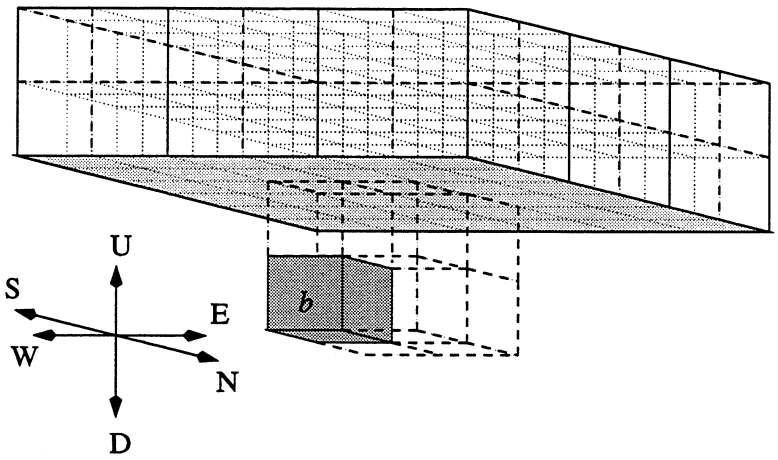

Figure 2: The *Uplist* for the box b (see Definition 3.5).

Up, *Down*, *North*, *South*, *East* and *West* lists. It is also easy to verify for two boxes b, c that

$$\begin{aligned} c \in Uplist(b) &\Leftrightarrow b \in Downlist(c), \\ c \in Northlist(b) &\Leftrightarrow b \in Southlist(c), \\ c \in Eastlist(b) &\Leftrightarrow b \in Westlist(c). \end{aligned} \tag{31}$$

Furthermore, suppose that two boxes b and c are of unit volume and that $c \in Uplist(b)$. Then for any point $X_0 = (x_0, y_0, z_0) \in b$ and any point $X = (x, y, z) \in c$, the vector $X - X_0 = (x - x_0, y - y_0, z - z_0)$ satisfies the inequality

$$1 \leq z - z_0 \leq 4, \;\; 0 \leq \sqrt{(x - x_0)^2 + (y - y_0)^2} \leq 4\sqrt{2}. \tag{32}$$

Note that this is precisely the condition (25) in Lemma 2.8.

Remark 3.2 When there is no danger of confusion, we will use *Uplist(b)* to refer to the geometrical region defined by the *union* of all boxes in the *Uplist* of box b. This is a slight abuse of notation, since *Uplist(b)* is, strictly speaking, a *set* of boxes. We will take the same liberty with *Downlist(b)*, *Northlist(b)*, *Southlist(b)*, *Eastlist(b)*, *Westlist(b)* and $Coll(b)$.

3.1 Rotation Based Translation Operators

In this section, we describe a simple scheme for reducing the cost of applying any of the three operators $\mathcal{T}_{MM}, \mathcal{T}_{ML}, \mathcal{T}_{LL}$ to an arbitrary vector from $O(p^4)$ to $O(p^3)$ operations. The scheme is based on the observation that when a multipole or local expansion is translated along the z-axis, the cost is reduced from $O(p^4)$ to $O(p^3)$ [5, 12, 22]. The following lemma is obtained immediately from the resulting simplification of formulae (13), (17) and (21).

Lemma 3.1 *If, in Theorems 2.3, 2.4 and 2.5, the spherical coordinates of the point X_0 are $(\rho, 0, 0)$, then the formulae (13), (17) and (21) assume the form*

$$M_j^k = \sum_{n=0}^{j} \frac{O_{j-n}^k \cdot A_n^0 \cdot A_{j-n}^k \cdot \rho^n \cdot Y_n^0(0,0)}{A_j^k}, \tag{33}$$

$$L_j^k = \sum_{n=0}^{\infty} \frac{O_n^k \cdot A_n^k \cdot A_j^k \cdot Y_{j+n}^0(0,0)}{(-1)^n A_{j+n}^0 \cdot \rho^{j+n+1}}, \tag{34}$$

$$L_j^k = \sum_{n=j}^{p} \frac{O_n^k \cdot A_{n-j}^0 \cdot A_j^k \cdot Y_{n-j}^0(0,0) \cdot \rho^{n-j}}{(-1)^{n+j} \cdot A_n^k}, \tag{35}$$

respectively.

Definition 3.6 The special cases of the linear operators $\mathcal{T}_{MM}$, $\mathcal{T}_{ML}$, and $\mathcal{T}_{LL}$ defined by the formulae (33), (34), and (35) will be denoted by $\mathcal{T}_{MM}^z$, $\mathcal{T}_{ML}^z$, and $\mathcal{T}_{LL}^z$ respectively.

Observation 3.3 (Rotation Based Translation Operators) Inspection of formulae (33), (34), (35) indicates that each of the operators $\mathcal{T}_{MM}^z$, $\mathcal{T}_{ML}^z$ and $\mathcal{T}_{LL}^z$ can be applied numerically to an arbitrary pth-order expansion for a cost proportional to p^3. Thus, a translation operator can be applied to an arbitrary vector for a cost proportional to p^3 via the following procedure. First, the system of coordinates is rotated so that the new z-axis points to the desired translation center. Then, the expansion is translated via one of the formulae (33), (34) and (35). Finally, the translated expansion is rotated back to the original system of coordinates. Since each of the three stages costs $O(p^3)$ operations, the cost of the whole process has also been reduced to $O(p^3)$ operations. Formally, the scheme we have outlined corresponds to the factorizations

$$\mathcal{T}_{MM} = \mathcal{R}(\Omega^{-1}) \circ \mathcal{T}_{MM}^z \circ \mathcal{R}(\Omega), \tag{36}$$

$$\mathcal{T}_{ML} = \mathcal{R}(\Omega^{-1}) \circ \mathcal{T}_{ML}^z \circ \mathcal{R}(\Omega), \tag{37}$$

$$\mathcal{T}_{LL} = \mathcal{R}(\Omega^{-1}) \circ \mathcal{T}_{LL}^z \circ \mathcal{R}(\Omega), \tag{38}$$

where $\mathcal{R}(\Omega)$ is defined in section 2.2 and $\mathcal{R}(\Omega^{-1})$ denotes the inverse rotation operator.

3.2 Plane Wave Based Translation Operators

In three-dimensional fast multipole schemes, the operator $\mathcal{T}_{ML}$ (converting multipole expansions into local ones) tends to be applied much more frequently then the operators $\mathcal{T}_{MM}, \mathcal{T}_{LL}$ which shift multipole and local expansions. Ignoring boundary effects, one ends up applying $\mathcal{T}_{ML}$ to the multipole expansion for each box about 189 times when the charge distribution is uniform. The operators $\mathcal{T}_{MM}, \mathcal{T}_{LL}$, on the other hand, are applied roughly once per box. In the algorithm of this paper, the operators $\mathcal{T}_{MM}, \mathcal{T}_{LL}$ are applied via the order p^3 scheme described in the preceding section; $\mathcal{T}_{ML}$ is applied by means of a much more complicated procedure, involving the plane wave representation introduced in on Lemma 2.8 of section 2.3.

The following observation provides an expansion of the form (29) for the potential generated by a collection of charges. It is an immediate consequence of Lemma 2.8.

