Acta Numerica (1997), *pp.* 229–269

A new version of the Fast Multipole Method for the Laplace equation in three dimensions

Leslie Greengard*
Courant Institute of Mathematical Sciences,
New York University,
New York, NY 10012, USA

Vladimir Rokhlin†
Departments of Mathematics and Computer Science,
Yale University,
New Haven, CT 06520, USA

We introduce a new version of the Fast Multipole Method for the evaluation of potential fields in three dimensions. It is based on a new diagonal form for translation operators and yields high accuracy at a reasonable cost.

CONTENTS

* The work of this author was supported by DARPA/AFOSR under Contract F49620-95-C-0075 and by the US Department of Energy under Contract DEFGO288ER25053.

† The work of this author was supported by DARPA/AFOSR under Contract F49620-95-C-0075 and by ONR under Grant N00014-96-1-0188.

1. Introduction

In this paper, we introduce a new version of the Fast Multipole Method (FMM) for the evaluation of potential fields in three dimensions. The scheme evaluates all pairwise interactions in large ensembles of particles, *i.e.* expressions of the form

$$\Phi(x_j) = \sum_{\substack{i=1 \\ i \neq j}}^{n} \frac{q_i}{\|x_j - x_i\|} \tag{1.1}$$

for the gravitational or electrostatic potential, and

$$E(x_j) = \sum_{\substack{i=1 \\ i \neq j}}^{n} q_i \frac{x_j - x_i}{\|x_j - x_i\|^3} \tag{1.2}$$

for the field, where $x_1, x_2, \ldots, x_n$ are points in $\mathbb{R}^3$, and $q_1, q_2, \ldots, q_n$ are a set of (real) coefficients. Here $\|\cdot\|$ denotes the Euclidean norm.

The evaluation of expressions of the form (1.1) is closely related to a number of important problems in applied mathematics, physics, chemistry and biology. Molecular dynamics and Hartree–Fock calculations in chemistry, the evolution of large-scale gravitational systems in astrophysics, capacitance extraction in electrical engineering, and vortex methods in fluid dynamics are all examples of areas where simulations require rapid and accurate evaluation of sums of the form (1.1) and (1.2). When certain closely related interactions are considered as well, involving expressions of the form

$$\Phi(x_j) = \sum_{\substack{i=1 \\ i \neq j}}^{n} q_i \frac{e^{ik\|x_j - x_i\|}}{\|x_j - x_i\|}, \tag{1.3}$$

the list of applications becomes even more extensive.

This paper is a continuation (after an interval of several years) of a sequence of joint papers by the authors, starting with Greengard and Rokhlin (1987) and Carrier, Greengard and Rokhlin (1988) which introduced the Fast Multipole Method in two dimensions. Subsequent work extended the method to three dimensions (Greengard 1988, Greengard and Rokhlin 1988*a*, 1988*b*), and there followed a number of versions of the scheme, both by the present authors and by other researchers; see, for example, Anderson (1992), Nabors, Korsmeyer, Leighton and White (1994), Berman (1995), Epton and Dembart (1995), Elliott and Board (1996). After about ten years of research, however, a somewhat unsatisfactory picture has emerged. In short, there now exist extremely efficient algorithms for the evaluation of the two-dimensional analogues of (1.1), (1.2) with (practically) arbitrarily high precision, as well as very efficient and accurate algorithms for a host of related problems (Rokhlin 1988, Alpert and Rokhlin 1991, Beylkin, Coifman and Rokhlin 1991, Coifman and Meyer 1991, Greengard and Strain 1991, Strain 1991, Alpert, Beylkin, Coifman and Rokhlin 1993). However, for the sums (1.1) and (1.2), there are few practical schemes, and these provide only limited accuracy. Since most real-world problems are three-dimensional, it can be said that analysis-based 'fast' methods are a promising group of techniques, but that they have not yet lived up to all their expectations.

In the present paper, we try to remedy this situation. We describe a version of the Fast Multipole Method in three dimensions that produces high accuracy at an acceptable computational cost. As will be seen from the numerical examples in Section 9, the new scheme has a break-even point of $n \sim 2000$ when compared with direct calculation in single precision; with 10-digit accuracy, the break-even point is $n \sim 5000$; with 3-digit accuracy, it is $n \sim 500$. The approach uses a considerably more involved mathematical (and numerical) apparatus than is customary in the design of fast multipole-type algorithms. This apparatus is based on a new diagonal form for translation operators acting on harmonic functions, extending the two-dimensional version introduced by Hrycak and Rokhlin (1995). The overall approach bears some resemblance to that used in Fast Multipole Methods for high-frequency scattering problems, which are based on diagonal forms of translation operators for the Helmholtz equation (Rokhlin 1990*b*, 1995, Epton and Dembart 1995).

2. Philosophical preliminaries

We begin with an overview of analysis-based 'fast' numerical algorithms, concentrating on the evaluation of expressions of the form (1.1). Where possible, we summarize the current 'state of the art' in the field.

If we define the $n \times n$ matrix A by the formula

$$A_{ij} = \frac{1}{\|x_j - x_i\|}, \tag{2.1}$$

we can rewrite (1.1) in the form

$$\Phi = Aq, \tag{2.2}$$

with $\Phi, q \in \mathbb{R}^n$ (the expression (1.2) can be rewritten in a similar fashion). Obviously, straightforward evaluation of either of the expressions (1.1), (1.2) requires $O(n^2)$ operations (evaluating n potentials at n points), and for large-scale problems this estimate is prohibitively large. On the other hand, the evaluation of expressions of the forms (1.1), (1.2) is an integral part of the numerical solution of many important problems in applied mathematics, and during the last decade, several 'fast' schemes have been proposed for this purpose, that is, schemes whose computational cost is less than $O(n^2)$. Typically, such methods require $O(n)$ or $O(n \log n)$ operations (Rokhlin 1985, Anderson 1986, Greengard and Rokhlin 1987, Carrier et al. 1988, Rokhlin 1988, 1990*b*, Brandt and Lubrecht 1990, Brandt 1991, Alpert and Rokhlin

1991, Beylkin et al. 1991, Coifman and Meyer 1991, Greengard and Strain 1991, Strain 1991, 1992, Epton and Dembart 1995). All of them are based on the straightforward observation that the potentials are smooth functions in $\mathbb{R}^3$, except when x_i is near x_j, and as a result, large submatrices of A are well approximated by low-rank matrices. Clearly, applying a matrix of dimension $n \times n$ but rank J to an arbitrary vector requires only nJ operations (as opposed to n^2); this simple observation leads directly to a variety of asymptotically 'fast' schemes for the evaluation of (1.1); below, we illustrate the construction of such schemes with a simple example.

Suppose that, in the expression (1.1), the points $x_1, x_2, \ldots, x_n$ are equispaced and lie on the interval $[-1, 1]$, so that

$$x_1 = -1, \quad x_2 = -1 + h, \quad \ldots, \quad x_{n-1} = 1 - h, \quad x_n = 1, \tag{2.3}$$

where $h = 2/(n-1)$. Given three integers l, m, k such that

$$\begin{aligned} 1 &\leq l \leq n, \\ 1 &\leq m \leq n, \\ 1 &\leq k \leq n - l, \\ 1 &\leq k \leq n - m, \end{aligned} \tag{2.4}$$

we will denote by $A_{l,m,k}$ the submatrix of A consisting of such elements A_{ij} that

$$\begin{aligned} l &\leq i \leq l + k - 1, \\ m &\leq j \leq m + k - 1, \end{aligned} \tag{2.5}$$

and say that $A_{l,m,k}$ is separated from the diagonal if

$$\mid l - (m + k - 1) \mid > k, \tag{2.6}$$

and

$$\mid m - (l + k - 1) \mid > k. \tag{2.7}$$

In other words, we will say that the submatrix $A_{l,m,k}$ of the matrix A is separated from the diagonal if its distance from the diagonal of A is greater than or equal to its own size (Figure 1). We will construct a rudimentary 'fast' algorithm for the application of the matrix A to an arbitrary vector by means of the following lemma; its proof is based on several well-known facts, all of which can be found in Dahlquist and Bjork (1974).

Lemma 2.1 For any integer $p \leq 1$, and any l, m, k satisfying the conditions (2.4), there exists a matrix $B_{l,m,k}$ of dimension $k \times k$ and rank J, such that

$$||A_{l,m,k} - B_{l,m,k}|| \leq \frac{1}{4^J}. \tag{2.8}$$

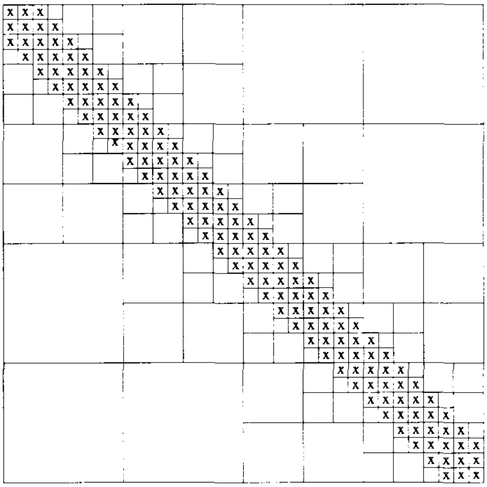

Fig. 1. Subdivision of matrix into well separated blocks. The submatrices marked by an X are not well separated from the diagonal

In other words, any submatrix of A separated from the diagonal is of rank J, to the precision $1/4^J$.

Outline of proof. We start by defining the function $f : \mathbb{R}^2 \to \mathbb{R}^1$ by the formula

$$f(x,y) = \frac{1}{\|x-y\|}, \tag{2.9}$$

and observing that f is smooth everywhere in $\mathbb{R}^2$, except when $x = y$. We will say that the square $[a, a+c] \times [b, b+c] \subset \mathbb{R}^2$ is separated from the diagonal if

$$\mid a+c-b \mid > c, \tag{2.10}$$

and

$$\mid b+c-a \mid > c, \tag{2.11}$$

and observe that on any such square, the function f can be expanded in a two-dimensional Chebychev series, that is, represented in the form

$$f(x,y) = \sum_{p,q=0}^{\infty} \alpha_{pq} T_p\left(\frac{2x}{c} - \frac{c+2a}{c}\right) T_q\left(\frac{2y}{c} - \frac{c+2b}{c}\right), \tag{2.12}$$

with T_j denoting the jth Chebychev polynomial. Finally, we observe that for any a, b, c satisfying the conditions (2.10), (2.11), the convergence of the expansion (2.12) is given by the formula

$$\left| f(x,y) - \sum_{p,q=0}^{J} \alpha_{pq} T_p\left(\frac{2x}{c} - \frac{c+2a}{c}\right) T_q\left(\frac{2y}{c} - \frac{c+2b}{c}\right) \right| < \frac{1}{4^J}. \tag{2.13}$$

In other words, for any square separated from the diagonal, the expansion (2.12) converges to accuracy ε after no more than $\log_4(\varepsilon)$ terms. Combining (2.6), (2.7) and (1.1) with (2.12) and (2.13), we observe that, for any i, j satisfying the inequalities (2.5),

$$\left| A_{ij} - \sum_{p,q=0}^{J} \alpha_{pq} T_p\left(\frac{2x_i}{c} - \frac{c+2a}{c}\right) T_q\left(\frac{2y_j}{c} - \frac{c+2b}{c}\right) \right| < \frac{1}{4^J}, \tag{2.14}$$

with $a = (2l)/n - 1$, $b = (2m)/n - 1$, $c = (2k)/n$. The matrix $B_{l,m,k}$ defined by

$$(B_{l,m,k})_{ij} = \sum_{p,q=0}^{J} \alpha_{pq} T_p\left(\frac{2x_i}{c} - \frac{c+2a}{c}\right) T_q\left(\frac{2y_j}{c} - \frac{c+2b}{c}\right) \tag{2.15}$$

clearly satisfies the desired condition (2.8). □

In order to develop a fast algorithm, we first subdivide the matrix A into a collection of submatrices, as depicted in Figure 1. Each of the submatrices in this structure is separated from the diagonal, except the submatrices near the diagonal whose ranks are small simply because their dimensionality is small. By virtue of Lemma 2.1, each of the separated submatrices is of rank J, to the accuracy 4^{-J}. In order to apply A to an arbitrary vector with fixed but finite accuracy (which is always the case in numerical computations), we can apply each of the submatrices to the appropriate part of the vector for a cost proportional to kJ, where k is the size of the submatrix. Adding up the costs for all such submatrices, we obtain the operation count of

$$Jn \log n \sim \log\left(\frac{1}{\varepsilon}\right) n \log n, \tag{2.16}$$

instead of n^2.

