IEEE Trans. on Computer Aided Design of Integrated Cirucits and Systems, 10(11):1447–1459, November 1991.

FastCap: A Multipole Accelerated 3-D Capacitance Extraction Program

K. Nabors and J. White
Research Laboratory of Electronics
Dept. of Electrical Engineering and Computer Science
Massachusetts Institute of Technology
Cambridge, MA 02139 U.S.A.

Abstract

In this paper a fast algorithm for computing the capacitance of a complicated 3-D geometry of ideal conductors in a uniform dielectric is described and its performance in the capacitance extractor FastCap is examined. The algorithm is an acceleration of the boundary-element technique for solving the integral equation associated with the multiconductor capacitance extraction problem. Boundary-element methods become slow when a large number of elements are used because they lead to dense matrix problems which are typically solved with some form of Gaussian elimination. This implies that the computation grows as n^3, where n is the number of panels or tiles needed to accurately discretize the conductor surface charges. In this paper we present a generalized conjugate residual iterative algorithm with a multipole approximation to compute the iterates. This combination reduces the complexity so that accurate multiconductor capacitance calculations grow nearly as nm, where m is the number of conductors. Performance comparisons on integrated circuit bus crossing problems show that for problems with as few as twelve conductors the multipole accelerated boundary element method can be nearly 500 times faster than Gaussian elimination based algorithms, and five to ten times faster than the iterative method alone, depending on required accuracy.

1 Introduction

In the design of high performance integrated circuits and integrated circuit packaging, there are many cases where accurate estimates of the capacitances of complicated three

[0]This work was supported by the Defense Advanced Research Projects Agency contract N00014-87-K-825, the National Science Foundation and grants from I.B.M. and Analog Devices.

The authors are with the Research Laboratory of Electronics, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA, 02139, U.S.A.

dimensional structures are important for determining final circuit speeds or functionality. Two examples of complicated three-dimensional structures for which capacitance strongly affects performance are dynamic memory cells, and the chip carriers commonly used in high density packaging. In these problems, capacitance extraction is made tractable by assuming the conductors are ideal and are embedded in a piecewise-constant dielectric medium. Then to compute the capacitances, Laplace's equation is solved numerically over the charge free region with the conductors providing boundary conditions.

Although there are a variety of numerical methods that can be used to solve Laplace's equation, for three-dimensional capacitance calculations the usual approach is to apply a boundary-element technique to the integral form of Laplace's equation [14, 12, 11]. In these approaches the surfaces or edges of all the conductors are broken into small panels or tiles and it is assumed that on each panel i, a charge, q_i, is uniformly or piecewise linearly distributed. The potential on each panel is then computed by summing the contributions to the potential from all the panels using Laplace's equation Green's functions. In this way, a matrix of potential coefficients, P, relating the set of n panel potentials and the set of n panel charges is constructed. The resulting $n \times n$ system of equations must be solved to compute capacitances. Typically, Gaussian elimination or Cholesky factorization is used to solve the system of equations, in which case the number of operations is order n^3. Clearly, this approach becomes computationally intractable if the number of panels exceeds several hundred, and this limits the size of the problem that can be analyzed to one with a few conductors.

An approach to reducing the computation time that is particularly effective for computing the diagonal terms of the capacitance matrix, also referred to as the self-capacitances, is to ignore small contributions to the potential coefficient matrix due to pairs of panels which are separated by large distances [1]. In this paper we present a similar approach, which approximates small potential coefficients with multipole expansions. We show that this approach produces an algorithm which accurately computes both the self and coupling capacitances, and has a computational complexity of nearly mn, where m is the number of conductors. Our algorithm, which is really the pasting together of three well-known algorithms [13], is presented in three sections. To begin, in the next section one of the standard integral equation approaches is briefly described, and it is shown that the algorithm requires the solution of an $n \times n$ dense nearly symmetric matrix. Then, in Section 3, a generalized conjugate residual algorithm is described, and is shown to reduce the complexity of the calculation to roughly order mn^2. In Section 4, it is shown that the major computation of the conjugate residual algorithm, evaluation of a potential field from a charge distribution, can be computed in order n time using a multipole algorithm. In Section 5, we describe some experimental results and in Section 6 we present our conclusions and acknowledgments. Finally, some implementation details are presented in an appendix.

2 The Integral Equation Approach

Consider a system of m ideal conductors embedded in a uniform lossless dielectric medium. For such a system, the relation between the m conductor potentials, denoted

by $\hat{p} \in \Re^m$, and the m total charges on each conductor, denoted by $\hat{q} \in \Re^m$, is given by $\hat{q} = C\hat{p}$, where $C \in \Re^{m \times m}$ is referred to as the capacitance matrix. The j^{th} column of C can be calculated by solving for the total charges on each of the conductors when the j^{th} conductor is at unit potential, and all the other conductors are at zero potential. Then the charge on conductor i, $\hat{q}_i$, is equal to C_{ij}.