Observation 3.4 Suppose that N charges of strengths $q_1, q_2, ..., q_N$ are located at points $X_1, X_2, \cdots, X_N$ in $\mathbb{R}^3$ with Cartesian coordinates $(x_1, y_1, z_1), (x_2, y_2, z_2), ..., (x_N, y_N, z_N)$, respectively. Suppose further that all points $X_1, X_2, ..., X_N$ are inside a cubic box b with unit volume centered at the origin and that $X = (x, y, z) \in \mathbb{R}^3$ such that $X \in Uplist(b)$. Let $\Phi(X)$ denote the potential generated by the charges $q_1, q_2, ..., q_N$ and let Ψ_ε be defined by the formula

$$\Psi_\varepsilon(X) = \sum_{k=1}^{s(\varepsilon)} \sum_{j=1}^{M_k} W(k,j) \cdot e^{-\lambda_k z} \cdot e^{i\lambda_k \cdot (x \cdot \cos(\alpha_{j,k}) + y \cdot \sin(\alpha_{j,k}))}, \tag{39}$$

with the coefficients $W(k,j)$ given by the formula

$$W(k,j) = \frac{w_k}{M_k} \sum_{l=1}^{N} q_i \cdot e^{\lambda_k z_l} \cdot e^{-i\lambda_k \cdot (x_l \cdot \cos(\alpha_{j,k}) + y_l \cdot \sin(\alpha_{j,k}))}, \tag{40}$$

for all $k = 1, ..., s(\varepsilon), j = 1, ..., M_k$. Then, if $A = \sum_{l=1}^{N} |q_l|$, we have the estimate

$$|\Phi(X) - \Psi_\varepsilon(X)| < A\,\varepsilon. \tag{41}$$

Observation 3.5 A somewhat involved analysis shows that, under the conditions of the preceding observation, $s(\varepsilon) \sim p$, where p is chosen according to (7) to achieve the same accuracy using a multipole expansion. Likewise, the total number of exponential basis functions S_{exp} in (39) is of the same order as the total number of multipole moments (p^2) in (7) in order that the two expansions provide the same precision ε.

Expansions of the form (39) will be referred to as *exponential expansions.* Their main utility is that translation takes a particularly simple form.

Theorem 3.2 (Diagonal translation) *Suppose that a function $\Psi_\varepsilon(X) : \mathbb{R}^3 \mapsto \mathbb{C}$ is defined by the formula (39), which we view as an expansion centered at the origin for $X = (x, y, z)$. Then, for any vector $X_0 = (x_0, y_0, z_0) \in \mathbb{R}^3$, we have the shifted expansion*

$$\Psi_\varepsilon(X) = \sum_{k=1}^{s(\varepsilon)} \sum_{j=1}^{M_k} V(k,j) \cdot e^{-\lambda_k (z - z_0)} \cdot e^{i\lambda_k \cdot ((x - x_0) \cdot \cos(\alpha_{j,k}) + (y - y_0) \cdot \sin(\alpha_{j,k}))}, \tag{42}$$

where

$$V(k,j) = W(k,j) \cdot e^{-\lambda_k z_0} \cdot e^{i\lambda_k \cdot (x_0 \cdot \cos(\alpha_{j,k}) + y_0 \cdot \sin(\alpha_{j,k}))}, \tag{43}$$

for $k = 1, ..., s(\varepsilon),\ j = 1, ..., M_k$.

Definition 3.7 Formula (43) defines a linear operator mapping the coefficients $\{W(k,j)\}$ to the coefficients $\{V(k,j)\}$. This linear operator will be denoted by $\mathcal{D}_{exp}$.

The operator $\mathcal{D}_{exp}$ provides a tool for translating expansions of the form (39) at a cost of $O(S_{exp}) \sim O(p^2)$ operations. In FMM algorithms, however, it is convenient to be able to use multipole and local expansions. Thus, in order to be able to use the operator $\mathcal{D}_{exp}$, linear operators converting multipole expansions into exponential expansions and exponential expansions into local expansions have to be constructed. The following two theorems provide such operators.

Theorem 3.3 *Suppose that N charges of strengths $q_1, q_2, \cdots, q_N$ are located inside a box b of volume d^3 centered at the origin, ε is a positive real number and p is an integer such that for any point $X \in$ Uplist(b) with spherical coordinates (r, θ, ϕ), the potential $\Phi(X)$ generated by the charges $q_1, q_2, \cdots, q_N$ satisfies the inequality*

$$\left| \Phi(X) - \sum_{n=0}^{p} \sum_{m=-n}^{n} \frac{O_n^m}{r^{n+1}} \cdot Y_n^m(\theta, \phi) \right| < \varepsilon. \tag{44}$$

Then

$$\left| \Phi(X) - \sum_{k=1}^{s(\varepsilon)} \sum_{j=1}^{M_k} W(k,j) \cdot e^{-(\lambda_k/d)\cdot z} \cdot e^{i(\lambda_k/d)\cdot(x\cdot\cos(\alpha_{j,k})+y\cdot\sin(\alpha_{j,k}))} \right| < (A/d+1)\cdot\varepsilon, \tag{45}$$

where (x, y, z) are the Cartesian coordinates of X, $A = \sum_{l=1}^{N} |q_l|$, and

$$W(k,j) = \frac{w_k/d}{M_k} \sum_{m=-p}^{p} (-i)^{|m|} \cdot e^{im\cdot\alpha_{j,k}} \sum_{n=|m|}^{p} \frac{O_n^m}{\sqrt{(n-m)!(n+m)!}} (\lambda_k/d)^n, \tag{46}$$

for $k = 1, ..., s(\varepsilon)$, $j = 1, ..., M_k$.

Definition 3.8 Formula (46) defines a linear operator converting the coefficients $\{O_n^m\}$ into the coefficients $\{W(k,j)\}$. This linear mapping will be denoted by C_{MX}.

Theorem 3.4 *Suppose that N charges of strengths $q_1, q_2, \cdots, q_N$ are located inside a box b of volume d^3 centered at the origin, ε is a positive real number, and that for any point $X = (x, y, z) \in$ Uplist(b), the potential $\Phi(X)$ generated by the charges $q_1, q_2, \cdots, q_N$ satisfies the inequality*

$$\left| \Phi(X) - \sum_{k=1}^{s(\varepsilon)} \sum_{j=1}^{M_k} W(k,j) \cdot e^{-(\lambda_k/d)\cdot z} \cdot e^{i(\lambda_k/d)\cdot(x\cdot\cos(\alpha_{j,k})+y\cdot\sin(\alpha_{j,k}))} \right| < (A/d)\cdot\varepsilon, \tag{47}$$

where $A = \sum_{l=1}^{N} |q_l|$. Then there exists an integer p, such that

$$\left| \Phi(X) - \sum_{n=0}^{p} \sum_{m=-n}^{n} L_n^m \cdot Y_n^m(\theta, \phi) \cdot r^n \right| < (A/d+1)\cdot\varepsilon, \tag{48}$$

where (r, θ, ϕ) are the spherical coordinates of X and

$$L_n^m = \frac{(-i)^{|m|}}{\sqrt{(n-m)!(n+m)!}} \sum_{k=1}^{s(\varepsilon)} (-\lambda_k/d)^n \sum_{j=1}^{M_k} W(k,j) \cdot e^{im\cdot\alpha_{j,k}}, \tag{49}$$

for $n = 0, ..., p$, $m = -n, ..., n$.

Definition 3.9 Formula (49) defines a linear operator converting the coefficients $\{W(k,j)\}$ into the coefficients $\{L_n^m\}$. This linear mapping will be denoted by $\mathcal{C}_{XL}$.

Remark 3.6 It is easy to see that (46) can be evaluated numerically for $k = 1, ..., s(\varepsilon), j = 1, ..., M_k$, at a cost proportional to p^3. Indeed, we first calculate $(2p+1) \cdot s(\varepsilon)$ quantities $F_{k,m}$ defined by the formula

$$F_{k,m} = \sum_{n=|m|}^{p} \frac{O_n^m}{\sqrt{(n-m)!(n+m)!}} (\lambda_k/d)^n, \tag{50}$$

for $k = 1, ..., s(\varepsilon), m = -p, ..., p$. This step requires $O(s(\varepsilon) \cdot p^2)$ operations. We then evaluate the coefficients $W(k,j)$ via the formula

$$W(k,j) = \frac{w_k/d}{M_k} \sum_{m=-p}^{p} (-i)^{|m|} \cdot e^{im \cdot \alpha_{j,k}} \cdot F_{k,m}, \tag{51}$$

for $k = 1, ..., s(\varepsilon)$, $j = 1, ..., M_k$, at a cost of $O(S_{exp} \cdot p)$ operations. Thus, the total cost of applying the operator $\mathcal{C}_{MX}$ numerically to a pth-order multipole expansion is

$$Cost(\mathcal{C}_{MX}) \sim O(p^2 s(\varepsilon) + p S_{exp}) \sim O(p^3), \tag{52}$$

making use of Observation 3.5. A similar argument shows that the operator $\mathcal{C}_{XL}$ can also be evaluated numerically for a cost proportional to p^3.