The scheme outlined above is extremely simple, but representative of the current approach to the design of 'fast' summation algorithms. Several comments are in order.

1. It is easy to see that the matrix A defined in (2.1) with the spacing defined by (2.3) is in fact a Toeplitz matrix that can be applied to an arbitrary vector for a cost proportional to $n \log n$ via the Fast Fourier Transform. This situation occurs sometimes, both in one and higher dimensions. However, the Toeplitz nature of the matrix A is lost when the points are not distributed on a uniform grid, and direct application

of the FFT becomes impossible. For 'somewhat uniformly' distributed points x_i, various types of local corrections have been successfully utilized. When the points are not distributed uniformly (for example, on a curve or surface), FFT-based methods become ineffective.

2. As described, the scheme is only applicable to one-dimensional problems, and under very limited conditions. In most situations, the subdivision of the matrix has to be modified, taking into account the geometric distribution of points in order to locate submatrices whose 'numerical rank' is low. Examples of such subdivisions can be found in Carrier et al. (1988), Van Dommelen and Rundensteiner (1989), Beylkin et al. (1991) and Nabors et al. (1994).

3. The scheme is extremely simple and general. It is entirely unrelated to the detailed nature of the matrix A, needing only some inequality like (2.13). In other words, so long as the entries of the matrix A are smooth functions of their indices away from the diagonal, a scheme of the type outlined above will work. In fact, even that is not necessary; the elements of the matrix have only to be *sufficiently smooth functions of their indices on a sufficiently large part of the matrix.*

4. The scheme admits a large number of modifications; the most obvious ones replace the Chebychev expansion in (2.12) with other approximations; one should be careful in doing so, since under many conditions the Chebychev approximation is optimal (among polynomial approximations), or nearly so. Some of the special-purpose approximation schemes that have been used successfully employ wavelets and related bases (Beylkin et al. 1991, Alpert et al. 1993).

   Another obvious modification is a change in the choice of submatrices of low rank; the use of rectangular submatrices (as opposed to the square ones in Figure 1) permits coarser subdivisions and tends to result in more efficient algorithms.

5. Algorithms of the type described above usually do not work for problems where the matrix A is a discretization of an integral operator with an oscillatory kernel, since such discretizations (normally) have a more or less constant number of nodes per wavelength of the dominant oscillation. As a result, the rank of each submatrix is proportional to its size, and the resulting algorithms have CPU time estimates of the order $O(n^2)$. Sometimes, the calculation can be accelerated by reducing the size of the constant (Wagner and Chew 1994), but the asymptotic complexity in such cases is the same as for the direct approach. For certain classes of oscillatory problems (such as Helmholtz and Schrödinger equations at high frequency), there exist asymptotically 'fast' schemes

based on a different (and considerably more involved) analytical apparatus; see, for example, Rokhlin (1988, 1990*b*, 1993), Canning (1989, 1992, 1993), Coifman and Meyer (1991), Bradie, Coifman and Grossmann (1993), Coifman, Rokhlin and Wandzura (1993, 1994), Wagner and Chew (1994), Epton and Dembart (1995). As noted in the introduction, these schemes are related to the scheme we will present below. They are, however, outside the scope of this paper.

3. Mathematical preliminaries I

In this section, we briefly derive the multipole expansion of a charge distribution and refer the reader to Kellogg (1953), Jackson (1975), Wallace (1984), and Greengard (1988) for more detailed discussions.

If a point charge of strength q is located at $P_0 = (x_0, y_0, z_0)$, then the potential and electrostatic field due to this charge at a distinct point $P = (x, y, z)$ are given by

$$\Phi = \frac{1}{R} \tag{3.1}$$

and

$$\mathbf{E} = -\nabla\Phi = \left(\frac{x - x_0}{R^3}, \frac{y - y_0}{R^3}, \frac{z - z_0}{R^3}\right), \tag{3.2}$$

respectively, where R denotes the distance between points P_0 and P.

We would like to derive a series expansion for the potential at P in terms of its distance from the origin r. For this, let the spherical coordinates of P be (r, θ, ϕ) and of P_0 be (ρ, α, β). Letting γ be the angle between the vectors P and P_0, we have from the cosine rule

$$R^2 = r^2 + \rho^2 - 2r\rho\cos\gamma, \tag{3.3}$$

with

$$\cos\gamma = \cos\theta\cos\alpha + \sin\theta\sin\alpha\cos(\phi - \beta). \tag{3.4}$$

Thus,

$$\frac{1}{R} = \frac{1}{r\sqrt{1 - 2\frac{\rho}{r}\cos\gamma + \frac{\rho^2}{r^2}}} = \frac{1}{r\sqrt{1 - 2u\mu + \mu^2}}, \tag{3.5}$$

having set

$$\mu = \frac{\rho}{r} \quad \text{and} \quad u = \cos\gamma. \tag{3.6}$$

For $\mu < 1$, we may expand the inverse square root in powers of μ, resulting in a series of the form

$$\frac{1}{\sqrt{1 - 2u\mu + \mu^2}} = \sum_{n=0}^{\infty} P_n(u)\mu^n \tag{3.7}$$

where

$$P_0(u) = 1, \quad P_1(u) = u, \quad P_2(u) = \frac{3}{2}\left(u^2 - \frac{1}{3}\right), \quad \ldots \tag{3.8}$$

and, in general, $P_n(u)$ is the Legendre polynomial of degree n. Our expression for the field now takes the form

$$\frac{1}{R} = \sum_{n=0}^{\infty} \frac{\rho^n}{r^{n+1}} P_n(u). \tag{3.9}$$

The angular parameter u, however, depends on both the source and the target locations. A more general representation will require the introduction of spherical harmonics, which are solutions of the Laplace equation obtained by separation of variables in spherical coordinates. Any harmonic function Φ can be expanded in the form

$$\Phi = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} \left(L_n^m r^n + \frac{M_n^m}{r^{n+1}} \right) Y_n^m(\theta, \phi). \tag{3.10}$$

The terms $Y_n^m(\theta, \phi) r^n$ are referred to as spherical harmonics of degree n or *solid harmonics*, the terms $Y_n^m(\theta, \phi)/r^{n+1}$ are called spherical harmonics of degree $-n-1$ or *multipoles*, and the coefficients L_n^m and M_n^m are known as the moments of the expansion.

The spherical harmonics can be expressed in terms of partial derivatives of $1/r$ (Wallace 1984) as

$$\frac{Y_n^0(\theta, \phi)}{r^{n+1}} = A_n^0 \frac{\partial^n}{\partial z^n} \left(\frac{1}{r} \right). \tag{3.11}$$

For $m > 0$, we have

$$\frac{Y_n^m(\theta, \phi)}{r^{n+1}} = A_n^m \left(\frac{\partial}{\partial x} + i \frac{\partial}{\partial y} \right)^m \left(\frac{\partial}{\partial z} \right)^{n-m} \left(\frac{1}{r} \right), \tag{3.12}$$

and

$$\frac{Y_n^{-m}(\theta, \phi)}{r^{n+1}} = A_n^m \left(\frac{\partial}{\partial x} - i \frac{\partial}{\partial y} \right)^m \left(\frac{\partial}{\partial z} \right)^{n-m} \left(\frac{1}{r} \right), \tag{3.13}$$

where

$$A_n^m = \frac{(-1)^n}{\sqrt{(n-m)!(n+m)!}}. \tag{3.14}$$

They also satisfy the relation

$$Y_n^m(\theta, \phi) \equiv \sqrt{\frac{(n-|m|)!}{(n+|m|)!}} P_n^{|m|}(\cos\theta) e^{im\phi}, \tag{3.15}$$

where we have omitted the normalization factor of $\sqrt{(2n+1)/4\pi}$, to match the definitions (3.11)–(3.13) given above. The special functions P_n^m are called associated Legendre functions and can be defined by Rodrigues' formula

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

Theorem 3.1 (Addition theorem for Legendre polynomials) Let P and Q be points with spherical coordinates (r,θ,ϕ) and (ρ,α,β), respectively, and let γ be the angle subtended between them. Then

$$P_n(\cos\gamma) = \sum_{m=-n}^{n} Y_n^{-m}(\alpha,\beta) Y_n^m(\theta,\phi). \tag{3.16}$$

Combining Theorem 3.1 and equation (3.9), we have

$$\frac{1}{R} = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} \rho^n Y_n^{-m}(\alpha,\beta) \frac{Y_n^m(\theta,\phi)}{r^{n+1}}. \tag{3.17}$$

It is now straightforward to expand the field due to a collection of sources in terms of multipoles.

Theorem 3.2 (Multipole expansion) Suppose that k charges of strengths $\{q_i,\ i=1,\ldots,k\}$ are located at the points $\{Q_i = (\rho_i,\alpha_i,\beta_i),\ i = 1,\ldots,k\}$, with $|\rho_i| < a$. Then for any $P = (r,\theta,\phi) \in \mathbb{R}^3$ with $r > a$, the potential $\Phi(P)$ is given by

$$\Phi(P) = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} \frac{M_n^m}{r^{n+1}} Y_n^m(\theta,\phi), \tag{3.18}$$

where

$$M_n^m = \sum_{i=1}^{k} q_i \rho_i^n Y_n^{-m}(\alpha_i,\beta_i). \tag{3.19}$$

Furthermore, for any $p \geq 1$,

$$\left| \Phi(P) - \sum_{n=0}^{p} \sum_{m=-n}^{n} \frac{M_n^m}{r^{n+1}} Y_n^m(\theta,\phi) \right| \leq \frac{A}{r-a} \left(\frac{a}{r}\right)^{p+1}, \tag{3.20}$$

where

$$A = \sum_{i=1}^{k} |q_i|. \tag{3.21}$$

Proof. The formula (3.19) follows from equation (3.17) and superposition. The error bound is obtained from the triangle inequality and the fact that the ratios ρ_i/r are bounded from above by a/r. □

Suppose now that $r = 2a$ in the context of the preceding theorem. Then the error bound (3.20) becomes

$$\left|\Phi(P) - \sum_{n=0}^{p} \sum_{m=-n}^{n} \frac{M_n^m}{r^{n+1}} Y_n^m(\theta, \phi)\right| \leq \frac{A}{a} \left(\frac{1}{2}\right)^{p+1}, \tag{3.22}$$

and setting $p = \log_2(1/\varepsilon)$ yields a precision ε relative to the ratio A/a.

4. An $N \log N$ algorithm

Theorem 3.2 is all that is required to construct a simple fast algorithm of arbitrary precision. To reduce the number of issues addressed, we assume that the particles are fairly homogeneously distributed in a square so that adaptive refinement is not required.

In order to make systematic use of multipole expansions, we introduce a hierarchy of boxes which refine the computational domain into smaller and smaller regions. At refinement level 0, we have the entire computational domain. Refinement level $l+1$ is obtained recursively from level l by subdivision of each box into eight equal parts. This yields a natural tree structure, where the eight boxes at level $l+1$ obtained by subdivision of a box at level l are considered its children.

Definition 4.1 Two boxes are said to be *near neighbours* if they are at the same refinement level and share a boundary point (a box is a near neighbour of itself).

Definition 4.2 Two boxes are said to be *well separated* if they are at the same refinement level and are not near neighbours.