To find the conductor charge distributions given the conductor potentials, it is necessary to solve the first-kind integral equation

$$\psi(x) = \int_{surfaces} G(x, x')\sigma(x')da' \tag{1}$$

for the surface charge density σ, where x, $x' \in \Re^3$ and are positions in 3-space, da' is the incremental surface area, ψ is the surface potential and is known, and $G(x, x')$ is the Green's function, which is $\frac{1}{\|x-x'\|}$ in free space [1]. Here, $\|x - x'\|$ denotes the Euclidean distance between x and x'. Given the surface charge density σ, the total charge on the i^{th} conductor, Q_i, can be computed from

$$Q_i = \int_{i^{th}\ conductor's\ surface} \sigma(x')da'. \tag{2}$$

There are a variety of approaches for numerically computing the conductor surface charge density given the conductor potentials, some of which involve reformulating (1) as a partial differential equation, and using finite difference methods in three space dimensions [16, 4]. We will focus on the boundary-element methods applied directly to solving (1) [14, 12, 11], as they have proved to be efficient and accurate when applied to problems with ideal conductors in a uniform dielectric medium. These methods are also referred to as panel methods [6], or the method of moments [5], in other application domains. This class of method exploits the fact that the charge is restricted to the surface of the conductors, and rather than discretizing all of free space, just the surface charge on the conductors is discretized. The surface potential, which is known, is related to the discretized surface charge through integrals of Green's functions. The so-constructed system can then be solved for the discretized surface charge.

The simplest commonly used approach to constructing a system of equations that can be solved for the discretized surface charge is the "point-matching" or collocation method. In this method, the surfaces of m conductors in free space are discretized into a total of n 2-dimensional panels (See for example Fig. 5b). For each panel k, an equation is written that relates the potential at the center of that k^{th} panel to the sum of the contributions to that potential from the charge distribution on all n panels. That is,

$$p_k = \sum_{l=1}^{n} \int_{panel_l} \frac{\sigma_l(x')}{\|x' - x_k\|}\, da', \tag{3}$$

where x_k is the center of panel k, x' is the position on the surface of panel l, p_k is the potential at the center of panel k, and $\sigma_l(x')$ is the surface charge density on the l^{th} panel. The integral in (3) is the free space Green's function multiplied by the charge density

[1] Note that the scale factor $1/4\pi\epsilon_0$ can be ignored here, and reintroduced later to give the results in units of farads.

and integrated over the surface of the l^{th} panel. Note that as the distance between panel k and panel l becomes large compared to the surface area of panel l, the integral reduces to $\frac{q_l}{\|x_l - x_k\|}$ where x_l is the center of the l^{th} panel and q_l is the total charge on panel l.

In a first-order collocation method (higher order methods are rarely used), it is assumed that the surface charge density on the panel is constant [12]. In that case (3) can be simplified to

$$p_k = \sum_{l=1}^{n} \frac{q_l}{a_l} \int_{panel_l} \frac{1}{\|x' - x_k\|} \, da', \tag{4}$$

where a_l is the surface area of panel l. When applied to the collection of n panels, a dense linear system results,

$$Pq = p \tag{5}$$

where $P \in \Re^{n \times n}$; q, $p \in \Re^n$ and

$$P_{kl} = \frac{1}{a_l} \int_{panel_l} \frac{1}{\|x' - x_k\|} \, da'. \tag{6}$$

Note that q and p are the vectors of *panel* charges and potentials rather than the *conductor* charge and potential vectors, $\hat{q}$ and $\hat{p}$ mentioned above. The dense linear system of (5) can be solved, typically by some form of Gaussian elimination, to compute panel charges from a given set of panel potentials. To compute the j^{th} column of the capacitance matrix, (5) must be solved for q, given a p vector where $p_k = 1$ if panel k is on the j^{th} conductor, and $p_k = 0$ otherwise. Then the ij^{th} term of the capacitance matrix is computed by summing all the panel charges on the i^{th} conductor, that is

$$C_{ij} = \sum_{k \in conductor_i} q_k. \tag{7}$$

3 Solution by the Generalized Conjugate Residual Method

In order to solve for a complete $m \times m$ capacitance matrix, the $n \times n$ matrix of potential coefficients, P, must be factored once, usually into $P = LU$, where L and U are strictly lower and upper triangular respectively, and this requires order n^3 operations. Then, as there are m conductors, the factored system must solved m times with m different right-hand sides, and this requires order mn^2 operations. Since n is the total number of panels into which the conductor surfaces are discretized, m is necessarily much smaller than n. Therefore, the n^3 time for factorization dominates for large problems, but factorization can be avoided by using iterative methods to solve the m charge distribution problems.