The proofs of Theorems 3.2, 3.3, 3.4 can be found in [12]. The following observation follows immediately from Theorems 3.2, 3.3 and 3.4.

Observation 3.7 (Multipole to local translation for the Uplist) Suppose that b, c are two boxes such that c is in the *Uplist* of b. Then the translation operator $\mathcal{T}_{ML}$ which converts a multipole expansion centered in b to a local expansion centered in c can be applied via the following procedure. First, convert the multipole expansion centered in b into an exponential expansion via the operator $\mathcal{C}_{MX}$; then, use the operator $\mathcal{D}_{exp}$ to translate the resulting exponential expansion to the center of box c; finally, convert the latter expansion into a local expansion in box c via the operator $\mathcal{C}_{XL}$. In short,

$$\mathcal{T}_{ML} = \mathcal{C}_{XL} \circ \mathcal{D}_{exp} \circ \mathcal{C}_{MX}. \tag{53}$$

Observation 3.8 (Multipole to local translation: general case) The decomposition (53) of the operator $\mathcal{T}_{ML}$ is valid only when box c is in the *Uplist* of box b. When box c is not in the *Uplist* of box b, the operator $\mathcal{T}_{ML}$ can easily be applied by first rotating the system of coordinates, so that in the new coordinate system, box c lies in the *Uplist* of box b, applying the operator $\mathcal{T}_{ML}$ via (53) to the rotated expansion, and finally rotating back to the original system of coordinates. Formally, this corresponds to the factorization

$$\mathcal{T}_{ML} = \mathcal{R}(\Omega^{-1}) \circ \mathcal{C}_{XL} \circ \mathcal{D}_{exp} \circ \mathcal{C}_{MX} \circ \mathcal{R}(\Omega). \tag{54}$$

The rotation operators $\mathcal{R}(\Omega)$ are described in section 2.2.

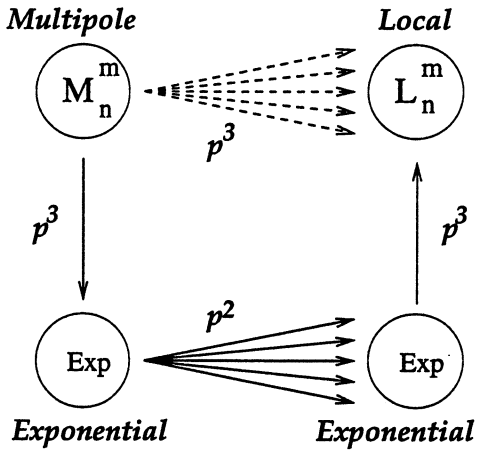

Figure 3: A large number of multipole-to-local translations, each costing $O(p^3)$ operations are replaced by a single multipole-to-exponential operator costing $O(p^3)$ operations, a large number of exponential translations costing $O(p^2)$ operations, and a single exponential-to-local operator costing $O(p^3)$ operations.

Remark 3.9 As mentioned earlier, application of the translation operators $\mathcal{T}_{ML}$ is a dominant part of FMM algorithms, occurring up to 189 times per box. Naive application of these operators results in a cost of roughly $189 \cdot p^4$ operations per box, which is prohibitively expensive in most cases. Fast rotation-based schemes [5, 22, 12] use Observation 3.3 to reduce the cost to roughly $189 \cdot 3 \cdot p^3$ operations per box; the resulting FMM schemes are fairly efficient in low-precision applications. Theorems 3.2, 3.3, 3.4 of this subsection can be used to reduce the cost of application of the operators $\mathcal{T}_{ML}$ to approximately $20 \cdot p^3 + 189 \cdot p^2$ operations per box. Indeed, in order to account for the interaction of box b with its *Uplist* boxes, we use the operator $\mathcal{C}_{MX}$ of Theorem 3.3 to convert b's multipole expansion into an exponential one for a cost proportional to p^3. We then use the operator $\mathcal{D}_{exp}$ of Theorem 3.2 to translate the resulting exponential expansion to each of the boxes in *Uplist(b)*, for a cost propotional to $N(\mathit{Uplist}(b)) \cdot p^2$. Subsequently, we convert the accumulated exponential expansion for each box into a local one via the operator $\mathcal{C}_{XL}$ of Theorem 3.4, for a cost proportional to p^3. This procedure is illustrated in Figure 3. The analogous process must, of course, be repeated for the *Downlist, Northlist, Southlist, Eastlist*, and *Westlist*. For the *Northlist, Southlist, Eastlist*, and *Westlist* (but not for the *Downlist*), there is an additional cost proportional to $2 \cdot p^3$ operations per box to rotate the coordinate system, as described in Observation 3.8. The total cost for each of the six interaction lists is summarized in the following

$$\begin{aligned} \mathit{Cost}(\mathit{Uplist}) &\sim 2 \cdot p^3 + N(\mathit{Uplist}(b)) \cdot p^2, \\ \mathit{Cost}(\mathit{Downlist}) &\sim 2 \cdot p^3 + N(\mathit{Downlist}(b)) \cdot p^2, \\ \mathit{Cost}(\mathit{Northlist}) &\sim 4 \cdot p^3 + N(\mathit{Northlist}(b)) \cdot p^2, \\ \mathit{Cost}(\mathit{Southlist}) &\sim 4 \cdot p^3 + N(\mathit{Southlist}(b)) \cdot p^2, \end{aligned} \tag{55}$$

$$\begin{aligned} Cost(Eastlist) &\sim 4 \cdot p^3 + N(Eastlist(b)) \cdot p^2, \\ Cost(Westlist) &\sim 4 \cdot p^3 + N(Westlist(b)) \cdot p^2, \end{aligned}$$

respectively. Combining (55) with the fact that the maximum total number of boxes in the interaction list is 189, we obtain

$$Cost(\mathcal{T}_{ML}) \sim 20 \cdot p^3 + 189 \cdot p^2. \tag{56}$$

Remark 3.10 The procedure of the preceding section has been further accelerated. First, symmetry considerations can be used to reduce number of translations per box from 189 to 40 without any loss of precision. We refer the reader to [12] for details. Second, while the expansions (5) and (8) are expressed in terms of spherical harmonics, they are being used to represent potentials inside or outside of regions that are cubic in shape. Clearly, spherical harmonics are not an optimal basis for this purpose. Special-purpose harmonics have been developed for the representation of potentials in such regions; they have been incorporated in our implementation and the timings presented in Section 6 below reflect this additional improvement. The procedure itself is fairly involved, and will be reported at a later date [6].

4 The non-adaptive FMM

In this section, we describe the non-adaptive FMM algorithm of [12], combining the factorization (54) of the translation operator $\mathcal{T}_{ML}$ with the factorizations (36), (38) of the operators $\mathcal{T}_{MM}, \mathcal{T}_{LL}$. We present it here as a reference for the subsequent adaptive procedure. For details, the reader is referred to the original paper [12].

In the FMM, the set of all boxes at level l is denoted by B_l, with B_0 consisting of the computational box itself. With each box b, we associate fourteen expansions about its center.