Definition 4.3 With each box i we associate an *interaction list*, consisting of the children of the near neighbours of i's parent which are well separated from box i (Figure 4).

Definition 4.4 With each box i at level l we associate a multipole expansion $\Phi_{l,i}$ about the box centre, which describes the far field induced by the particles contained inside the box.

The basic idea is to consider clusters of particles at successive levels of spatial refinement, and to compute interactions between distant clusters by means of multipole expansions when possible. It is clear that at levels 0 and 1, there are no pairs of boxes that are well separated. At level 2, on the other hand, sixty-four boxes have been created and there is a number of well separated pairs. Multipole expansions can then be used to compute

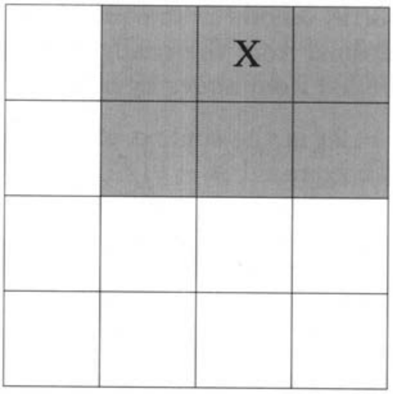

Fig. 2. The first step of the algorithm, depicted in two space dimensions for clarity. Interactions between particles in box X and its near neighbours (grey) are not computed. Interactions between well separated boxes are computed via multipole expansions

interactions between these well separated pairs (Figure 2) with rigorous bounds on the error. In fact, it is easy to see that the bound (3.20) applies with the ratio $a/r < 1/\sqrt{3}$. Thus, to achieve a given precision ε, we need to use $p = \log_{\sqrt{3}}(1/\varepsilon)$ terms.

It remains to compute the interactions between particles contained in each box with those contained in the box's near neighbours, and this is done recursively. We first refine each level 2 box to create level 3. For a given level 3 box, we then seek to determine which other level 3 boxes can be interacted with by means of multipole expansions. Since those boxes outside the region of the *parent*'s nearest neighbours are already accounted for (at level 2), they can be ignored. Since interactions with near neighbours cannot be accounted for accurately by means of an expansion, they can also be ignored for the moment. The remaining boxes correspond exactly to the interaction list defined above (Figure 3).

The nature of the recursion is now clear. At every level, the multipole expansion is formed for each box due to the particles it contains. The resulting expansion is then evaluated for each particle in the region covered by its interaction list (Figure 4).

We halt the recursive process after roughly $\log_8 N$ levels of refinement. The amount of work done at each level is of the order $O(N)$. To see this, note first that approximately $N\,p^2$ operations are needed to create all expansions, since each particle contributes to p^2 expansion coefficients. Secondly, from the point of view of a single particle, there are at most 189 boxes (the maximum size of the interaction list) whose expansions are computed, so that $189\,N\,p^2$ operations are needed for all evaluations.

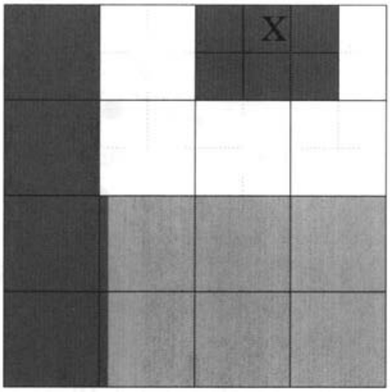

Fig. 3. The second step of the algorithm, depicted in two space dimensions. After refinement, note that the particles in the box marked X have already interacted with the most distant particles (light grey). They are now well separated from the particles in the white boxes, so that these interactions can be computed via multipole expansions. The near neighbour interactions (dark grey) are not computed

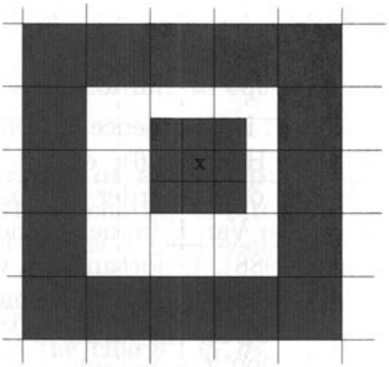

Fig. 4. Subsequent steps of the algorithm. The interaction list for box X is indicated in white. In three dimensions, it contains up to 189 boxes

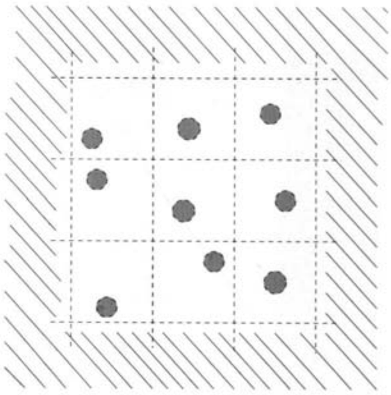

Fig. 5. At the finest level, interactions with near neighbours are computed directly. In three dimensions, there are up to 27 near neighbours

At the finest level, we have created roughly $8^{\log_8 N} = N$ boxes and it remains only to compute interactions between nearest neighbours. By the assumption of homogeneity, there are $O(1)$ particles per box, so that this last step requires about $27N$ operations (Figure 5). The total cost is approximately

$$189Np^2 \log_8 N + 27N. \tag{4.1}$$

The algorithm just described is, in essence, a nonadaptive version of the one proposed by Barnes and Hut (1986), except that it achieves arbitrary precision through the use of high order expansions. Two-dimensional schemes of this type are due to Van Dommelen and Rundensteiner (1989) and Odlyzko and Schönhage (1988). Unfortunately, while such schemes have good asymptotic work estimates, the three-dimensional versions provide only modest speedups at high precision for the values of N encountered in present day applications. At $N = 100,000$, for example, seven digits of accuracy require $p \approx 20$, and the $N \log N$ scheme is only two to three times faster than the direct $O(N^2)$ method. In order to accelerate the calculation significantly, we need some further analytic machinery.

5. Mathematical preliminaries II

The FMM relies on three translation operators, acting on either multipole (far field) or solid harmonic (local) expansions. They are described in the next three theorems (Greengard and Rokhlin 1988*a*, Greengard 1988).

Theorem 5.1 (Translation of a multipole expansion) Suppose that l charges of strengths $q_1, q_2, \ldots, q_l$ are located inside the sphere D of radius a with centre at $Q = (\rho, \alpha, \beta)$, and that for points $P = (r, \theta, \phi)$ outside D, the potential due to these charges is given by the multipole expansion

$$\Phi(P) = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} \frac{O_n^m}{r'^{\,n+1}} Y_n^m(\theta', \phi'), \tag{5.1}$$

where $P - Q = (r', \theta', \phi')$. Then for any point $P = (r, \theta, \phi)$ outside the sphere D_1 of radius $(a + \rho)$,

$$\Phi(P) = \sum_{j=0}^{\infty} \sum_{k=-j}^{j} \frac{M_j^k}{r^{j+1}} Y_j^k(\theta, \phi), \tag{5.2}$$

where

$$M_j^k = \sum_{n=0}^{j} \sum_{m=-n}^{n} \frac{O_{j-n}^{k-m} i^{|k|-|m|-|k-m|} A_n^m A_{j-n}^{k-m} \rho^n Y_n^{-m}(\alpha, \beta)}{A_j^k}, \tag{5.3}$$

with A_n^m defined by equation (3.14). Furthermore, for any $p \geq 1$,

$$\left| \Phi(P) - \sum_{j=0}^{p} \sum_{k=-j}^{j} \frac{M_j^k}{r^{j+1}} Y_j^k(\theta, \phi) \right| \leq \left(\frac{\sum_{i=1}^{l} |q_i|}{r - (a + \rho)} \right) \left(\frac{a + \rho}{r} \right)^{p+1}. \tag{5.4}$$

Definition 5.1 The linear operator mapping old multipole coefficients $\{O_j^k : 0 \leq n \leq p,\ -n \leq m \leq n\}$, to new multipole coefficients $\{M_j^k : 0 \leq n \leq p,\ -n \leq m \leq n\}$ according to equation (5.3) will be denoted by $\mathcal{T}_{MM}$.

Theorem 5.2 (Conversion of a multipole expansion into a local expansion) Suppose that l charges of strengths $q_1, q_2, \ldots, q_l$ are located inside the sphere D_Q of radius a with centre at $Q = (\rho, \alpha, \beta)$, and that $\rho > (c + 1)a$ with $c > 1$. Then the corresponding multipole expansion (5.1) converges inside the sphere D_0 of radius a centred at the origin. Inside D_0, the potential due to the charges $q_1, q_2, \ldots, q_l$ is described by a local expansion:

$$\Phi(P) = \sum_{j=0}^{\infty} \sum_{k=-j}^{j} L_j^k Y_j^k(\theta, \phi) r^j, \tag{5.5}$$

where

$$L_j^k = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} \frac{O_n^m i^{|k-m|-|k|-|m|} A_n^m A_j^k Y_{j+n}^{m-k}(\alpha, \beta)}{(-1)^n A_{j+n}^{m-k} \rho^{j+n+1}}, \tag{5.6}$$

with A_r^s defined by equation (3.14). Furthermore, for any $p \geq 1$,

$$\left| \Phi(P) - \sum_{j=0}^{p} \sum_{k=-j}^{j} L_j^k Y_j^k(\theta, \phi) r^{j+1} \right| \leq \left(\frac{\sum_{i=1}^{l} |q_i|}{ca - a} \right) \left(\frac{1}{c} \right)^{p+1}. \tag{5.7}$$

Definition 5.2 The linear operator mapping truncated multipole expansion coefficients $\{O_j^k : 0 \leq j \leq p, \ -j \leq k \leq j\}$ to local coefficients $\{L_j^k : 0 \leq j \leq p, \ -j \leq k \leq j\}$ according to equation (5.6) will be denoted by $\mathcal{T}_{ML}$.

Theorem 5.3 (Translation of a local expansion) Let $Q = (\rho, \alpha, \beta)$ be the origin of a local expansion

$$\Phi(P) = \sum_{n=0}^{p} \sum_{m=-n}^{n} O_n^m Y_n^m(\theta', \phi') r'^n, \tag{5.8}$$

where $P = (r, \theta, \phi)$ and $P - Q = (r', \theta', \phi')$. Then

$$\Phi(P) = \sum_{j=0}^{p} \sum_{k=-j}^{j} L_j^k Y_j^k(\theta, \phi) r^j, \tag{5.9}$$

where

$$L_j^k = \sum_{n=j}^{p} \sum_{m=-n}^{n} \frac{O_n^m i^{|m|-|m-k|-|k|} A_{n-j}^{m-k} A_j^k Y_{n-j}^{m-k}(\alpha, \beta) \rho^{n-j}}{(-1)^{n+j} A_n^m}, \tag{5.10}$$

with A_r^s defined by equation (3.14).

Definition 5.3 The linear operator mapping old local expansion coefficients $\{O_n^m : 0 \leq n \leq p, \ -n \leq m \leq n\}$ to new local expansion coefficients $\{L_n^m : 0 \leq n \leq p, \ -n \leq m \leq n\}$ according to equation (5.10) will be denoted by $\mathcal{T}_{LL}$.

6. The original FMM

We can now construct a scheme with cost proportional to N, by using Theorem 5.2 to convert the far field expansion of a source box into a local expansion inside a target box, rather than by direct evaluation of the far field expansion at individual target positions.

Definition 6.1 With each box i at level l we associate a local expansion $\Psi_{l,i}$ about the box centre, which describes the potential field induced by all particles outside box i's near neighbours.

Definition 6.2 With each box i at level l we associate a local expansion $\tilde{\Psi}_{l,i}$ about the box centre, which describes the potential field induced by all particles outside the near neighbours of i's *parent*.