From the definition P given by (6), it is clear that P is a positive nonsymmetric matrix and that the largest element in each row is the diagonal, though the matrix is *not* diagonally dominant. Therefore, conjugate-descent methods like the generalized conjugate residual (GCR) algorithm [15] given below in Algorithm 1, are likely to be more effective than the more familiar Gauss-Seidel or Gauss-Jacobi style algorithms.

Algorithm 1: GCR algorithm for solving $Pq = p$

/* The Setup. Note the u^{iter}'s are search directions and */
/* w is the residual. */
$w = p$; $q = 0$.
/* GCR Loop. */
For $iter = 0, 1, 2, ...$ until converged {
$u^{iter} = w$.
$pu^{iter} = Pw$.
/* P-orthogonalize pu^{iter} with respect to pu^m, $m < iter$. */
For $m = 0$ to $iter$ {
$\beta = pu^{iter\,T} pu^m$.
$u^{iter} = u^{iter} - \beta u^m$.
$pu^{iter} = pu^{iter} - \beta pu^m$.
}
/* Normalize the direction. */
$pu^{iter} = pu^{iter} / \|pu^{iter}\|$.
$u^{iter} = u^{iter} / \|pu^{iter}\|$.
/* Update the charge and the residual. */
$\alpha = w^T pu^{iter}$.
$q = q + \alpha u^{iter}$.
$w = w - \alpha pu^{iter}$.
}

4 Accelerating the Matrix-Vector Product

As can be seen from examining Algorithm 1, assuming the number of iterations required is small, the major costs of the GCR algorithm are initially forming the dense matrix P, and in each iteration computing the matrix-vector product Pw, both of which require order n^2 operations. Computing the capacitance matrix with Algorithm 1 is therefore at least order mn^2, and may be higher if the number of GCR iterations increases with the problem size. Note that if the number of panels per conductor is low, Algorithm 1 may not be much more efficient than using direct factorization.

An approach that avoids forming most of P, and reduces the cost of computing the matrix-vector product Pw, can be derived by recalling that if w is thought of as representing charges distributed on panels, then Pw is a potential due to that charge distribution. In addition, if the distance between the centers of panel i and panel j is large compared to the panel sizes, then $P_{ij} \approx \frac{1}{\|x_i - x_j\|}$. That is, for widely separated panels, the j^{th} panel charge has the same effect on the potential at x_i as would a point charge of value w_j located at panel j's center.

To see how these observations can help simplify the computation of Pw, consider the situation (depicted in 2-D for simplicity) in Figs. 1 and 2. In either figure, the obvious approach to determining the potential at the n_1 evaluation points from the n_2

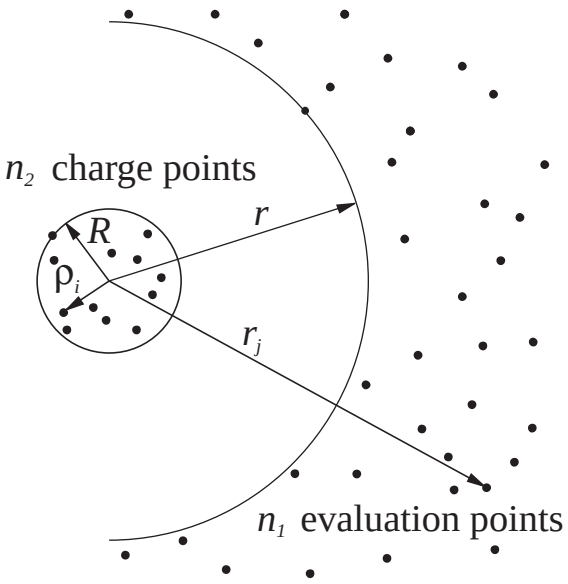

Figure 1: The evaluation point potentials are approximated with a multipole expansion.

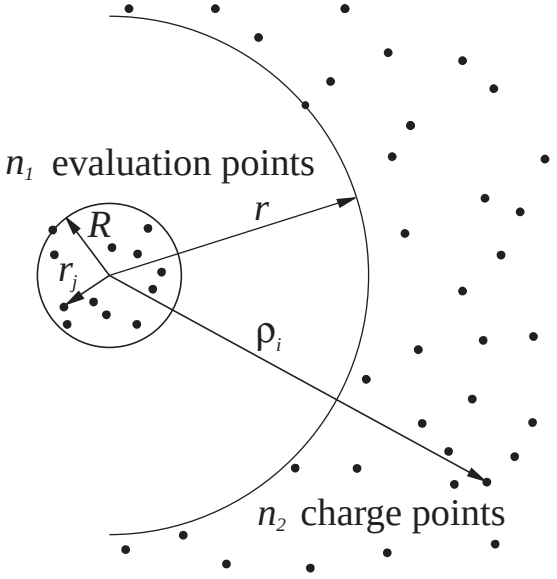

Figure 2: The evaluation point potentials are approximated with a local expansion.