- A *multipole expansion* Φ_b of the form (5) represents the potential generated by charges contained inside b; it is valid in $\mathbb{R}^3 \setminus Coll(b)$.
- A *local expansion* Ψ_b of the form (8) represents the potential generated by all charges outside $Coll(b)$; it is valid inside box b.
- Six *outgoing exponential expansions* W_b^{Up}, W_b^{Down}, W_b^{North}, W_b^{South}, W_b^{East}, and W_b^{West} of the form (39), representing the potential generated by all charges located in b and valid in $Uplist(b)$, $Downlist(b)$, $Northlist(b)$, $Southlist(b)$, $Eastlist(b)$, and $Westlist(b)$, respectively.
- Six *incoming exponential expansions* V_b^{Up} V_b^{Down}, V_b^{North}, V_b^{South}, V_b^{East}, and V_b^{West} of the form (39), representing the potential inside b generated by all charges located in $Downlist(b)$, $Uplist(b)$, $Southlist(b)$, $Northlist(b)$, $Westlist(b)$, and $Eastlist(b)$, respectively.

Initialization

Comment [Choose number of refinement levels $\mathbf{NLEV} \approx \log_8 N$, and the order p of the multipole expansions. The number of boxes at the finest level is then $8^{\mathbf{NLEV}}$, and the average number of particles per box is $s = N/(8^{\mathbf{NLEV}})$. Denote the set of all boxes at level l by B_l.]

Upward Pass

Step 1

Do for each box $b \in B_{\mathbf{NLEV}}$,
  Form multipole expansion Φ_b of potential field due to
  particles in box b at b's center, via Theorem 2.1.
End do

Step 2

Do for levels $l = \mathbf{NLEV} - 1, ..., 2$,
  Do for each box $b \in B_l$,
    Form multipole expansion Φ_b about the center of b by
    merging expansions from its eight children via Theorem 2.3.
    (In applying $\mathcal{T}_{MM}$, use the factorization of Observation 3.3.)
  End do
End do

Downward Pass

Initialization

Set $\Psi_b = (0, 0, ..., 0)$ for all boxes.

Step 3A

Do for levels $l = 2, ..., \mathbf{NLEV}$,
  Do for each box $b \in B_l$,
    Form local expansion Ψ_b about the center of b by
    using Theorem 2.5 to shift the local expansion of b's parent to b.
    (In applying $\mathcal{T}_{LL}$, use the factorization of Observation 3.3.)
  End do

Step 3B

Do for $Dir = Up, Down, North, South, East, West$,
  Do for each box $b \in B_l$,
    Convert the multipole expansion Φ_b to the
    "outgoing" exponential W_b^{Dir}, via Theorem 3.3.
    Do for each box $c \in Dir - list(b)$,
      Translate W_b^{Dir} from b to c via Theorem 3.2 and add to V_c^{Dir}.
    End do
  End do
  Do for each box $c \in B_l$,
    Convert the incoming exponential V_c^{Dir} to the
    local expansion Ψ_c, via Theorem 3.4.
  End do
End do
End do

Step 4

Do for each box $b \in B_{\mathbf{NLEV}}$,
  For each particle in box b, evaluate Ψ_b at the particle position.
End do

Step 5

Do for each box $b \in B_{\mathbf{NLEV}}$,
  For each particle in box b,
  compute interactions with particles in b's colleagues directly.
End do

---

5 The adaptive FMM

The preceding algorithm is efficient for reasonably uniform distributions of particles, but its performance deteriorates significantly for non-uniform distributions. To remedy this situation, we construct an adaptive version of the scheme. Our strategy follows closely that used in [4] for the two dimensional case. Starting with the computational box, we build our structure recursively. If the box under consideration contains no charges, its existence is immediately forgotten. If it contains fewer than s charges (where s is an appropriately chosen positive integer), it is not subdivided further and considered *childless*. Otherwise, it is considered a *parent box* and subdivided into its eight children. The procedure is then repeated for each of the latter. As in the nonadaptive case, the set of all nonempty boxes at level l is denoted by B_l, with B_0 consisting of the computational box itself.

5.1 Adaptive lists

In order to describe the adaptive scheme, we will need the following additional lists.

Definition 5.1 *List 1* of a childless box b, denoted by $L_1(b)$, is defined to be the set consisting of b and all childless boxes adjacent to b. If b is a parent box, its List 1 is empty.

Definition 5.2 *List 2* of a box b, denoted by $L_2(b)$, is the set consisting of all children of the colleagues of b's parent that are well separated from b.

Definition 5.3 *List 3* of a childless box b, denoted by $L_3(b)$, is the set consisting of all descendents of b's colleagues that are not adjacent to b, but whose parent boxes are adjacent to b. If b is a parent box, its list 3 is empty.

Note that any box c in $L_3(b)$ is smaller than b and is separated from b by a distance not less than the side of c, and not greater than the side of b.

Definition 5.4 *List 4* of a box b, denoted by $L_4(b)$, consists of boxes c such that $b \in L_3(c)$; in other words, $c \in L_4(b)$ if and only if $b \in L_3(c)$.

Note that all boxes in $L_4(b)$ are childless and are larger than b.

Figure 4 shows the four lists for a box b in two dimensions. Of these, *List 1* and *List 2* have simple analogues in the non-adaptive algorithm of Section 4. Specifically, *List 1* of some finest level box b would consist of its colleagues, whose interactions will be accounted for directly. *List 2* of b would consist of boxes that are of the same size as b and are well separated: i.e., the interaction list of Definition 3.4. Lists 3 and 4 do not have analogues in the non-adaptive scheme.

$L_2(b)$ is subdivided further into *Uplist*(b), *Downlist*(b), *Northlist*(b), *Southlist*(b), *Eastlist*(b), and *Westlist*(b), by obvious analogy with Definition 3.5.

With each box b, we also associate fourteen expansions by analogy with those described in section 4. The only difference is that the *multipole expansion* Φ_b is valid in $\mathbb{R}^3 \setminus (L_1(b) \cup L_3(b))$. Similarly, the *local expansion* Ψ_b represents the potential inside b generated by all charges outside $L_1(b) \cup L_3(b)$.

ADAPTIVE FMM ALGORITHM

Initialization

Choose precision ε and the order of the multipole expansions p. Choose the maximum number s of charges allowed in a childless box. Define B_0 to be the smallest cube containing all sources (the computational domain).

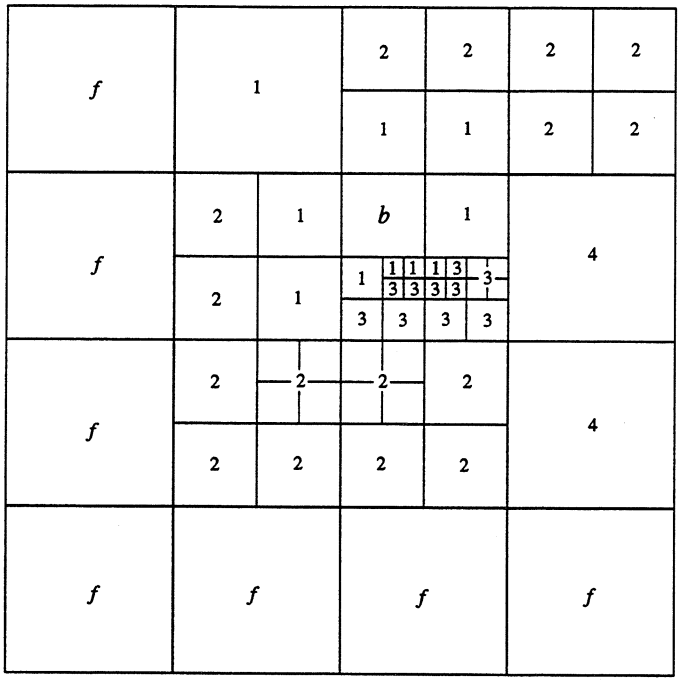

Figure 4: Lists 1-4 for box b

Build Tree Structure

Step 0

Do for levels $l = 0, 1, 2, ...$
  Do for each box $b \in B_l$
    If b contains more than s charges **then**
      Divide b into eight child boxes. Ignore empty children
      and add the nonempty child boxes to B_{l+1}.
    End if
  End do
End do

Comment [Denote the greatest refinement level obtained above by **NLEV** and the total number of boxes created as **NBOX**. Create the four lists for each box.]