ALGORITHM 1

The parent of a box j will be denoted by $p(j)$. The list of children of a box j will be denoted by $c(j)$. The interaction list of a box j will be denoted by *ilist*(j).

```
Upward pass
Initialization
Choose the number of refinement levels $n \approx \log_8 N$, and the order of the multipole expansion desired $p$. The number of boxes at the finest level is then $8^n$, and the average number of particles per box is $s = N/(8^n)$.

Step 1
Form multipole expansions $\Phi_{n,i}$ of potential field due to particles in each box about the box centre at the finest mesh level, via Theorem 3.2.

Step 2
For levels $l = n-1, \ldots, 2$,
    Form multipole expansion $\Phi_{l,j}$ about the centre of each box at level $l$ by merging expansions from its eight children via Theorem 5.1.
        $\Phi_{l,j} = \sum_{k \in c(j)} \mathcal{T}_{MM} \Phi_{l+1,k}$.

Downward pass
Initialization
Set $\Psi_{1,1} = \Psi_{1,2} = \cdots = \Psi_{1,8} = (0, 0, \ldots, 0)$.

Step 3
For levels $l = 2, \ldots, n$,
    Form the expansion $\tilde{\Psi}_{l,j}$ for each box $j$ at level $l$, by using Theorem 5.3 to shift the local $\Psi$ expansion of $j$'s parent to $j$ itself.
        $\tilde{\Psi}_{l,j} = \mathcal{T}_{LL} \Psi_{l-1,p(j)}$.

    Form $\Psi_{l,j}$ by using Theorem 5.2 to convert the multipole expansion $\Phi_{l,k}$ of each box $k$ in the interaction list of box $j$ to a local expansion about the centre of box $j$, adding these local expansions together, and adding the result to $\tilde{\Psi}_{l,j}$.
        $\Psi_{l,j} = \tilde{\Psi}_{l,j} + \sum_{k \in \text{ilist}(j)} \mathcal{T}_{ML} \Phi_{l,k}$.

Step 4
For each particle in each box $j$ at the finest level $n$,
    evaluate $\Psi_{n,j}$ at the particle position.

Step 5
For each particle in each box $j$ at the finest level $n$,
    compute interactions with particles in near neighbour boxes directly.
```

Since s is the average number of particles per box at the finest level, there are approximately N/s boxes in the tree hierarchy. Therefore, Step 1 requires approximately Np^2 work, Step 2 requires $(N/s)\,p^4$ work, Step 3 requires $189(N/s)\,p^4$ work, Step 4 requires $N\,p^2$ work, and Step 5 requires $27N\,s$ work. Thus, a reasonable estimate for the total operation count is

$$191\left(\frac{N}{s}\right)p^4 + 2Np^2 + 27Ns. \tag{6.1}$$

With $s = 2p^2$, the operation count becomes approximately

$$150N\,p^2. \tag{6.2}$$

This would appear to beat the estimate (4.1) for any N, but there is a subtle catch. The number of terms p needed for a fixed precision in the $N \log N$ scheme is smaller than the number of terms needed in the FMM described above. To see why, consider two interacting cubes A and B of unit volume, with sources in A and targets in B. The worst-case multipole error decays like $(\sqrt{3}/3)^p$, since $\sqrt{3}/2$ is the radius of the smallest sphere enclosing cube A and $3/2$ is the shortest distance to a target in B. The conversion of a multipole expansion in A to a local expansion in B, however, satisfies an error bound which depends on the smallest sphere enclosing B as well as the smallest sphere enclosing A. From equation (5.7), the worst case error is less than $(0.76)^p$, although with more detailed analysis, one can show that the error is bounded by $(0.75)^p$ (Petersen, Smith and Soelvason 1995).

In the original FMM (Greengard and Rokhlin 1988*a*, Greengard 1988), it was suggested that one redefine the nearest neighbour list to include 'second nearest neighbours,' so that boxes which interact via multipole expansions are separated by at least two intervening boxes of the same size. The error can then be shown to decay approximately like $(0.4)^p$. However, the number of near neighbours increases from 27 to 125 and the size of the interaction list increases from 189 to 875.

It is clear that the major obstacle to achieving reasonable efficiency at high precision is the cost of the multipole to local translations ($189p^4$ operations per box). There are several schemes that have been suggested for reducing the cost of applying translation operators. The simplest is based on rotating the coordinate system so that the vector connecting the source box B and the target box C lies along the z-axis, shifting the expansion along the z-axis, and then rotating back to the original coordinate system.

6.1. The FMM using rotation matrices

We begin with the following obvious result.

Lemma 6.1 Consider a harmonic function given by

$$\Phi(P) = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} \left(L_n^m r^n + \frac{M_n^m}{r^{n+1}} \right) Y_n^m(\theta, \phi),$$

where (r, θ, ϕ) are the spherical coordinates of the point P. If we rotate the coordinate system through an angle β in the positive sense about the z-axis, then

$$\Phi(P) = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} \left(\tilde{L}_n^m r^n + \frac{\tilde{M}_n^m}{r^{n+1}} \right) Y_n^m(\theta, \phi'),$$

where (r, θ, ϕ') are the new coordinates of P,

$$\tilde{L}_n^m = L_n^m \, e^{im\beta}, \quad \text{and} \quad \tilde{M}_n^m = M_n^m \, e^{im\beta}.$$

Definition 6.3 Given a rotation angle β, the diagonal operator mapping old multipole coefficients to rotated multipole coefficients ($O_n^m \to O_n^m \, e^{im\beta}$) will be denoted by $\mathcal{R}_z(\beta)$.

We also need to be able to rotate the coordinate system about the y-axis.

Lemma 6.2 Consider a harmonic function given by

$$\Phi(P) = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} \left(L_n^m r^n + \frac{M_n^m}{r^{n+1}} \right) Y_n^m(\theta, \phi),$$

where (r, θ, ϕ) are the spherical coordinates of the point P. If we rotate the coordinate system through an angle α in the positive sense about the y-axis, then there exist coefficients $R(n, m, m', \alpha)$ such that

$$\Phi(P) = \sum_{n=0}^{\infty} \sum_{m'=-n}^{n} \left(\tilde{L}_n^{m'} r^n + \frac{\tilde{M}_n^{m'}}{r^{n+1}} \right) Y_n^{m'}(\theta', \phi'),$$

where (r, θ, ϕ') are the new coordinates of P,

$$\tilde{L}_n^{m'} = \sum_{m=-n}^{n} R(n, m, m', \alpha) L_n^m \tag{6.3}$$

and

$$\tilde{M}_n^{m'} = \sum_{m=-n}^{n} R(n, m, m', \alpha) M_n^m. \tag{6.4}$$

Proof. See Biedenharn and Louck (1981) for a complete discussion and for a variety of methods that can be used to compute the coefficients $R(n, m, m', \alpha)$. □

Lemma 6.3 In order to shift a multipole expansion a distance ρ along the z-axis, one can replace equation (5.3) with the simpler formula

$$M_j^k = \sum_{n=0}^{j} \frac{O_{j-n}^k A_n^0 A_{j-n}^k \rho^n Y_n^0(1,0)}{A_j^k}. \tag{6.5}$$

In order to convert a multipole expansion centred at the origin into a local expansion centred at $(0,0,\rho)$, one can replace equation (5.6) with the simpler formula

$$L_j^k = \sum_{n=0}^{\infty} \frac{O_n^m A_n^k A_j^k Y_{j+n}^0(1,0)}{(-1)^n A_{j+n}^0 \rho^{j+n+1}}, \tag{6.6}$$

In order to translate the centre of a local expansion from the origin to the point $(0,0,\rho)$, one can replace equation (5.10) with the simpler formula

$$L_j^k = \sum_{n=j}^{p} \frac{O_n^m A_{n-j}^0 A_j^k Y_{n-j}^0(1,0)\rho^{n-j}}{(-1)^{n+j} A_n^k}, \tag{6.7}$$

Definition 6.4 Given a rotation angle α, the diagonal operator mapping old multipole coefficients to rotated multipole coefficients according to formula (6.3) or (6.4) will be denoted by $\mathcal{R}_y(\alpha)$. The special cases of the linear operators $\mathcal{T}_{MM}$, $\mathcal{T}_{ML}$, and $\mathcal{T}_{LL}$ which shift a distance ρ in the z-direction according to the formulae (6.5), (6.6), and (6.7) will be denoted by $\mathcal{T}_{MM}^z(\rho)$, $\mathcal{T}_{ML}^z(\rho)$, and $\mathcal{T}_{LL}^z(\rho)$.

We can now combine Lemmas 6.1, 6.2 and 6.3 to obtain the desired factorizations of $\mathcal{T}_{MM}$, $\mathcal{T}_{ML}$, $\mathcal{T}_{LL}$.

Lemma 6.4

$$\begin{aligned}
\mathcal{T}_{MM} &= \mathcal{R}_z(-\beta)\mathcal{R}_y(-\alpha)\mathcal{T}_{MM}^z(\rho)\mathcal{R}_y(\alpha)\mathcal{R}_z(\beta), \\
\mathcal{T}_{ML} &= \mathcal{R}_z(-\beta)\mathcal{R}_y(-\alpha)\mathcal{T}_{ML}^z(\rho)\mathcal{R}_y(\alpha)\mathcal{R}_z(\beta), \\
\mathcal{T}_{LL} &= \mathcal{R}_z(-\beta)\mathcal{R}_y(-\alpha)\mathcal{T}_{LL}^z(\rho)\mathcal{R}_y(\alpha)\mathcal{R}_z(\beta),
\end{aligned}$$

where (ρ,α,β) is the desired shifting vector.

Clearly, the cost of applying $\mathcal{T}_{MM}$, $\mathcal{T}_{ML}$, or $\mathcal{T}_{LL}$ by means of the preceding factorization is

$$O(p^2) + O(p^3) + O(p^3) + O(p^3) + O(p^2).$$

Thus, the total computational cost of the FMM can be reduced to approximately

$$191\left(\frac{N}{s}\right)3p^3 + 2Np^2 + 27Ns.$$

With $s = 3p^{3/2}$, the operation count becomes

$$270Np^{3/2} + 2Np^2. \tag{6.8}$$

7. Mathematical preliminaries III

Over the last few years, a number of 'fast' or diagonal translation schemes have been developed that require $O(p^2)$ work (Greengard and Rokhlin 1988*b*, Berman 1995, Elliott and Board 1996). Unfortunately, they are all subject to certain numerical instabilities. The instabilities can be overcome, but at additional cost, the details of which we leave to the cited papers.

The latest generation of fast algorithms is based on combining multipole expansions with exponential or 'plane wave' expansions. The reason for using exponentials is that translation corresponds to multiplication and, like the earlier fast schemes, requires only $O(p^2)$ work. Unlike in the earlier diagonal schemes, however, no numerical instabilities are encountered. The two-dimensional theory is described in Hrycak and Rokhlin (1995), and we present the three-dimensional theory here.

Remark 7.1 A complicating feature of the new approach is that *six* plane wave expansions will be associated with each box, one emanating from each face of the cube. To fix notation, we will refer to the $+z$ direction as *up*, to the $-z$ direction as *down*; to the $+y$ direction as *north*, to the $-y$ direction as *south*; to the $+x$ direction as *east*, and to the $-x$ direction as *west*. The interaction list for each box will be subdivided into six lists, one associated with each direction.

Definition 7.1 The *Uplist* for a box B consists of elements of the interaction list that lie *above* B and are separated by at least one box in the $+z$ direction (Figure 6). The *Downlist* for a box B consists of elements of the interaction list that lie *below* B and are separated by at least one box in the $-z$ direction. The *Northlist* for a box B consists of elements of the interaction list that lie *north* of B, are separated by at least one box in the $+y$ direction, and are not contained in the Up- or Downlists. The *Southlist* for a box B consists of elements of the interaction list that lie *south* of B, are separated by at least one box in the $-y$ direction, and are not contained in the Up- or Downlists. The *Eastlist* for a box B consists of elements of the interaction list that 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 Southlists. The *Westlist* for a box B consists of elements of the interaction list that 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 Southlists.