point-charges involves $n_1 * n_2$ operations; at each of the n_1 evaluation points one simply sums the contribution to the potential from n_2 charges. An accurate approximation for the potentials for the case of Fig. 1 can be computed in many fewer operations using *multipole expansions*, which exploit the fact that $r >> R$ (defined in Fig. 1). That is, the details of the distribution of the charges in the inner circle of radius R in Fig. 1 do not strongly effect the potentials at the evaluation points outside the outer circle of radius r. It is also possible to compute an accurate approximation for the potentials at the evaluation points in the inner circle of Fig. 2 in many fewer than $n_1 * n_2$ operations using *local expansions*, which again exploit the fact that $r >> R$ (as in Fig. 2). In this second case, what can be ignored is the details of the evaluation point distribution.

4.1 Multipole Expansions

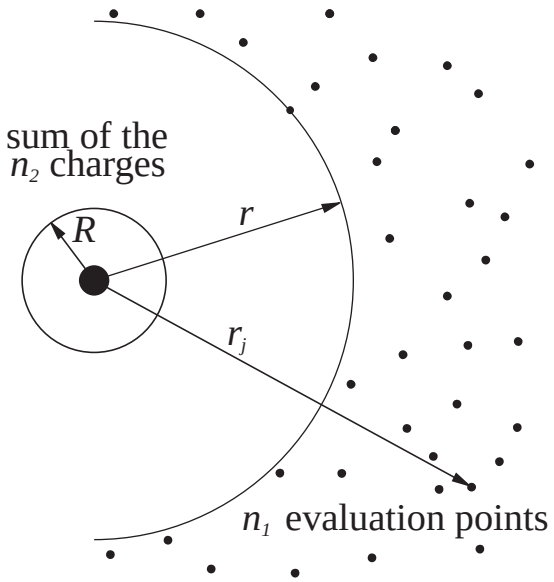

Figure 3: The charges are replaced by the first multipole expansion coefficient.

A rough approximation to the effect of the n_2 charges in the inner circle in Fig. 1 can be derived by replacing those charges with a single charge equal to their sum, placed at the inner circle's center (See Fig. 3). The number of operations to compute the n_1 potentials given this simplification is then $n_2 + n_1$, n_2 operations to compute the sum of charges, and n_1 operations to calculate the potentials at the evaluation points. Note that the accuracy of this approximation improves as the separation r between the nearest evaluation point and the center of the inner circle containing the charges increases compared to the inner circle's radius.

In the simplified approach above, the potential due to the charges in Fig. 1 is approximated by

$$\frac{\sum_{i=1}^{n_2} q_i}{r_j}, \tag{8}$$

where r_j is the distance between the center of the charge circle and the j^{th} evaluation point. Such an approximation is referred to as a monopole approximation, and is the first

term in the general multipole approximation for charge distributions. In general, the true potential, ψ, due to point charges inside a sphere at a location outside the radius of the sphere can be approximated arbitrarily accurately by a truncated *multipole* expansion,

$$\psi(r_j, \theta_j, \phi_j) \approx \sum_{n=0}^{l} \sum_{m=-n}^{n} \frac{M_n^m}{r_j^{n+1}} Y_n^m(\theta_j, \phi_j), \tag{9}$$

where l is the order of the expansion, r_j, θ_j and ϕ_j are the spherical coordinates of the j^{th} evaluation location referenced to the sphere's center. The $Y_n^m(\theta_j, \phi_j)$ factors are called spherical harmonics [2, 8] and the M_n^m are complex weights known as the multipole coefficients. The coefficients are related to the charges by

$$M_n^m = \sum_{i=1}^{n_2} q_i \rho_i^n Y_n^{-m}(\alpha_i, \beta_i), \tag{10}$$

where ρ_i, α_i, and β_i are the spherical coordinates of the i^{th} charge relative to the sphere's center. It has been shown that the truncated multipole expansion error is bounded by

$$|\psi(r_j, \theta_j, \phi_j) - \sum_{n=0}^{l} \sum_{m=-n}^{n} \frac{M_n^m}{r_j^{n+1}} Y_n^m(\theta_j, \phi_j)| < K_1 \left(\frac{R}{r_j}\right)^{l+1} \leq K_1 \left(\frac{R}{r}\right)^{l+1} \tag{11}$$

where K_1 is independent of l, and r and R are as in Figs. 1 and 3, the distance to the nearest evaluation point and the radius of the sphere of charge, respectively [2]. If the nearest evaluation point is well outside the sphere, then (11) implies that all the evaluation point potentials can be accurately computed using just a few terms of the multipole expansion.