Do for each box $b_i, i = 1, 2, ..., \mathbf{NBOX}$
  Create lists $L_1(b_i), L_2(b_i), L_3(b_i), L_4(b_i)$.
  Split $L_2(b_i)$ into *Up, Down, North, South, East, West* lists.
End do

Upward Pass

Comment [During the upward pass, a pth-order multipole expansion is formed for each box b about its center, representing the potential in $\mathbb{R}^3 \setminus (L_1(b) \cup L_3(b))$ due to all charges in b.]

Step 1

Comment [For each childless box b, form a multipole expansion about its center from all charges in b.]

Do for each box $b_i, i = 1, 2, ..., \mathbf{NBOX}$
  If b_i is childless **then**
    Use Theorem 2.1 to form pth-order multipole expansion Φ_{b_i},
    representing the potential in $\mathbb{R}^3 \setminus (L_1(b) \cup L_3(b))$ due to all charges in b_i.
  End if
End do

Step 2

Comment [For each parent box, form a multipole expansion about its center by merging multipole expansions from its children.]

Do for levels $l = \mathbf{NLEV} - 1, \mathbf{NLEV} - 2, ..., 0$
  Do for each box $b \in B_l$
    If b is a parent box **then**
      Use the operator $\mathcal{T}_{MM}$ to merge multipole expansions from
      its children into Φ_b.
    End if
  End do
End do

Downward Pass

Comment [During the downward pass, a pth-order local expansion is generated for each box b about its center, representing the potential in b due to all charges outside $(L_1(b) \cup L_3(b))$.]

Step 3

Comment [For each box b, add to its local expansion the contribution due to charges in $L_4(b)$.]

Do for each box $b_i, i = 1, 2, \cdots, \mathbf{NBOX}$
  Do for each box $c \in L_4(b_i)$
    If the number of charges in $b_i \leq p^2$ **then**

Comment [The number of charges in b_i is small. It is faster to use direct calculation than to generate the contribution to the local expansion Ψ_{b_i} due to charges in c; act accordingly.]

Calculate potential field at each particle point in b_i
directly from charges in c.
Else

Comment [The number of charges in b_i is large. It is faster to generate the contribution to the local expansion Ψ_{b_i} due to charges in c than to use direct calculation; act accordingly.]

Generate a local expansion at b_i's center due to
charges in c, and add to Ψ_{b_i}.
End if
End do
End do

Step 4

Comment [For each box b on level l with $l = 2, 3, ..., \mathbf{NLEV}$ and for each direction $Dir = Up, Down, North, South, East, West$, create from box b's multipole expansion the outgoing exponential W_b^{Dir} in direction Dir, using the operator $\mathcal{C}_{MX}$. Translate W_b^{Dir} to the center of each box $c \in Dirlist(b)$ using Corollary 3.2, and add the translated expansions to its incoming exponential expansion V_c^{Dir}.]

Do for levels $l = 2, 3, ..., \mathbf{NLEV}$
Do for $Dir = Up, Down, North, South, East, West$
Do for each box $b \in B_l$
Use the operator $\mathcal{C}_{MX}$ to convert multipole expansion
Φ_b into exponential W_b^{Dir}.
Do for each box $c \in Dirlist(b)$
Translate the outgoing exponential expansion W_b^{Dir} to the center of box c
using the diagonal translation operator $\mathcal{D}_{XX}$, and add the translated
expansion to the incoming exponential expansion V_c^{Dir}.
End do
End do

Comment [For each box c on level l, convert the exponential expansion V_c^{Dir} into a local expansion and add it to Ψ_c.]

Do for each box $c \in B_l$
Use the operator $\mathcal{C}_{XL}$ to convert the exponential expansion V_c^{Dir}
into a local expansion, and add it to Ψ_c.
End do
End do
End do

Step 5

Comment [For each parent box b, shift the center of its local expansion to its children.]

Do for each box $b_i, i = 1, 2, \cdots,$ **NBOX**
  If b_i is a parent box **then**
    Use the operator $\mathcal{T}_{LL}$ to shift the local expansion Ψ_{b_i} to the centers of its children, and add the translated expansions to children's local expansions.
  End if
End do

Evaluation of Potentials

Step 6

Comment [Include contribution to potential from local expansion at leaf nodes.]

Do for each box $b_i, i = 1, 2, ...,$ **NBOX**
  If b_i is childless **then**
    Calculate the potential at each charge in b_i from the local expansion Ψ_{b_i}.
  End if
End do

Step 7

Comment [Include contribution from direct interactions.]

Do for each box $b_i, i = 1, 2, ...,$ **NBOX**
  If b_i is childless **then**
    Calculate the potential at each charge in b_i
    directly due to all charges in $L_1(b_i)$.
  End if
End do

Step 8

Comment [For each childless box b, evaluate the potential due to all charges in $L_3(b)$.]

Do for each box $b_i, i = 1, 2, ...,$ **NBOX**
  If b_i is childless **then**
    Do for each box $c \in L_3(b_i)$
      If the number of charges in $c \leq p^2$ **then**

Comment [The number of charges in c is small. It is faster to use direct calculation than to evaluate the multipole expansion Φ_c; act accordingly.]

Calculate the potential at each charge in b_i
directly from charges in c.
Else

Comment [The number of charges in c is large. It is faster to evaluate the expansion Φ_c than to use direct calculation; act accordingly.]

Calculate the potential at each charge in b_i
from multipole expansion Φ_c.
End if
End do
End if
End do

Remark 5.1 *Step 3* in the above algorithm could be simplified without increasing the asymptotic CPU time estimate of the latter. Specifically, we could always generate the contribution to the local expansion Ψ_b due to charges in c, even when the number of charges in c is small. However, the actual computation time would increase somewhat. A similar observation can be made about *Step 8* of the above algorithm.

Remark 5.2 In the actual implementation of the adaptive algorithm, we have introduced several minor modifications, designed primarily to reduce the memory requirements of the scheme. In particular, *Steps 3, 4*, and *5* of the downward pass have been combined to eliminate some of the intermediate storage.

5.2 Complexity Analysis and Comparison with Tree Codes

The cost of the FMM algorithm of this paper (like the cost of older schemes of this type) can be separated into two parts. The first part concerns the construction of the data structure (*Step 0*); the second part concerns the calculation of the potentials.

If N denotes the total number of particles in the system, the CPU time estimate for the first part is $O(N \log N)$ in the general case and $O(N)$ for reasonably uniform distributions of particles, where "bin sorting" can be used instead of the recursive procedure outlined above. The CPU time requirements for the second part are $O(N)$ in all cases. In practice, however, the first part uses a negligible proportion of the total CPU time.

There has been some confusion in the literature concerning computational complexity, partly because of an erroneous proof in the original paper [4] addressing the two dimensional case. A correct proof can be found in [17], under very general assumptions about the distribution of charges. We omit the detailed analysis of the asymptotic time and storage estimates for the algorithm of this paper since it does not differ materially from that in [17]. For reasonably

uniform distributions, it is easy to see that the asymptotic cost of the nonadaptive algorithm is approximately

$$27N\,s + 2\,N\,p^2 + 189\,\frac{N}{s}\,p^2 + 20\frac{N}{s}\,p^3,$$

where s is the number of charges per box at the finest level. The first term comes from direct interactions with colleagues, the second comes from forming and evaluating multipole and local expansions at the finest level, and the last two come from multipole-to-local translations, as shown in (56). Using symmetry considerations, it is possible to reduce the factor 189 to 40 (see Remark 3.10 above). Setting $s \approx p^{3/2}$, we see that the work required by the nonadaptive FMM is of the order

$$O(Np^{3/2}).$$

Similarly, the storage costs are of the order

$$O(\frac{N}{s}\,p^2) \sim O(N\,p^{3/2}).$$

In the adaptive case, precise estimates are more involved, but the reader will note in the numerical examples below that both CPU times and storage requirements are at a maximum for the most homogenous distributions.