It is easy to verify that the original interaction list is the union of the Up-, Down-, North-, South-, East- and Westlists. It is also easy to verify that

$$\begin{aligned} C \in \mathrm{Uplist}(B) &\quad\Leftrightarrow\quad B \in \mathrm{Downlist}(C) \\ C \in \mathrm{Northlist}(B) &\quad\Leftrightarrow\quad B \in \mathrm{Southlist}(C) \\ C \in \mathrm{Eastlist}(B) &\quad\Leftrightarrow\quad B \in \mathrm{Westlist}(C). \end{aligned} \tag{7.1}$$

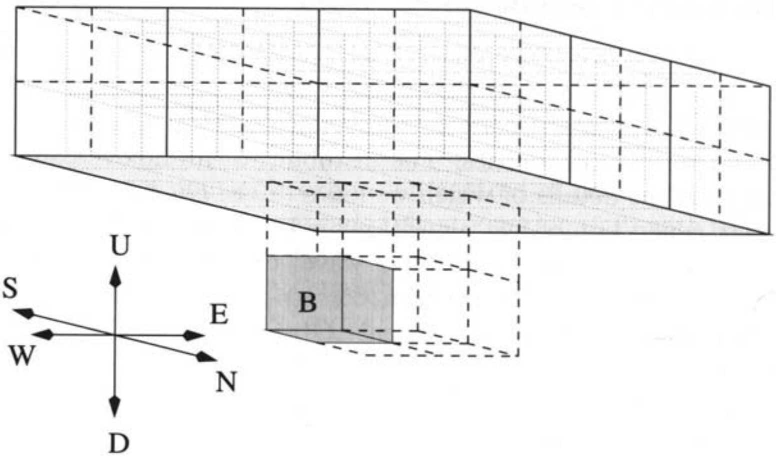

Fig. 6. The Uplist for the box B (see Definition 7.1)

Given a source location $P = (x_0, y_0, z_0)$ and a target location $Q = (x, y, z)$, our starting point is the well-known integral representation (Morse and Feshbach 1953, p. 1256)

$$\begin{aligned} &\frac{1}{\sqrt{(x-x_0)^2 + (y-y_0)^2 + (z-z_0)^2}} \\ &= \frac{1}{2\pi} \int_0^\infty e^{-\lambda(z-z_0)} \int_0^{2\pi} e^{i\lambda((x-x_0)\cos\alpha + (y-y_0)\sin\alpha)}\, \mathrm{d}\alpha\, \mathrm{d}\lambda \\ &= \int_0^\infty e^{-\lambda(z-z_0)} J_0(\lambda\sqrt{(x-x_0)^2 + (y-y_0)^2})\, \mathrm{d}\lambda, \qquad (7.2) \end{aligned}$$

valid for $z > z_0$.

To get a discrete representation, we must use an appropriate quadrature formula. The inner integral, with respect to α, is easily handled by the trapezoidal rule (which achieves spectral accuracy for periodic functions), but the outer integral requires more care. Laguerre quadrature is an appropriate choice here, but even better performance can be obtained using generalized Gaussian quadrature rules (Yarvin and Rokhlin 1996). These have been designed with the geometry of the interaction list in mind.

Because of the restriction that $z > z_0$, we will assume, for the moment, that the source P is contained in a box B and that the target Q lies in a box $C \in \text{Uplist}(B)$. The following lemma describes several discrete approximations of the double integral in (7.2) as double sums.

Lemma 7.1 Let $P \in B$ and $Q \in C \in \text{Uplist}(B)$, where B is a box of unit volume. Then

$$\left| \frac{1}{r_{PQ}} - \sum_{k=1}^{9} \frac{w_k}{M(k)} \sum_{j=1}^{M(k)} e^{-\lambda_k[(z-z_0)-i(x-x_0)\cos\alpha_j-(y-y_0)\sin\alpha_j]} \right| < 10^{-3}, \quad (7.3)$$

where $\alpha_j = 2\pi j/M(k)$, and the weights $w_1, \ldots, w_9$, nodes $\lambda_1, \ldots, \lambda_9$, and values $M(1), \ldots, M(9)$ are given in Section 12, Table 5. (The total number of exponentials required is 109.)

$$\left| \frac{1}{r_{PQ}} - \sum_{k=1}^{18} \frac{w_k}{M(k)} \sum_{j=1}^{M(k)} e^{-\lambda_k[(z-z_0)-i(x-x_0)\cos\alpha_j-(y-y_0)\sin\alpha_j]} \right| < 10^{-6}, \quad (7.4)$$

where $\alpha_j = 2\pi j/M(k)$, and the weights $w_1, \ldots, w_{18}$, nodes $\lambda_1, \ldots, \lambda_{18}$, and values $M(1), \ldots, M(18)$ are given in Section 12, Table 6. (The total number of exponentials required is 558.)

$$\left| \frac{1}{r_{PQ}} - \sum_{k=1}^{30} \frac{w_k}{M(k)} \sum_{j=1}^{M(k)} e^{-\lambda_k[(z-z_0)-i(x-x_0)\cos\alpha_j-(y-y_0)\sin\alpha_j]} \right| < 5 \times 10^{-11}, \quad (7.5)$$

where $\alpha_j = 2\pi j/M(k)$, and the weights $w_1, \ldots, w_{30}$, nodes $\lambda_1, \ldots, \lambda_{30}$, and values $M(1), \ldots, M(30)$ are given in Section 12, Table 7. (The total number of exponentials required is 1751.)

Remark 7.2 The formulae (7.3)–(7.5) are somewhat complex, but have a simple interpretation. The outer sums use the generalized Gaussian weights and nodes $\{w_k, \lambda_k\}$ obtained in Yarvin and Rokhlin (1996) to approximate the outer integral (with respect to λ), while the inner sums use the trapezoidal rule to approximate the inner integral (with respect to α). The number of nodes in each inner integral depends on the value λ_k for which the integration is being performed, and is denoted by $M(k)$. These are derived from standard estimates concerning Bessel functions (Watson 1944, pp. 227, 255; Rokhlin 1995).

Remark 7.3 In the remainder of this paper, we will assume that the desired precision ε is clear from the context, and will write

$$\left| \frac{1}{r_{PQ}} - \sum_{k=1}^{s(\varepsilon)} \sum_{j=1}^{M(k)} \frac{w_k}{M(k)} e^{-\lambda_k(z-z_0)} e^{i\lambda_k((x-x_0)\cos\alpha_j+(y-y_0)\sin\alpha_j)} \right| < \varepsilon, \quad (7.6)$$

where $\alpha_j = 2\pi j/M(k)$. This is a mild abuse of notation, since the weights, nodes and values $M(k)$ depend on ε as well. The total number of exponential basis functions used will be denoted by S_{exp}, so that

$$S_{\text{exp}} = \sum_{k=1}^{s(\varepsilon)} M(k).$$

Corollary 7.1 Let B be a box of unit volume centred at the origin containing N charges of strengths $\{q_l,\ l = 1, \ldots, N\}$, located at the points $\{Q_l = (x_l, y_l, z_l),\ l = 1, \ldots, N\}$. Then, for any P contained in Uplist(B), the potential $\Phi(P)$ satisfies

$$\left| \Phi(P) - \sum_{k=1}^{s(\varepsilon)} \sum_{j=1}^{M(k)} W(k,j) e^{-\lambda_k z} e^{i\lambda_k (x \cos\alpha_j + y \sin\alpha_j)} \right| < A\,\varepsilon, \tag{7.7}$$

where $A = \sum_{l=1}^{N} |q_l|$ and

$$W(k,j) = \sum_{l=1}^{N} q_i e^{\lambda_k z_l} e^{-i\lambda_k (x_l \cos\alpha_j + y_l \sin\alpha_j)}. \tag{7.8}$$

Corollary 7.2 (Diagonal translation) Let B be a box of unit volume centred at the origin containing N charges of strengths $\{q_l : l = 1, \ldots, N\}$, located at the points $\{Q_l = (x_l, y_l, z_l) : l = 1, \ldots, N\}$ and let C be a box in Uplist(B) centred at (x_1, y_1, z_1). For $P \in C$, let the potential $\Phi(P)$ be approximated by the exponential expansion centred at the origin

$$\Phi(P) = \sum_{k=1}^{s(\varepsilon)} \sum_{j=1}^{M(k)} W(k,j) e^{-\lambda_k z} e^{i\lambda_k (x \cos\alpha_j + y \sin\alpha_j)} + O(\varepsilon). \tag{7.9}$$

Then

$$\Phi(P) = \sum_{k=1}^{s(\varepsilon)} \sum_{j=1}^{M(k)} V(k,j) e^{-\lambda_k (z - z_1)} e^{i\lambda_k ((x - x_1)\cos\alpha_j + (y - y_1)\sin\alpha_j)} + O(\varepsilon), \tag{7.10}$$

where

$$V(k,j) = W(k,j)\, e^{-\lambda_k z_1} e^{i\lambda_k (x_1 \cos\alpha_j + y_1 \sin\alpha_j)}. \tag{7.11}$$

Definition 7.2 The diagonal operator mapping the original set of exponential expansion coefficients $\{W(k,j)\}$ to the shifted exponential expansion coefficients $\{V(k,j)\}$ according to (7.11) will be denoted by $\mathcal{D}_{\vec{BC}}$, where $\vec{BC} = (x_1, y_1, z_1)$ is the vector from the centre of B to the centre of C.

In the FMM, we will be given the multipole expansion of a charge distribution for a box B rather than the charge distribution itself, and will need to convert it to an exponential expansion. This is accomplished by the following theorem.

Theorem 7.1 Let B be a box of unit volume centred at the origin containing N charges of strengths $\{q_l,\ l=1,\dots,N\}$, located at the points $\{Q_l=(x_l,y_l,z_l),\ l=1,\dots,N\}$. Let $P \in C \in \text{Uplist}(B)$ and suppose that the potential $\Phi(P)$ is given as the multipole expansion

$$\Phi(P)=\sum_{n=0}^{\infty}\sum_{m=-n}^{n}\frac{M_n^m}{r^{n+1}}Y_n^m(\theta,\phi). \tag{7.12}$$

Then

$$\left|\Phi(P)-\sum_{k=1}^{s(\varepsilon)}\sum_{j=1}^{M(k)}W(k,j)e^{-\lambda_k z}e^{i\lambda_k(x\cos\alpha_j+y\sin\alpha_j)}\right|<A\,\varepsilon, \tag{7.13}$$

where $A=\sum_{l=1}^{N}|q_l|$ and

$$W(k,j)=\frac{w_k}{M(k)}\sum_{m=-\infty}^{\infty}(-i)^{|m|}e^{im\alpha_j}\sum_{n=|m|}^{\infty}\frac{M_n^m}{\sqrt{(n-m)!(n+m)!}}\lambda_k^n. \tag{7.14}$$

Proof. The formula (7.14) follows from the definitions (3.11) (3.12) and (3.13). The estimate (7.13) follows from Corollary 7.1. □

Definition 7.3 The linear operator mapping a finite multipole expansion $\{M_n^m : 0\le n\le p,\ -n\le m\le n\}$, to the corresponding set of coefficients in an exponential expansion $\{W(k,j)\}$ according to equation (7.14) will be denoted by $\mathcal{C}_{MX}$.

Once the multipole expansion for a source box has been converted into an exponential expansion (via Theorem 7.1) and translated to a target box centre (via Corollary 7.2), we will need to convert the exponential expansion back into a solid harmonic series. The following theorem provides the necessary machinery.