4.2 Local Expansions

Multipole expansions cannot be used to simplify calculating the potentials for the evaluation points in the smaller circle of Fig. 2, as the charges are too widely distributed. However, it is still possible to compute approximate potentials at the n_1 evaluation points due to the n_2 charges in $n_1 + n_2$ operations. To see this, consider that the potential at any of the n_1 evaluation points in the smaller circle is roughly the same as the potential evaluated at the center of the circle. Thus the potential at an evaluation point can be approximated by

$$\sum_{i=1}^{n_2} \frac{q_i}{\rho_i}, \tag{12}$$

where ρ_i is the distance from the i^{th} charge to the center of the circle containing the evaluation points. Estimating the potentials at the n_1 evaluation points therefore requires n_2 operations to compute the potential at the circle's center by (12), and n_1 additional operations to copy that result to the n_1 evaluation points. Note that the approximation improves as the separation between the charges and the circle's center increases compared to the circle's radius.

Just as in the multipole case, it is possible to improve the accuracy of the above *local* expansion by including the effect of the distance between an evaluation point and the

enclosing sphere's center. In general, the truncated local expansion approximation for the exact potential in a sphere due to charges outside the radius of the sphere is given by

$$\psi(r_j, \theta_j, \phi_j) \approx \sum_{n=0}^{l} \sum_{m=-n}^{n} L_n^m Y_n^m(\theta_j, \phi_j) r_j^n, \tag{13}$$

where l is the order of the expansion, r_j, θ_j and ϕ_j are the spherical coordinates of the j^{th} evaluation location with respect to the sphere's center, and the L_n^m factors are the complex local expansion coefficients. For a set of n_2 charges outside the sphere, the local expansion coefficients are given by

$$L_n^m = \sum_{i=1}^{n_2} \frac{q_i}{\rho_i^{n+1}} Y_n^{-m}(\alpha_i, \beta_i), \tag{14}$$

where ρ_i, α_i, and β_i are the spherical coordinates of the i^{th} charge relative to the center of the sphere containing the evaluation points. As for the multipole expansion, the error introduced by the local expansion is related to a ratio of distances,

$$|\psi(r_j, \theta_j, \phi_j) - \sum_{n=0}^{l} \sum_{m=-n}^{n} L_n^m Y_n^m(\theta_j, \phi_j) r_j^n| < K_2 \left(\frac{r_j}{r}\right)^{l+1} \leq K_2 \left(\frac{R}{r}\right)^{l+1} \tag{15}$$

where K_2 is independent of l, and r and R are as in Fig. 2, the distance to the nearest charge location and the radius of the sphere of evaluation points, respectively [2]. Therefore, if the charges are well outside the sphere then the potential inside the sphere can be accurately computed using just a few terms of the local expansion.

4.3 The Multipole Algorithm

Low order multipole and local expansions can be used to accurately compute the potentials at n evaluation points due to n panel charges in order n operations, even for general evaluation point and charge distributions, but the multipole and local expansions have to be applied carefully, both to ensure adequate separation, and to avoid excess calculation. Below we give a multipole algorithm for computing the potentials at the n panel center points due to n panel charges. The algorithm requires $O(n)$ operations, and was originally presented in [2] with variants in [13, 9, 17]. The algorithm is reproduced here, modified to fit the boundary-element calculations.

To begin, the smallest cube that contains the entire collection of panels for the problem of interest is determined. This cube will be referred to as the level 0, or root, cube. Then, the volume of the cube is broken into eight equally sized child cubes, referred to as level 1 cubes, and each has the level 0 cube as its parent. The panels are divided among the child cubes by associating a panel with a cube if the panel's center point is contained in the cube. Each of the level 1 cubes is then subdivided into eight level 2 child cubes and the panels are again distributed based on their center point locations. The result is a collection of 64 level 2 cubes and a 64-way partition of the panels. This process is repeated to produce L levels of cubes, and L partitions of panels starting with an 8-way partition and ending with an 8^L-way partition. The number of levels, L, is chosen so that

the maximum number of panels in a finest, or L^{th}, level cube is less than some threshold (four is a typical default).

The following terms are used to concisely describe the multipole algorithm.

Definition 1 *Evaluation Points of a Cube: The center points of the panels associated with the cube.*

Definition 2 *Nearest Neighbors of a Cube: Those cubes which have a corner in common with the given cube.*

Definition 3 *Second Nearest Neighbors of a Cube: Those cubes which are not nearest neighbors but have a corner in common with a nearest neighbor.*

Note that there are at most 124 nearest and second nearest neighbors of a cube, excluding the cube itself.