A second area where there has been some confusion concerns comparisons of the FMM with what are generally known as "tree codes." These were introduced independently of the FMM by Barnes and Hut [2]. (A related scheme by Appel [1] is more like the FMM than like a tree code.) In tree codes, all interactions are computed by either direct calculation or by evaluation of a multipole expansion for a source box at a well-separated target position. Within the FMM, however, one has four options for a source box b and a target box c:

1. compute interactions directly,
2. evaluate the multipole expansion for b at individual targets in c directly,
3. convert the field due to each source in b to a local expansion in c (which is later evaluated),
4. convert the multipole expansion in b to a local expansion in c (which is later evaluated).

A properly implemented FMM always selects the least expensive option (which is trivial to choose); thus, it is always more efficient than a tree code. We omitted this decision analysis in our original descriptions of the FMM [10, 11, 18] in order to focus on the central result, which is option 4 above. It is this option which reduces the cost to $O(N)$. It is easy to see that options 2 and 3 are appropriate only in Steps 3 and 8 above, when considering Lists 3 and 4. The analogues of Steps 3 and 8 here are Stages 5 and 6 in [4].

6 Numerical Results

The algorithm described in Section 5 has been implemented in Fortran 77, and numerical experiments have been carried out for a variety of charge distributions using a Sun UltraSPARC workstation with a CPU clock rate of 167 MHz. The results of our experiments are summarized in Tables 1-12, with all timings given in seconds.

In the first set of our experiments, the charges were distributed randomly but uniformly in the cube $[-0.5, 0.5] \times [-0.5, 0.5] \times [-0.5, 0.5]$; results are reported in Tables 1-3. In the second set, the charges were distributed randomly in the polar angles θ and ϕ on the surface of a sphere of radius 0.5, centered at the origin. Obviously, such a distribution is concentrated at the poles (Figure 5); results are reported in Tables 4-6. In the third set, the charges were distributed on the surface of a cylinder with height 1.0 and radius 0.05 (Figure 6); results are reported in Tables 7-9. In the final set of experiments, the charges were distributed on a complicated surface shown in Figure 7. The results for this configuration are reported in Tables 10-12. In all our experiments, the charge strengths were taken randomly from the interval $(-0.5, 0.5)$.

For each geometry, the numerical tests were performed with three-, six-, and nine-digit accuracy. For three-digit accuracy, the maximum number of charges allowed in a childless box was set to be 40. Corresponding numbers for six- and nine-digit accuracies are 100 and 180, respectively. The timings produced by the adaptive FMM algorithm were compared with those obtained by the direct calculation. Obviously, it was not practical to apply the direct scheme to large-scale ensembles of particles, due to excessive computation times. Thus, the direct algorithm was used to evaluate the potentials at the first 100 elements of the ensemble, and the resulting CPU time was extrapolated. Similarly, the accuracy of the algorithm was calculated at the first 100 particles via formula (57) below.

The tables are organized as follows.

1. The first column lists the number of charges used in the calculation.

2. The second column lists the number of levels used in the multipole hierarchy.

3. The third column lists the order of the multipole expansion used.

4. The fourth column lists the corresponding number of exponential basis functions.

5. The fifth column lists the amount of storage used by the adaptive FMM algorithm. In the three- and six-digit cases, we indicate the number of single precision (REAL*4) words used, while in the nine-digit case, we indicate the number of double precision (REAL*8) words used.

6. Columns six and seven contain the CPU times required by the adaptive FMM and the direct calculation, respectively. In the three- and six-digit cases, both the FMM and the direct calculations were performed in single precision; in the nine-digit case, both calculations were performed in double precision.

7. Column eight lists the L_2 norm of the error in the FMM approximation, which is computed via the formula

$$E = \left(\frac{\sum_{i=1}^{N} |\Phi(x_i) - \tilde{\Phi}(x_i)|^2}{\sum_{i=1}^{N} |\Phi(x_i)|^2} \right)^{1/2}, \tag{57}$$

where $\tilde{\Phi}(x_i)$ are potentials obtained by the FMM algorithm and $\Phi(x_i)$ are potentials computed by direct calculation in double precision.

Table 1: Timing results for the FMM for 3-digit of accuracy with charges uniformly distributed in a cube. Calculations were performed in single precision.

N	Levels	Boxes	p	S_{exp}	Storage	T_{FMM}	T_{DIR}	Error
20000	4	2267	10	52	1359822	13.3	233	$7.9 \cdot 10^{-4}$
50000	4	4681	10	52	3365896	24.7	1483	$5.2 \cdot 10^{-4}$
200000	5	33749	10	52	24789948	158	24330	$8.4 \cdot 10^{-4}$
500000	5	37449	10	52	28835176	268	138380	$7.0 \cdot 10^{-4}$
1000000	6	48324	10	52	34798506	655	563900	$7.1 \cdot 10^{-4}$

Table 2: Timing results for the FMM for 6-digit of accuracy with charges uniformly distributed in a cube. Calculations were performed in single precision.

N	Levels	Boxes	p	S_{exp}	Storage	T_{FMM}	T_{DIR}	Error
20000	3	585	19	258	1057852	15.9	233	$5.1 \cdot 10^{-7}$
50000	4	2065	19	258	3383488	69	1483	$2.8 \cdot 10^{-7}$
200000	4	4681	19	258	8220716	198	24330	$4.9 \cdot 10^{-7}$
500000	5	36665	19	258	64326704	586	138380	$4.4 \cdot 10^{-7}$
1000000	5	37449	19	258	66414780	1245	563900	$4.4 \cdot 10^{-7}$

The following observations can be made from these tables.

1. The application of the FMM to large scale three dimensional problems is within practical reach.
2. The actual CPU time required by the adaptive FMM algorithm grows approximately linearly with the number of particles N.
3. The algorithm breaks even with the direct calculation at about $N = 750$ for three-digit precision, $N = 1500$ for six-digit precision and $N = 2500$ for nine-digit precision.
4. The performance of the algorithm is quite insensitive to the distribution of charges.

Table 3: Timing results for the FMM for 9-digit of accuracy with charges uniformly distributed in a cube. Calculations were performed in double precision.

N	Levels	Boxes	p	S_{exp}	Storage	T_{FMM}	T_{DIR}	Error
20000	3	585	29	670	2012453	34	296	$2.8 \cdot 10^{-10}$
50000	3	585	29	670	2012453	96	1920	$1.6 \cdot 10^{-10}$
200000	4	4681	29	670	16479203	385	30800	$1.6 \cdot 10^{-10}$
500000	4	4681	29	670	16479203	1219	192600	$1.2 \cdot 10^{-10}$

Table 4: Timing results for the FMM for 3-digit of accuracy with charges distributed on the surface of a sphere. Calculations were performed in single precision.

N	Levels	Boxes	p	S_{exp}	Storage	T_{FMM}	T_{DIR}	Error
20000	7	1746	10	52	891080	8.7	233	$4.2 \cdot 10^{-4}$
50000	9	4757	10	52	2394568	21.6	1483	$3.6 \cdot 10^{-4}$
200000	11	18221	10	52	9126212	97	24330	$8.0 \cdot 10^{-4}$
500000	12	40717	10	52	20413944	224	138380	$6.4 \cdot 10^{-4}$
1000000	13	90139	10	52	45287934	473	563900	$5.5 \cdot 10^{-4}$

Table 5: Timing results for the FMM for 6-digit of accuracy with charges distributed on the surface of a sphere. Calculations were performed in single precision.