Theorem 7.2 Let B be a box of unit volume containing N charges of strengths $\{q_l,\ l=1,\dots,N\}$, located at the points $\{Q_l=(x_l,y_l,z_l),\ l=1,\dots,N\}$. Let P be contained in a box $C \in \text{Uplist}(B)$, centred at the origin, and suppose that the potential $\Phi(P)$ is given as the exponential expansion

$$\Phi(P)-\sum_{k=1}^{s(\varepsilon)}\sum_{j=1}^{M(k)}W(k,j)e^{-\lambda_k z}e^{i\lambda_k(x\cos\alpha_j+y\sin\alpha_j)}|<A\,\varepsilon, \tag{7.15}$$

where $A=\sum_{l=1}^{N}|q_l|$. Then

$$\left|\Phi(P)-\sum_{n=0}^{\infty}\sum_{m=-n}^{n}L_n^mY_n^m(\theta,\phi)r^n\right|<A\,\varepsilon, \tag{7.16}$$

where

$$L_n^m = \frac{(-i)^{|m|}}{\sqrt{(n-m)!(n+m)!}} \sum_{k=1}^{s(\varepsilon)} (-\lambda_k)^n \sum_{j=1}^{M(k)} W(k,j) e^{im\alpha_j}. \tag{7.17}$$

Proof. Equation (7.17) follows easily from the formula in Hobson (1955, p. 123),

$$(z + ix\cos\alpha + iy\sin\alpha)^n =$$
$$r^n \left\{ P_n(\cos\theta) + 2\sum_{m=1}^{n} (i)^{-m} \frac{n!}{(n+m)!} (-1)^m P_n^m(\cos\theta)\cos m(\phi - \alpha) \right\},$$

where (r, θ, ϕ) are the spherical coordinates of the point with Cartesian coordinates (x, y, z). □

Definition 7.4 The linear operator mapping the set of coefficients in an exponential expansion $\{W(k,j)\}$ to the coefficients in the corresponding truncated solid harmonic expansion $\{L_n^m : 0 \le n \le p, \ -n \le m \le n\}$, according to equation (7.17) will be denoted by $\mathcal{C}_{XL}$.

Remark 7.4 Theorems 7.1 and 7.2, like Theorem 5.2, are not quite the right tools needed to obtain rigorous error estimates for the FMM. In both cases, we have ignored the fact that the multipole and local expansions are truncated. It is straightforward but tedious to derive precise estimates, and we ignore this issue in the present paper. We should note that the nature of such estimates depends on how the multipole-to-exponential, multipole-to-solid harmonic or exponential-to-solid harmonic conversion is carried out. Formulae (7.14), (7.17) and (5.6) are the easiest to derive, being the Taylor expansions of the potential Φ. However, each of these conversions is simply a linear mapping from one set of basis functions to another. The formulae (7.17), (7.14), and (5.6) can be shown to correspond to minimizing the L_2 error on the surface of a *sphere* enclosing the given source or target box. One could choose a variety of other possible projections, such as minimizing the L_2 or L_∞ error on the surface of the corresponding box itself.

Remark 7.5 By inspection of formula (7.14), it is clear that the cost of applying the operator $\mathcal{T}_{MX}$ is $p^2\, s(\varepsilon) + p\, S_{\text{exp}}$. The same is true for the operator $\mathcal{T}_{XL}$. It is also worth noting that Fast Fourier Transforms can be used to reduce the cost of the outer sum in the truncated version of formula (7.14) and the inner sum in the truncated version of formula (7.17).

Corollary 7.3 (Multipole to local factorization) Let B be a box of unit volume and C a box in Uplist(B). If $\mathcal{T}_{ML}$ is the translation operator converting the multipole expansion centred in B to the local expansion centred in C, then

$$\mathcal{T}_{ML} = \mathcal{C}_{XL}\, \mathcal{D}_{\vec{BC}}\, \mathcal{C}_{MX}. \tag{7.18}$$

Remark 7.6 It is important to note that Lemma 7.1 provides a carefully designed quadrature formula which assumes that the source box B has unit volume and that the target is in B's Uplist. In order to use these quadrature weights and nodes, we need to rescale the multipole and local expansions so that the box dimension always has unit volume. To accomplish this, if

$$\Phi(P) = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} \frac{M_n^m}{r^{n+1}} Y_n^m(\theta, \phi) \tag{7.19}$$

is the multipole expansion for a box B of volume d^3, we simply write

$$\Phi(P) = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} \frac{(M_n^m/d^{n+1})}{(r/d)^{n+1}} Y_n^m(\theta, \phi). \tag{7.20}$$

The local expansion for a target box in B's interaction list is accumulated as

$$\Phi(P) = \sum_{j=0}^{\infty} \sum_{k=-j}^{j} L_j^k \, d^j Y_j^k(\theta, \phi) \left(\frac{r}{d}\right)^j . \tag{7.21}$$

Corollary 7.4 (Scaled multipole to local factorization) Let B be a box of volume d^3 and C a box in Uplist(B), with the vector from the centre of B to the centre of C given by (x_1, y_1, z_1). If $\mathcal{T}_{ML}$ is the translation operator converting the multipole expansion centred in B to the local expansion centred in C, then

$$\mathcal{T}_{ML} = \mathcal{D}_{d,L}\, \mathcal{C}_{XL} \mathcal{D}_{\overline{BC}} \mathcal{C}_{MX}\, \mathcal{D}_{d,M}, \tag{7.22}$$

where

$$\mathcal{D}_{d,M} M_n^m = M_n^m / d^{n+1}, \quad \mathcal{D}_{d,L} L_n^m = L_n^m / d^n,$$

and $\overline{BC} = \vec{BC}/d$.

The cost of a single multipole-to-local translation using the factorization of Corollary 7.4 is

$$2p^2 + 2p^2 s(\varepsilon) + 2pS_{\text{exp}} \approx 2p^3,$$

since $s \approx p$ and $S_{\text{exp}} \approx p^2$. If each translation were carried out in this manner, we would not improve on the rotation-based scheme discussed in Section 6.1. However, once the multipole expansion for a box B has been converted to an exponential expansion (via the application of $\mathcal{D}_{d,M}$ and $\mathcal{C}_{MX}$), it can be translated to each box in its Uplist at a cost of $S_{\text{exp}} \approx p^2$ operations. Conversely, once a box B has *accumulated* all the exponential expansions transmitted from its Downlist (see equation (7.1)), a single application of the operators $\mathcal{C}_{XL}$ and $\mathcal{D}_{d,L}$ yields the local harmonic expansion describing the field due to the sources in the Downlist of box B (Figure 7).

Up to this point, we have considered only the exponential representation

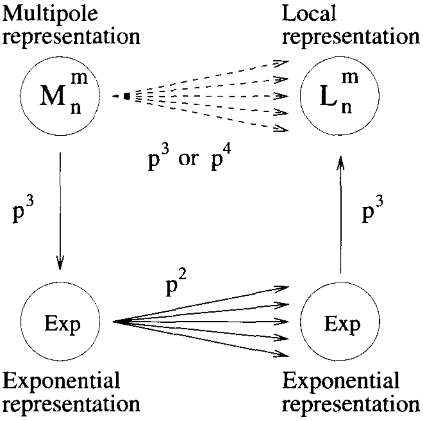

Fig. 7. In the new FMM, a large number of multipole-to-local translations, costing $O(p^3)$ or $O(p^4)$ work, can be replaced by a large number of exponential translations, costing $O(p^2)$ work

needed to shift information in the upward ($+z$) direction. As noted in the beginning of this section, however, there are six outgoing directions that need to be accounted for. The most straightforward way of generating the appropriate expansions is to rotate the coordinate system so that the z-axis points in the desired direction. The following lemma provides the necessary formulae.

Lemma 7.2 Let B be a box of volume d^3 and C a 'target' box. Let $\mathcal{T}_{ML}$ be the translation operator converting the multipole expansion centred in B to the local expansion centred in C.
If $C \in \text{Downlist}(B)$, then

$$\mathcal{T}_{ML}^{\text{Down}} = \mathcal{D}_{d,L}\,\mathcal{R}_y(-\pi)\,\mathcal{C}_{XL}\mathcal{D}_{\overline{BC}}\mathcal{C}_{MX}\,\mathcal{R}_y(\pi)\,\mathcal{D}_{d,M}.$$

If $C \in \text{Eastlist}(B)$, then

$$\mathcal{T}_{ML}^{\text{East}} = \mathcal{D}_{d,L}\,\mathcal{R}_y(-\pi/2)\,\mathcal{C}_{XL}\mathcal{D}_{\overline{BC}}\mathcal{C}_{MX}\,\mathcal{R}_y(\pi/2)\,\mathcal{D}_{d,M}.$$

If $C \in \text{Westlist}(B)$, then

$$\mathcal{T}_{ML}^{\text{West}} = \mathcal{D}_{d,L}\,\mathcal{R}_y(\pi/2)\,\mathcal{C}_{XL}\mathcal{D}_{\overline{BC}}\mathcal{C}_{MX}\,\mathcal{R}_y(-\pi/2)\,\mathcal{D}_{d,M}.$$

If $C \in \text{Northlist}(B)$, then

$$\mathcal{T}_{ML}^{\text{North}} = \mathcal{D}_{d,L}\,\mathcal{R}_y(-\pi/2)\,\mathcal{R}_z(-\pi/2)\,\mathcal{C}_{XL}\,\mathcal{D}_{\overline{BC}}\mathcal{C}_{MX}\,\mathcal{R}_y(\pi/2)\,\mathcal{R}_z(\pi/2)\,\mathcal{D}_{d,M}.$$

If $C \in \text{Southlist}(B)$, then

$$\mathcal{T}_{ML}^{\text{South}} = \mathcal{D}_{d,L}\,\mathcal{R}_y(\pi/2)\,\mathcal{R}_z(-\pi/2)\,\mathcal{C}_{XL}\,\mathcal{D}_{\overline{BC}}\mathcal{C}_{MX}\,\mathcal{R}_y(-\pi/2)\,\mathcal{R}_z(\pi/2)\,\mathcal{D}_{d,M},$$

where $\overline{BC}$ is the appropriately scaled vector from the centre of B to the centre of C in the rotated coordinate system. The operators $\mathcal{R}_z$ and $\mathcal{R}_y$ are defined in Section 6.1.

Definition 7.5 Let $\mathcal{T}_{ML}^{\mathrm{Up}}$ be given by the operator $\mathcal{T}_{ML}$ defined in equation (7.22). Then, for Dir $\in$ {Up, Down, East, West, North, South}, we will write

$$\mathcal{T}_{ML}^{\mathrm{Dir}} = \mathcal{Q}^{\mathrm{Dir}}\,\mathcal{D}_{\overline{BC}}\,\mathcal{P}^{\mathrm{Dir}},$$

so that

$$\begin{aligned}
\mathcal{Q}^{\mathrm{Up}} &= \mathcal{D}_{d,L}\,\mathcal{C}_{XL},\\
\mathcal{P}^{\mathrm{Up}} &= \mathcal{C}_{MX}\,\mathcal{D}_{d,M},\\
\mathcal{Q}^{\mathrm{Down}} &= \mathcal{D}_{d,L}\,\mathcal{R}_y(-\pi)\,\mathcal{C}_{XL},\\
\mathcal{P}^{\mathrm{Down}} &= \mathcal{C}_{MX}\,\mathcal{R}_y(\pi)\,\mathcal{D}_{d,M},
\end{aligned}$$

etc.

We are now in a position to describe the new FMM in detail.

8. The new FMM

ALGORITHM 2

The parent of a box j will be denoted by $p(j)$. The list of children of a box j will be denoted by $c(j)$. For each box j, the 'outgoing' exponential expansion with coefficients $\{W(n,m) : 1 \le n \le s(\varepsilon),\ 1 \le m \le M(n)\}$, will be denoted by W_j. We will also associate an 'incoming' exponential expansion with each box, denoted by V_j.

Upward pass

Initialization

Choose the number of refinement levels $n \approx \log_8 N$, and the order of the multipole expansion desired p. The number of boxes at the finest level is then 8^n, and the average number of particles per box is $s = N/(8^n)$.

Step 1

Form multipole expansions $\Phi_{n,i}$ of potential field due to particles in each box about the box centre at the finest mesh level, via Theorem 3.2.