Definition 4 *Interaction Cubes of a given cube: Those cubes which are either the second nearest neighbors of the given cube's parent, or are children of the given cube's parent's nearest neighbors, excluding nearest or second nearest neighbors of the given cube.*

There are a maximum of 189 interaction cubes for a given cube, roughly half are from a level one coarser than the level of the given cube, the rest are on the same level. The interaction cubes have two important properties. When combined with the given cube's nearest and second nearest neighbors, they entirely cover the same volume as the given cube's parent and the parent's nearest and second nearest neighbors. Also, the interaction cubes are such that the distance between a point in the given cube and a point in the interaction cube is more than half the distance between the centers of the given and interaction cubes. This latter property guarantees that when multipole expansions are used to approximate the effects of interaction cubes, and when these multipole expansions are gathered together in a local expansion for the given cube, the resulting approximation will converge rapidly with increasing expansion order.

Remark *As the charges in this problem are not point charges, but are distributed on panels, it is necessary to ensure that each panel is entirely contained in a finest level cube in order to ensure that evaluation points in a cube are well separated from panel charges in an interaction cube. This may require breaking a panel up into several panels, but as the multipole algorithm grows linearly with the number of panels, this is not a significant computational burden.*

The structure of the multipole algorithm for computing the n panel potentials from n panel charges is given below. The formulas for various transformations and shifts required are given in the appendix. A three letter key for each transformation is given to simplify finding the corresponding appendix formula.

Algorithm 2: Multipole Algorithm for Computing Pw

```
/*
THE DIRECT PASS: The potentials due to nearby charges are computed
directly.
*/
For each cube of the 8^L cubes on the finest level {
   /* Map panel charge distributions to potentials (Q2P). */
   Compute the potential at all the evaluation points in the cube
   due to the charge distributions on all the panels in the cube, in
   the cube's nearest neighbors, and in the cube's second nearest
   neighbors.
}

/*
THE UPWARD PASS: Computes a multipole expansion for every cube
at every level. The computation is order n because the multipole
expansion for any cube at a level coarser than the finest level is
computed by combining the multipole expansions of its children.
*/
For each cube of the 8^L cubes on the finest level {
   /* Map panel charges to multipole coefficients (Q2M). */
   Construct a multipole expansion for the charge distributions on all the
   panels in the cube, about the cube's center.
}

For each level i = L - 1 to 2 {
   For each cube of the 8^i cubes on level i {
      For each of the 8 children of the cube {
         /* Map multipole coefficients to multipole coefficients (M2M). */
         Shift the multipole expansion about the child cube's center
         to a multipole expansion about the cube's center and add it
         to the multipole expansion for the cube.
      }
   }
}

/*
THE DOWNWARD PASS: Computes a local expansion for every cube.
The local expansion includes the effects of all panel charges not in the
cube or its nearest and second nearest neighbors. Note that at the finest
level this includes the effects of all panels that are not treated in the
direct pass.
*/
For each level i = 2 to L {
```

```
    For each cube of the 8^i cubes on level i {
      /* Map local coefficients to local coefficients (L2L). */
      If the cube's parent has a local expansion, shift that expansion
      to a local expansion about the cube's center.

      For each of the cube's interaction cubes {
        /* Map multipole coefficients to local coefficients (M2L). */
        Convert the multipole expansion about the center of the
        interaction cube to a local expansion about the cube's
        center and add it to the local expansion for the cube.
      }
    }
  }

  /*
  THE EVALUATION PASS: Evaluates the local expansions at the finest level.
  */
  For each cube of the 8^L cubes on the lowest level {
    /* Map local coefficients into potentials (L2P). */
    Evaluate the cube's local expansion for the potential at all the
    evaluation points in the cube, and add those computed
    potentials to the evaluation point potentials.
  }
```

5 Implementation in FastCap

Our implementation of the multipole-accelerated capacitance extraction algorithm uses an optimization which exploits the fact that the conversion and shift operations are linear functions of the charges or the expansion coefficients, when the geometry is fixed. That is, the complicated evaluations involved in converting charges to potentials or multipole coefficients, shifting multipole coefficients, converting multipole coefficients to local coefficients, shifting local coefficients, and converting local coefficients to potentials, are all computed once, and stored as matrices which operate on charges or coefficients.