N	Levels	Boxes	p	S_{exp}	Storage	T_{FMM}	T_{DIR}	Error
20000	6	624	19	258	1037742	16	233	$2.4 \cdot 10^{-7}$
50000	7	1774	19	258	2774248	40	1483	$2.7 \cdot 10^{-7}$
200000	9	6790	19	258	10365264	183	24330	$2.3 \cdot 10^{-7}$
500000	10	18897	19	258	28580428	529	138380	$4.3 \cdot 10^{-7}$
1000000	11	33289	19	258	50405060	926	563900	$2.9 \cdot 10^{-7}$

Table 6: Timing results for the FMM for 9-digit of accuracy with charges distributed on the surface of a sphere. Calculations were performed in double precision.

N	Levels	Boxes	p	S_{exp}	Storage	T_{FMM}	T_{DIR}	Error
20000	5	429	29	670	1422805	33	296	$3.2 \cdot 10^{-11}$
50000	6	1091	29	670	3616209	98	1920	$8.1 \cdot 10^{-11}$
200000	8	4342	29	670	14394468	409	30800	$7.6 \cdot 10^{-11}$
500000	10	9009	29	670	29828865	1038	192600	$1.2 \cdot 10^{-10}$

Table 7: Timing results for the FMM for 3-digit of accuracy with charges distributed on the surface of a cylinder. Calculations were performed in single precision.

N	Levels	Boxes	p	S_{exp}	Storage	T_{FMM}	T_{DIR}	Error
20000	6	1963	10	52	1013298	8.2	233	$2.7 \cdot 10^{-4}$
50000	7	4084	10	52	2014394	20.8	1483	$4.0 \cdot 10^{-4}$
200000	8	18795	10	52	9056494	93	24330	$5.1 \cdot 10^{-4}$
500000	9	31093	10	52	15409424	194	138380	$5.1 \cdot 10^{-4}$
1000000	9	101374	10	52	49326404	457	563900	$4.9 \cdot 10^{-4}$

Table 8: Timing results for the FMM for 6-digit of accuracy with charges distributed on the surface of a cylinder. Calculations were performed in single precision.

N	Levels	Boxes	p	S_{exp}	Storage	T_{FMM}	T_{DIR}	Error
20000	5	505	19	258	868700	13.8	233	$2.5 \cdot 10^{-7}$
50000	6	2037	19	258	3180832	39	1483	$2.9 \cdot 10^{-7}$
200000	7	7001	19	258	10582852	143	24330	$5.6 \cdot 10^{-7}$
500000	8	19849	19	258	29654956	508	138380	$7.0 \cdot 10^{-7}$
1000000	8	29341	19	258	44253336	921	563900	$6.4 \cdot 10^{-7}$

Table 9: Timing results for the FMM for 9-digit of accuracy with charges distributed on the surface of a cylinder. Calculations were performed in double precision.

N	Levels	Boxes	p	S_{exp}	Storage	T_{FMM}	T_{DIR}	Error
20000	5	505	29	670	1676098	30	296	$2.8 \cdot 10^{-11}$
50000	6	751	29	670	2478241	86	1920	$5.1 \cdot 10^{-11}$
200000	7	2515	29	670	8348058	341	30800	$8.2 \cdot 10^{-11}$
500000	7	7344	29	670	24250893	795	192600	$9.4 \cdot 10^{-11}$

Table 10: Timing results for the FMM for 3-digit of accuracy with charges distributed as in Fig. 7. Calculations were performed in single precision.

N	Levels	Boxes	p	S_{exp}	Storage	T_{FMM}	T_{DIR}	Error
20880	7	1213	10	52	573996	6.7	243	$2.2 \cdot 10^{-4}$
51900	8	4184	10	52	1952046	17	1539	$2.7 \cdot 10^{-4}$
203280	9	15423	10	52	7204398	60	24730	$3.4 \cdot 10^{-4}$
503775	10	45837	10	52	21358082	164	141060	$3.3 \cdot 10^{-4}$
1007655	10	60427	10	52	28513092	282	568090	$2.9 \cdot 10^{-4}$

Table 11: Timing results for the FMM for 6-digit of accuracy with charges distributed as in Fig. 7. Calculations were performed in single precision.

N	Levels	Boxes	p	S_{exp}	Storage	T_{FMM}	T_{DIR}	Error
20880	7	1038	19	258	1601028	17	243	$1.3 \cdot 10^{-7}$
51900	8	1403	19	258	2165338	40	1539	$9.8 \cdot 10^{-8}$
203280	9	4447	19	258	6697050	149	24730	$1.2 \cdot 10^{-7}$
503775	9	15307	19	258	22662792	323	141060	$2.6 \cdot 10^{-7}$
1007655	10	45784	19	258	67176488	714	568090	$2.0 \cdot 10^{-7}$

Table 12: Timing results for the FMM for 9-digit of accuracy with charges distributed as in Fig. 7. Calculations were performed in double precision.

N	Levels	Boxes	p	S_{exp}	Storage	T_{FMM}	T_{DIR}	Error
20880	6	574	29	670	1856177	46	309	$3.6 \cdot 10^{-12}$
51900	7	1191	29	670	3855741	101	2020	$1.1 \cdot 10^{-10}$
203280	8	3883	29	670	12577869	342	32050	$6.5 \cdot 10^{-12}$
503775	9	11499	29	670	37263647	896	193900	$1.0 \cdot 10^{-11}$

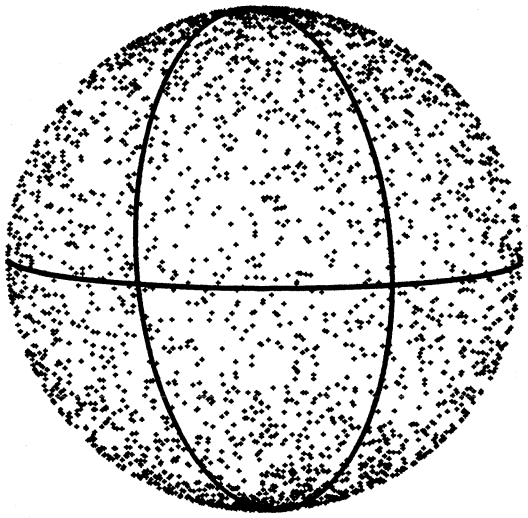

Figure 5: Charges distributed on the surface of a sphere.

Figure 6: Charges distributed on the surface of a cylinder.

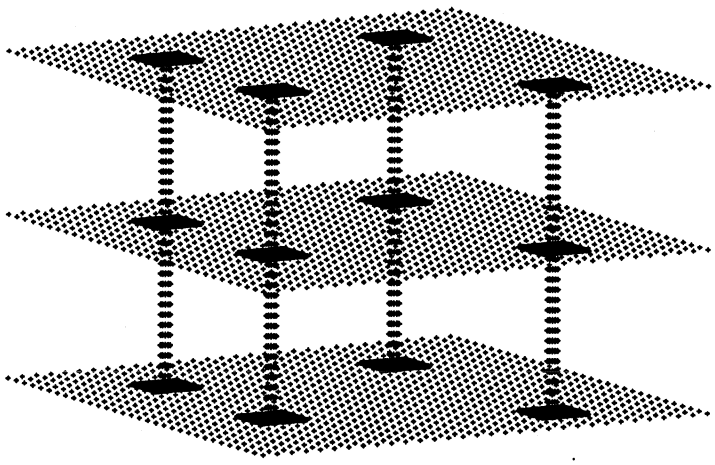

Figure 7: Charges distributed on a complicated object.

7 Generalizations and Conclusions

We have described an adaptive FMM for the Laplace equation based on a new diagonal form for translation operators acting on harmonic functions. It is related to the FMM for the high-frequency Helmholtz equation, in the sense that the latter is based on diagonal forms of translation operators for partial wave expansions [7, 19, 20].