Step 2

```
Do for levels $l = n-1, \ldots, 2$,
    Form multipole expansion $\Phi_{l,j}$ about the centre of each box at level $l$
    by merging expansions from its eight children via Theorem 5.1.
        $\Phi_{l,j} = \sum_{k \in c(j)} \mathcal{T}_{MM} \Phi_{l+1,k}$.
    (In applying $\mathcal{T}_{MM}$, use the factorization of Lemma 6.4.)
End do
```

Downward pass

Initialization

Set $\Psi_{1,1} = \Psi_{1,2} = \cdots = \Psi_{1,8} = (0, 0, \ldots, 0)$.

Step 3A

```
Do for levels $l = 2, \ldots, n$,
    Form the expansion $\tilde{\Psi}_{l,j}$ for each box $j$ at level $l$ by using
    Theorem 5.3 to shift the local $\Psi$ expansion of $j$'s parent to $j$ itself.
        $\tilde{\Psi}_{l,j} = \mathcal{T}_{LL} \Psi_{l-1,p(j)}$.
    (In applying $\mathcal{T}_{LL}$, use the factorization of Lemma 6.4.) Set $\Psi_{l.j} = \tilde{\Psi}_{l.j}$.
```

Step 3B

For each direction Dir = Up, Down, North, South, East, West, the opposite direction will be denoted by −Dir, so that −Up = Down, −Down = Up, *etc.* Thus, if a box B sends an outgoing expansion in direction Dir to Box C on its Dirlist, then C can be viewed as receiving the expansion from B which is an element of its −Dirlist (see equation (7.1)).

```
    Do for Dir = Up, Down, North, South, East, West,
        For each box $j$ at level $l$, convert the multipole expansion $\Phi_{l,j}$
        into the 'outgoing' exponential expansion for direction Dir.
            $W_j = \mathcal{P}^{\mathrm{Dir}} \Phi_{l,j}$.
        For each box $j$ at level $l$, collect the 'outgoing' exponential
        expansions from the −Dirlist of box $j$ as an 'incoming'
        exponential expansion
            $V_j = \sum_{k \in -\mathrm{Dirlist}} \mathcal{D}_{\tilde{kj}} W_k$,
        where $\tilde{kj}$ is the appropriately scaled vector from the centre of
        box $k$ to the centre of box $j$ in the rotated coordinate system.

        For each box $j$ at level $l$, convert the accumulated 'incoming'
        exponential expansion $V_j$ into a local harmonic expansion and
        add result to $\Psi_{l,j}$.
            $\Psi_{l,j} = \Psi_{l,j} + \mathcal{Q}^{\mathrm{Dir}} V_j$.
    End do
End do
```

Step 4

```
For each particle in each box $j$ at the finest level $n$,
    evaluate $\Psi_{n,j}$ at the particle position.
```

Step 5

```
For each particle in each box $j$ at the finest level $n$,
    compute interactions with particles in near neighbour boxes directly.
```

Since we are using the rotation scheme for applying $\mathcal{T}_{MM}$ and $\mathcal{T}_{LL}$ in Steps 2 and 3A, these now require a total of $3p^3\,(N/s)$ work, where s is the number of particles per box on the finest level. In Step 3B, the applications of the multipole to exponential operators $\mathcal{P}^{\mathrm{Dir}}$ and the exponential-to-local-operators $\mathcal{Q}^{\mathrm{Dir}}$ require a total of approximately $6\,p^3(N/s)$ work, while the exponential translations require approximately $189\,p^2\,(N/s)$ work. The total operation count is therefore of the order

$$189\,\frac{N}{s}\,p^2 + 2\,N\,p^2 + 27N\,s + 6\frac{N}{s}\,p^3.$$

With $s = 2p$, the total operation count is about

$$150\,N\,p + 5N\,p^2.$$

8.1. Current improvements

There are several ways in which the algorithm described above has been accelerated. Symmetry considerations, for example, allow the pairs of operators $\{\mathcal{P}^{\mathrm{Up}}, \mathcal{P}^{\mathrm{Down}}\}$, $\{\mathcal{P}^{\mathrm{North}}, \mathcal{P}^{\mathrm{South}}\}$, and $\{\mathcal{P}^{\mathrm{East}}, \mathcal{P}^{\mathrm{West}}\}$ to be applied simultaneously. The same is true for the adjoint pairs $\{\mathcal{Q}^{\mathrm{Up}}, \mathcal{Q}^{\mathrm{Down}}\}$, *etc.* Thus, the $6\,p^3(N/s)$ work needed in Step 3B can be replaced by $3\,p^3(N/s)$ work.

Even more significant is the fact that the number of translations per box can be reduced from 189 to less than 40. To see why, suppose that a box B at level l has eight children, denoted $B_1, \ldots, B_8$, and that boxes $C_1, \ldots, C_J$ lie in the Uplist of each child. In the new FMM described above, we accumulated an 'incoming' exponential expansion in each box C_j as

$$V_j = \sum_{k=1}^{8} \mathcal{D}_{\overline{B_k C_j}} W_k,$$

where W_k is the 'outgoing' exponential expansion for B_k. Repeating this for $j = 1, \ldots, J$ requires a total of $8\,J$ translations. Since all translations are diagonal, however, it is easy to verify that

$$\begin{aligned} V_j &= \sum_{k=1}^{8} \mathcal{D}_{\overline{BC_j}} \mathcal{D}_{\overline{B_k B}} W_k \\ &= \mathcal{D}_{\overline{BC_j}} \sum_{k=1}^{8} \mathcal{D}_{\overline{B_k B}} W_k. \end{aligned}$$

Thus, by first merging the 'outgoing' expansions, and then translating their sum to each target box C_j, only $8 + J$ translations are needed. It should be emphasized that this improvement relies on the diagonal form of the operators. One could try to merge expansions in this manner in the context of the original FMM, but the local expansion coefficients computed with and without merging would not be the same. There would be a significant loss of precision, consistent with the error bound (5.7).

8.2. Further improvements

There are several ways in which the scheme can be accelerated that have not been incorporated into the existing code. The most significant of these is probably a change in the choice of the translation operators $\mathcal{T}_{MM}$ and $\mathcal{T}_{LL}$, as well as the multipole-to-exponential and exponential-to-local conversion operators $\mathcal{C}_{MX}$ and $\mathcal{C}_{XL}$. As mentioned previously, the obvious formulae (5.3), (5.10), (7.14), and (7.17) are obtained via Taylor expansion and are clearly not optimal. Preliminary numerical experiments indicate that replacing them with more carefully optimized tools will reduce the cost of these calculations within the FMM by a factor of three. Furthermore, the improvement described in Remark 7.5 has not yet been implemented; we are using the explicit matrix form of the discrete Fourier transform in applying $\mathcal{C}_{MX}$ and $\mathcal{C}_{XL}$, rather than the FFT.

The incorporation of all these modifications is likely to reduce the overall cost by a factor of two.

9. Numerical results

The new FMM has been implemented in Fortran 77 and tested on uniform random distributions. The results of our experiments are summarized in Tables 1–4, with all times calculated in seconds using a Sun Ultra-1/140 workstation. In each table, the first column lists the number of particles, the second column lists the number of levels used in the multipole hierarchy, the third column lists the order of the multipole expansion used, and the fourth column lists the corresponding number of exponential basis functions. Columns five and six indicate the times required by the FMM and the direct calculation, respectively, and column seven lists the l^2 norm of the error in the FMM approximation

$$E = \left(\frac{\sum_{i=1}^{N} |\Phi(x_i) - \tilde{\Phi}(x_i)|^2}{\sum_{i=1}^{N} |\Phi(x_i)|^2} \right)^{1/2}. \tag{9.1}$$

For the largest simulations, with $N > 10000$, we have carried out the direct calculation on a subset of only 100 particles. The stated times, indicated in parentheses, are then computed by extrapolation and the errors are obtained by restricting the formula (9.1) to this subset.

Table 1. *Timing results for the FMM using fifth-order expansions and twenty-eight exponential basis functions*

N	Levels	p	S_{exp}	T_{FMM}	T_{dir}	Error
500	3	5	28	0.18	0.20	4.5×10^{-3}
5000	4	5	28	1.9	20.1	7.6×10^{-3}
40000	5	5	28	20	(1461)	7.0×10^{-3}
300000	6	5	28	175	(82475)	1.3×10^{-2}

Table 2. *Timing results for the FMM using ninth-order expansions and* 109 *exponential basis functions*

N	Levels	p	S_{exp}	T_{FMM}	T_{dir}	Error
2000	3	9	109	1.4	3.37	1.4×10^{-4}
10000	4	9	109	7.9	83	3.6×10^{-4}
80000	5	9	109	111	(5838)	4.1×10^{-4}

Table 3. *Timing results for the FMM using eighteenth-order expansions and* 558 *exponential basis functions*

N	Levels	p	S_{exp}	T_{FMM}	T_{dir}	Error
4000	3	18	558	8.3	13.4	1.1×10^{-7}
25000	4	18	558	68	(567)	1.5×10^{-7}
150000	5	18	558	495	(20100)	1.9×10^{-7}

Table 4. *Timing results for the FMM using thirtieth-order expansions and* 1751 *exponential basis functions*

N	Levels	p	S_{exp}	T_{FMM}	T_{dir}	Error
5000	3	30	1751	22	20.8	6.2×10^{-12}
50000	4	30	1751	316	(2280)	6.2×10^{-12}

10. Extensions and generalizations

The scheme presented in this paper is not adaptive and assumes that the distribution of points is reasonably uniform in space. In order to handle more general distributions, one needs to allow some regions to be subdivided into finer refinement levels than others. Adaptive structures of this type have been designed by several groups (Carrier et al. 1988, Van Dommelen and Rundensteiner 1989, Nabors et al. 1994) and we are in the process of incorporating these structures into the new FMM.

While a number of techniques now exist for high-frequency scattering problems (Rokhlin 1988, 1990, 1993, Canning 1989, 1992, 1993, Coifman and Meyer 1991, Bradie et al. 1993, Coifman et al. 1993, 1994, Wagner and Chew 1994, Epton and Dembart 1995), an important generalization of the algorithm of this paper is to the calculation of potentials governed by the Helmholtz equation at low frequency. By this we mean an environment in which the region of interest is no more than a few wavelengths in size, but contains a large number of discretization points (for example, due to the complexity of some structure being modelled). Algorithms for such problems are currently being designed.

11. Conclusions

A new version of the FMM has been developed. It is based on a new diagonal form for translation operators, and is significantly faster than previous implementations at any desired level of precision. Of particular interest is the fact that high precision calculations have been brought within practical reach.