As an example, consider forming a 1^{st}-order multipole expansion for a collection of k charges. Following (9), a 1^{st}-order multipole expansion has the form

$$\psi(r,\theta,\phi) \approx M_0^0 \frac{Y_0^0(\theta,\phi)}{r} + M_1^{-1} \frac{Y_1^{-1}(\theta,\phi)}{r^2} + M_1^0 \frac{Y_1^0(\theta,\phi)}{r^2} + M_1^1 \frac{Y_1^1(\theta,\phi)}{r^2}, \qquad (16)$$

where M_0^0, ..., M_1^1 are complex multipole coefficients. Since M_n^0 is real for all n, and M_n^{-m} is always the complex conjugate of M_n^m, the multipole expansion can be written in terms of real coefficients as

$$\psi(r,\theta,\phi) \approx \bar{M}_0^0 \frac{P_0^0(\cos\theta)}{r} + \bar{M}_1^0 \frac{P_1^0(\cos\theta)}{r^2} + \bar{M}_1^1 \frac{P_1^1(\cos\theta)\cos\phi}{2r^2} + \check{M}_1^1 \frac{P_1^1(\cos\theta)\sin\phi}{2r^2}, \quad (17)$$

where $\bar{M}_0^0$, $\bar{M}_1^0$, $\bar{M}_1^1$ and $\tilde{M}_1^1$ are real coefficients and $P_n^m(\cos\theta)$ is the associated Legendre function of degree n and order m. This equation appears as (30) in the appendix, where it is discussed in more detail. This low order expansion can be more simply represented as

$$\psi(x,y,z) \approx \bar{M}_0^0 \frac{1}{r} + \bar{M}_1^0 \frac{z}{r^3} - \bar{M}_1^1 \frac{x}{2r^3} - \tilde{M}_1^1 \frac{y}{2r^3}, \tag{18}$$

where x, y and z are the evaluation point's cartesian coordinates and $r = \sqrt{x^2+y^2+z^2}$, as usual.

The real coefficients are calculated using the appendix formulas (32) and (33) which are analogous to (10). Writing the four equations for the four real coefficients as one matrix equation yields the $4 \times k$ linear system

$$\begin{bmatrix} P_0^0(\cos\alpha_1) & \cdots & P_0^0(\cos\alpha_k) \\ \rho_1 P_1^0(\cos\alpha_1) & \cdots & \rho_k P_1^0(\cos\alpha_k) \\ 2\rho_1 P_1^1(\cos\alpha_1)\cos\beta_1 & \cdots & 2\rho_k P_1^1(\cos\alpha_k)\cos\beta_k \\ 2\rho_1 P_1^1(\cos\alpha_1)\sin\beta_1 & \cdots & 2\rho_k P_1^1(\cos\alpha_k)\sin\beta_k \end{bmatrix} \begin{bmatrix} q_1 \\ \vdots \\ q_k \end{bmatrix} = \begin{bmatrix} \bar{M}_0^0 \\ \bar{M}_1^0 \\ \bar{M}_1^1 \\ \tilde{M}_1^1 \end{bmatrix}, \tag{19}$$

where $q_1, \ldots, q_k$ are the values of the k charges. The $4 \times k$ matrix is called the Q2M conversion matrix. Its i^{th} column depends only on the coordinates of the i^{th} charge. Substituting for the associated Legendre functions using (26) and (27) from the appendix and switching to rectangular coordinates simplifies the matrix to

$$\begin{bmatrix} 1 & \cdots & 1 \\ z_1 & \cdots & z_k \\ -2x_1 & \cdots & -2x_k \\ -2y_1 & \cdots & -2y_k \end{bmatrix} \begin{bmatrix} q_1 \\ \vdots \\ q_k \end{bmatrix} = \begin{bmatrix} \bar{M}_0^0 \\ \bar{M}_1^0 \\ \bar{M}_1^1 \\ \tilde{M}_1^1 \end{bmatrix}. \tag{20}$$

Note that the first row of the matrix implies that $\bar{M}_0^0$ is the sum of all the charge strengths, making it identical to the coefficient M_0^0 in (16).

Since the Q2M matrix is a function of the charge positions alone, its entries need be calculated only once if several multipole algorithm potential evaluations are required for the same charge geometry. In the notation of Algorithm 2, this amounts to using the multipole algorithm to compute several Pw products with the same P but with different w vectors. Each time the multipole algorithm is used to form a different Pw product, a new vector of charge strengths is multiplied by the Q2M matrix yielding a vector of updated multipole expansion coefficients. In a similar way, geometry dependent matrices for all the other multipole algorithm conversions and shifts can be constructed and used repeatedly in subsequent Pw product calculations.

In our implementation of the complete multipole-accelerated capacitance extraction algorithm, given below, the shift and conversion coefficients are computed once and stored.

Algorithm 3: Multipole-Accelerated Capacitance Extraction Algorithm

/* Setup Phase. */
Divide the m conductors into a total of n panels.