The present scheme admits a number of extensions. The most straightforward ones are to the Helmholtz equation at low frequencies and to the Yukawa equation. The corresponding multipole expansions are well-known, and appropriate plane wave representations have been derived (see, for example, [13]).

From a more abstract perspective, it is worth noting that the main improvement made in this paper and in [12] over earlier FMMs is due to the use of one basis for representing the far field due to a collection of sources (spherical harmonics) and a separate basis for translating information between boxes in the FMM data structure (plane waves). The applicability of this approach is not limited to the Laplace and Helmholtz equations. We are currently in the process of constructing such optimal (or nearly optimal) bases for more general potentials, including those that do not satisfy a partial differential equation, but possess certain less stringent analytical properties. A forthcoming paper [8] describes such an algorithm for the square root of the Laplacian in two dimensions; further generalizations will be reported at a later date.

8 Appendix

The three tables in this Appendix contain the nodes and weights (in columns 2 and 3) needed for discretization of the outer integral in Lemma 2.8. Column 4 contains the number of discretization points needed in the inner integral, which we denote by M_k^d.

Table 13: Nodes, weights and M_k^3 for 3-digit accuracy.

k	*Node*	*Weight*	M_k^3
1	0.10934746769000	0.27107502662774	4
2	0.51769741015341	0.52769158843946	8
3	1.13306591611192	0.69151504413879	16
4	1.88135015110740	0.79834400406452	16
5	2.71785409601205	0.87164160121354	24
6	3.61650274907449	0.92643839116924	24
7	4.56271053303821	0.97294622259483	8
8	5.54900885348528	1.02413865844686	4

Table 14: Nodes, weights and M_k^6 for 6-digit accuracy.

k	$Node$	$Weight$	M_k^6
1	0.05599002531749	0.14239483712194	8
2	0.28485138101968	0.31017671029271	8
3	0.66535367065853	0.44557516683709	16
4	1.16667904805296	0.55303383994159	16
5	1.76443027413431	0.63944903363523	24
6	2.44029832236380	0.70997911214019	32
7	3.18032180991515	0.76828253949732	32
8	3.97371715777193	0.81713201141707	32
9	4.81216799410634	0.85872191623337	48
10	5.68932314511487	0.89480789582390	48
11	6.60040479444377	0.92680189417317	48
12	7.54190497469911	0.95586282708096	48
13	8.51136569298099	0.98299145008230	48
14	9.50723242759128	1.00913395385703	48
15	10.52874809650967	1.03531774600508	48
16	11.57587019602884	1.06318427913963	8
17	12.65078163968520	1.10232109521088	4

Table 15: Nodes, weights and M_k^9 for 9-digit accuracy.

k	$Node$	$Weight$	M_k^9
1	0.03705701953816	0.09473396337900	8
2	0.19219683859955	0.21384206006426	16
3	0.46045971214897	0.32031528543989	16
4	0.82805130101422	0.41254929390710	16
5	1.28121229944787	0.49176691815621	24
6	1.80792019276297	0.55998309037174	32
7	2.39814728074333	0.61909314036708	32
8	3.04359012306582	0.67064351982741	32
9	3.73732742924096	0.71586567032066	48
10	4.47354768940212	0.75576118553096	48
11	5.24735518169467	0.79116885492295	48
12	6.05462948620944	0.82280556212477	64
13	6.89191648795972	0.85129012269433	64
14	7.75633860708838	0.87715909928110	64
15	8.64551915195994	0.90087981520398	64
16	9.55751929613924	0.92286282936149	72
17	10.49078760616705	0.94347471535979	72
18	11.44412262341269	0.96305166489156	80
19	12.41664955395045	0.98191478773737	80
20	13.40781311788324	1.00038891281291	88
21	14.41739038894472	1.01882849188686	88
22	15.44553016867884	1.03765781507554	88
23	16.49282861241170	1.05744113465683	88
24	17.56045648926099	1.07903824697122	72
25	18.65046484106274	1.10434337868208	32
26	19.76847686619416	1.14488166506896	4

References

[1] A. W. APPEL (1985), "An efficient program for many-body simulation", *SIAM J. Sci. Stat. Comput.* **6**, 85–103.

[2] J. BARNES AND P. HUT (1986), "A hierarchical $O(N \log N)$ force-calculation algorithm", *Nature* **324**, 446–449.

[3] L. C. BIEDENHARN AND J. D. LOUCK (1981), *Angular Momentum in Quantum Physics : Theory and Application*, Addison-Wesley, Reading, Mass.

[4] J. CARRIER, L. GREENGARD, AND V. ROKHLIN (1988), "A fast adaptive multipole algorithm for particle simulations", *SIAM J. Sci. Statist. Comput.* **9**, 669–686.

[5] H. CHENG (1995). *Fast, accurate methods for the evaluation of harmonic fields in composite materials*, Ph.D. thesis, New York University.

[6] H. CHENG AND V. ROKHLIN (1998). "Compression of Translation operators in Fast Multipole Algorithms in Three Dimensions", in preparation.

[7] R. COIFMAN, V. ROKHLIN, AND S. WANDZURA (1993), "The fast multipole method for the wave equation: a pedestrian prescription", *IEEE Antennas and Propagation Mag.* **35**, 7.

[8] Z. GIMBUTAS, L. GREENGARD, AND M. MINION (1998), "A fast multipole method for the square root of the Laplacian", in preparation.

[9] L. GREENGARD (1988), *The Rapid Evaluation of Potential Fields in Particle Systems*, MIT Press, Cambridge, Mass.

[10] L. GREENGARD AND V. ROKHLIN (1987), "A fast algorithm for particle simulations", *J. Comput. Phys.* **73**, 325–348.

[11] L. GREENGARD AND V. ROKHLIN (1988a). "Rapid evaluation of potential fields in three dimensions", in *Vortex Methods*, C. Anderson and C. Greengard (eds.), Lecture Notes in Mathematics, vol. 1360, Springer-Verlag, 121.

[12] L. GREENGARD AND V. ROKHLIN (1997). "A new version of the fast multipole method for the Laplace equation in three dimensions", *Acta Numerica* 6, 229-269.

[13] J. HUANG, L. GREENGARD, V. ROKHLIN, AND S. WANDZURA, "Accelerating Fast Multipole Methods for Low Frequency Scattering" CMCL Report 1998-003, New York University (to appear in IEEE Computational Science and Engineering Magazine).

[14] J. D. JACKSON (1975), *Classical Electrodynamics*, Wiley, New York.

[15] O. D. KELLOGG (1953), *Foundations of Potential Theory*, Dover, New York.

[16] MORSE AND FESHBACH (1953), *Methods of Theoretical Physics*, McGraw-Hill, New York.

[17] K. NABORS, F. T. KORSMEYER, F. T. LEIGHTON, AND J. WHITE (1994). "Preconditioned, adaptive, multipole-accelerated iterative methods for three-dimensional first-kind integral equations of potential theory", *SIAM J. Sci. Stat. Comput.* **15**, 714.

[18] V. ROKHLIN (1985), "Rapid solution of integral equations of classical potential theory", *J. Comput. Phys.* **60**, 187–207.

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

[20] V. ROKHLIN (1993), "Diagonal forms of translation operators for the Helmholtz equation in three dimensions", *Appl. and Comput. Harmonic Analysis* **1**, 82–93.

[21] P. R. WALLACE (1984). *Mathematical Analysis of Physical Problems*, Dover, New York.

[22] C.A. WHITE AND M. HEAD-GORDON (1996). "Rotating around the quartic angular momentum barrier in Fast Multipole Method calculation", *J. Chem. Phys.* **105**, 5061.

[23] N. YARVIN AND V. ROKHLIN (1996). "Generalized Gaussian quadratures and singular value decompositions of integral operators", *Department of Computer Science Research Report 1109, Yale University.*