12. Tables: quadrature weights and nodes

Table 5. *Columns 1 and 2 contain the nine weights and nodes needed for discretization of the outer integral in (7.2) at three-digit accuracy. Column 3 contains the number of discretization points needed in the inner integral, which we denote by $M(k)$*

Node	Weight	$M(k)$
0.09927399673971	0.24776441819008	4
0.47725674637049	0.49188566500464	7
1.05533661382183	0.65378749137677	11
1.76759343354008	0.76433038408784	15
2.57342629351471	0.84376180565628	20
3.44824339201583	0.90445883985098	20
4.37680983554726	0.95378613136833	24
5.34895757205460	0.99670261613218	7
6.35765785313375	1.10429422730252	1

Table 6. *Columns 1 and 2 contain the eighteen weights and nodes for discretization of the outer integral in (7.2) at six-digit accuracy. Column 3 contains the number of discretization points needed in the inner integral, which we denote by $M(k)$*

Node	Weight	$M(k)$
0.05278852766117	0.13438265914335	5
0.26949859838931	0.29457752727395	8
0.63220353174689	0.42607819361148	12
1.11307564277608	0.53189220776549	16
1.68939496140213	0.61787306245538	20
2.34376200469530	0.68863156078905	25
3.06269982907806	0.74749099381426	29
3.83562941265296	0.79699192718599	34
4.65424734321562	0.83917454386997	38
5.51209386593581	0.87570092283745	43
6.40421268377278	0.90792943590067	47
7.32688001906175	0.93698393742461	51
8.27740099258238	0.96382546688788	56
9.25397180602489	0.98932985769673	59
10.25560272374640	1.01438284597917	59
11.28208829787774	1.04003654374165	51
12.33406790967692	1.06815489269567	4
13.41492024017240	1.10907580975537	1

Table 7. *Columns 1 and 2 contain the thirty weights and nodes for discretization of the outer integral in (7.2) at ten-digit accuracy. Column 3 contains the number of discretization points needed in the inner integral, which we denote by* $M(k)$

Node	Weight	$M(k)$
0.03239542384523	0.08289159611006	7
0.16861844033714	0.18838810673274	10
0.40611377169029	0.28485143005306	14
0.73466473057596	0.37041553715895	18
1.14340561998398	0.44539043894975	22
1.62232408412252	0.51100452150290	26
2.16276138867422	0.56865283856139	30
2.75739199003682	0.61958013174010	35
3.40002470112078	0.66481004321965	39
4.08539104793552	0.70517204769960	43
4.80897515497095	0.74134967169016	48
5.56688915983444	0.77392103530415	53
6.35578243654166	0.80338600122756	57
7.17277232990713	0.83018277269650	62
8.01538803542112	0.85469824839953	66
8.88152313049502	0.87727539085565	71
9.76939480982937	0.89821948245755	76
10.67750922034750	0.91780416582368	80
11.60463289992789	0.93627766216629	85
12.54977061299652	0.95386940504388	89
13.51215012257297	0.97079739700556	94
14.49121482655196	0.98727684670885	97
15.48662587630224	1.00353112433459	103
16.49827659770404	1.01980697905712	107
17.52632405530625	1.03639774457222	110
18.57124579700721	1.05368191266322	112
19.63393428118300	1.07219343903929	108
20.71585163675095	1.09278318162014	84
21.81939113866225	1.11737373706779	4
22.95080495008893	1.15786184931141	1

REFERENCES

B. K. Alpert and V. Rokhlin (1991), 'A fast algorithm for the evaluation of Legendre expansions', *SIAM J. Sci. Statist. Comput.* **12**, 158–179.

B. K. Alpert, G. Beylkin, R. Coifman and V. Rokhlin (1993), 'Wavelet-like bases for the fast solution of second-kind integral equations', *SIAM J. Sci. Statist. Comput.* **14**, 159–184.

C. R. Anderson (1986), 'A method of local corrections for computing the velocity field due to a distribution of vortex blobs', *J. Comput. Phys.* **62**, 111–123.

C. R. Anderson (1992), 'An implementation of the fast multipole method without multipoles', *SIAM J. Sci. Statist. Comput.* **13**, 923–947.

A. W. Appel (1985), 'An efficient program for many-body simulation', *SIAM J. Sci. Statist. Comput.* **6**, 85–103.

J. Barnes and P. Hut (1986), 'A hierarchical $O(N \log N)$ force-calculation algorithm', *Nature* **324**, 446–449.

L. Berman (1995), 'Grid-multipole calculations', *SIAM J. Sci. Comput.* **16**, 1082–1091.

G. Beylkin, R. Coifman and V. Rokhlin (1991), 'Fast wavelet transforms and numerical algorithms I', *Comm. Pure Appl. Math.* **44**, 141–183.

L. C. Biedenharn and J. D. Louck (1981), *Angular Momentum in Quantum Physics: Theory and Application*, Addison Wesley, London.

J.A. Board, J. W. Causey, J. F. Leathrum, A. Windemuth and K. Schulten (1992), 'Accelerated molecular dynamics simulation with the parallel fast multipole method', *Chem. Phys. Let.* **198**, 89–94.

B. Bradie, R. Coifman and A. Grossmann (1993), 'Fast numerical computations of oscillatory integrals related to acoustic scattering, I', *Appl. Comput. Harm. Anal.* **1**, 94–99.

A. Brandt (1991), 'Multilevel computations of integral transforms and particle interactions with oscillatory kernels', *Comp. Phys. Comm.* **65**, 24–38.

A. Brandt and A. A. Lubrecht (1990), 'Multilevel matrix multiplication and fast solution of integral equations', *J. Comput. Phys.* **90**, 348–370.

F. X. Canning (1989), 'Reducing moment method storage from order N^2 to order N', *Electron. Let.* **25**, 1274–1275.

F. X. Canning (1992), 'Sparse approximation for solving integral equations with oscillatory kernels', *SIAM J. Sci. Statist. Comput.* **13**, 71–87.

F. X. Canning (1993), 'Improved impedance matrix localization method', *IEEE Trans. Antennas and Propagation* **41**, 658–667.

J. Carrier, L. Greengard and V. Rokhlin (1988), 'A fast adaptive multipole algorithm for particle simulations', *SIAM J. Sci. Statist. Comput.* **9**, 669–686.

R. Coifman and Y. Meyer (1991), 'Remarques sur l'analyse de Fourier à fenêtre', *C. R. Acad. Sci. Paris* **312**, Serie 1, 259–261.

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**, 7–12.

R. Coifman, V. Rokhlin and S. Wandzura (1994), 'Faster single-stage Multipole Method for the wave equation', 10th Annual Review of Progress in Applied Computational Electromagnetics, Vol. 1, pp. 19-24, Monterey, CA, Applied Computational Electromagnetics Society.

G. Dahlquist and A. Bjork (1974), *Numerical Methods* Prentice-Hall, Englewood Cliffs, NJ.

H.-Q. Ding, N. Karasawa and W. A. Goddard, III (1992), 'Atomic level simulations on a million particles: The Cell Multipole Method for Coulomb and London nonbond interactions, *J. Chem. Phys.* **97**, 4309–4315.

W. D. Elliott and J.A. Board (1996), 'Fast Fourier Transform accelerated fast multipole algorithm', *SIAM J. Sci. Comput.* **17**, 398–415.

M. A. Epton and B. Dembart (1995), 'Multipole translation theory for three-dimensional Laplace and Helmholtz equations', *SIAM J. Sci. Comput.* **16**, 865–897.

A. Greenbaum, L. Greengard and G. B. McFadden (1993), 'Laplace's equation and the Dirichlet–Neumann map in multiply connected domains', *J. Comput. Phys.* **105**, 267–278.

L. Greengard (1988), *The Rapid Evaluation of Potential Fields in Particle Systems*, MIT Press, Cambridge, MA.

L. Greengard (1990), 'The numerical solution of the N-body problem', *Computers in Physics* **4**, 142–152.

L. Greengard (1994), 'Fast algorithms for classical physics', *Science* **265**, 909–914.

L. Greengard and J.-Y. Lee (1996), 'A direct adaptive Poisson solver of arbitrary order accuracy', *J. Comput. Phys.* **125**, 415–424.

L. Greengard and M. Moura (1994), 'On the numerical evaluation of electrostatic fields in composite materials', in *Acta Numerica*, Vol. 3, Cambridge University Press, pp. 379–410.

L. Greengard and V. Rokhlin (1987), 'A fast algorithm for particle simulations', *J. Comput. Phys.* **73**, 325–348.

L. Greengard and V. Rokhlin (1988*a*), 'Rapid evaluation of potential fields in three dimensions', in *Vortex Methods*, C. Anderson and C. Greengard (eds.), Lecture Notes in Mathematics, vol. 1360, Springer, 121–141.

L. Greengard and V. Rokhlin (1988*b*), 'On the efficient implementation of the fast multipole algorithm', *Department of Computer Science Research Report 602, Yale University.*

L. Greengard and V. Rokhlin (1989), 'On the evaluation of electrostatic interactions in molecular modeling', *Chemica Scripta* **29A**, 139–144.

L. Greengard and J. Strain (1991), 'The fast Gauss transform', *SIAM J. Sci. Statist. Comput.* **12**, 79–94.

L. Greengard, M. C. Kropinski and A. Mayo (1996), 'Integral equation methods for Stokes flow and isotropic elasticity', *J. Comput. Phys.* **125**, 403–414.

M. Gu and S. C. Eisenstat (1992), 'A divide-and-conquer algorithm for the symmetric tridiagonal eigenproblem', *Department of Computer Science Research Report 932, Yale University.*

W. Hackbusch and Z. P. Nowak (1989), 'On the fast matrix multiplication in the boundary element method by panel clustering', *Numer. Math.* **54**, 463–491.

E. W. Hobson (1955), *Spherical and Ellipsoidal Harmonics*, Dover, New York.

R. W. Hockney and J. W. Eastwood (1981), *Computer Simulation Using Particles*, McGraw-Hill, New York.

T. Hrycak and V. Rokhlin (1995), 'An improved fast multipole algorithm for potential fields', *Department of Computer Science Research Report 1089, Yale University.*

J. D. Jackson (1975), *Classical Electrodynamics*, Wiley, New York.

O. D. Kellogg (1953), *Foundations of Potential Theory*, Dover, New York.

P. M. Morse and H. Feshbach (1953), *Methods of Theoretical Physics*, McGraw-Hill, New York.

K. Nabors and J. White (1991), 'FastCap: a multipole accelerated 3-D capacitance extraction program', *IEEE Trans. Computer-Aided Design* **10**, 1447–1459.

K. Nabors and J. White (1992), 'Multipole-accelerated capacitance extraction algorithms for 3-D structures with multiple dielectrics', *IEEE Trans. Circuits and Systems* **39**, 946–954.

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. Statist. Comput.* **15**, 714–735.

A. M. Odlyzko and A. Schönhage (1988), 'Fast algorithms for multiple evaluations of the Riemann zeta function', *Trans. Amer. Math. Soc.* **309**, 797–809.

H. G. Petersen, E. R. Smith and D. Soelvason (1995), 'Error estimates for the fast multipole method. II. The three-dimensional case', *Proc. R. Soc. London, Series A* **448**, 401–418.

V. Rokhlin (1985), 'Rapid solution of integral equations of classical potential theory', *J. Comput. Phys.* **60**, 187–207.

V. Rokhlin (1988), 'A fast algorithm for the discrete Laplace transformation', *J. Complexity* **4**, 12–32.

V. Rokhlin (1990*a*), 'End-point corrected trapezoidal quadrature rules for singular functions', *Computers Math. Applic.* **20**, 51–62.

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

V. Rokhlin (1993), 'Diagonal forms of translation operators for the Helmholtz equation in three dimensions', *Appl. Comput. Harm. Anal.* **1**, 82–93.

V. Rokhlin (1995), 'Sparse diagonal forms of translation operators for the Helmholtz equation in two dimensions, *Department of Computer Science Research Report 1095, Yale University.*

J. M. Song and W. C. Chew (1995), 'Multilevel fast multipole algorithm for solving combined field integral equations of electromagnetic scattering', *Microwave and Opt. Technol. Letters* **10** 14–19.

J. Strain (1991), 'The Fast Gauss Transform with variable scales', *SIAM J. Sci. Statist. Comput.* **12**, 1131–1139.

J. Strain (1992), 'The Fast Laplace Transform based on Laguerre functions', *Math. Comp.* **58**, 275–283.

L. Van Dommelen and E. A. Rundensteiner (1989), 'Fast, adaptive summation of point forces in the two-dimensional Poisson equation', *J. Comput. Phys.* **83**, 126–147.

P. R. Wallace (1984), *Mathematical Analysis of Physical Problems*, Dover, New York.

R. L. Wagner and W. C. Chew (1994), 'A ray-propagation fast multipole algorithm', *Microwave and Opt. Technol. Letters* **7** 348–351.

R. L. Wagner and W. C. Chew (1995), 'A study of wavelets for the solution of electromagnetic integral equations', *IEEE Antennas Propag.* **43** 802–810.

H. Y. Wang and R. LeSar (1995), 'An efficient fast-multipole algorithm based on an expansion in the solid harmonics', *J. Chem. Phys.* **104**, 4173–4179.

G. N. Watson (1944), *A Treatise on the Theory of Bessel Functions*, Cambridge University Press.

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.*