Divide the problem domain into a hierarchy of cubes, so that each
  of the finest level cubes has a maximum of 4 panels.
Compute the conversion and shifting matrices from the topology.
/* Loop Through all the Conductors. */
For $j = 1$ to m {
  /* Set the Potential of the Panels on Conductor j to one. */
  For $k = 1$ to n {
    If panel k is on conductor j, set $p_k = 1$.
    Else $p_k = 0$.
  }
  /* Solve for the Panel Charges using MGCR. */
  Use GCR (Algorithm 1) to solve $Pq = p$, using
    Multipole (Algorithm 2) to compute Pw.
  /* Sum the Charges on Conductor i to Compute C_{ij} */
  For $i = 1$ to m $C_{ij} = \sum_{k \in conductor_i} q_k$.
}

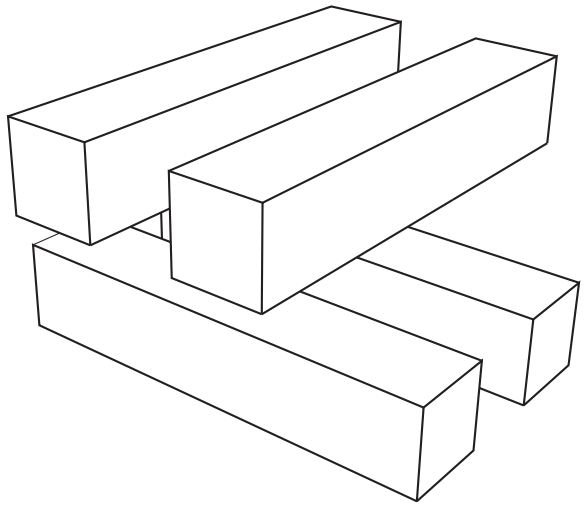

Figure 4: Bus structure test problem with 2×2 conductors.

To determine the effectiveness of this approach, the multipole accelerated algorithm was tested on the easily parameterized bus structure given in Fig. 4, for busses with 2×2 conductors through 6×6 conductors. The conductor surfaces are discretized by first cutting each conductor into sections based on where pairs of conductors overlap. In the 2×2 bus example, each conductor is cut into five sections (see Fig. 5a), and in the 6×6 example each conductor is divided into thirteen sections. The discretization is then completed by dividing each face of each section into nine panels, as demonstrated in Fig. 5b. The edge panels have widths that are 10% of the inner panel widths to accurately discretize the expected increased charge density near conductor edges [14].

In Table 1 below, we report the results of our experiments with the various approaches to solving (5), the matrix problem associated with the boundary element method. In the

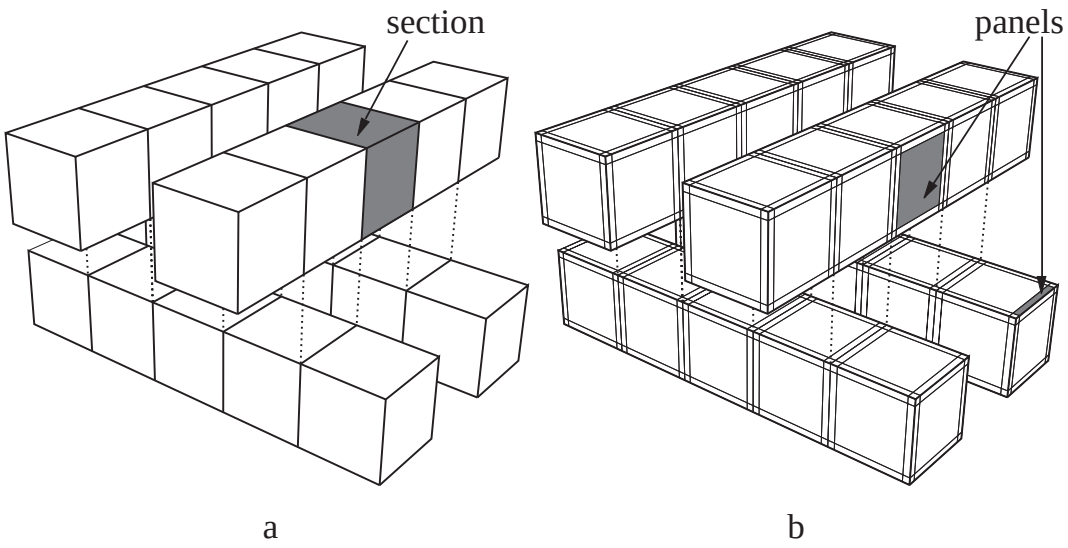

Figure 5: Conductor sections are divided into panels.

table, the total number of panels resulting from the conductor surface discretization is given, followed by the CPU times (on an I.B.M. 6000) required to compute the entire $m \times m$ capacitance matrix, where m is the number of conductors. Three methods for solving (5) are compared: direct or LU factorization, GCR, and multipole accelerated GCR (MGCR). The MGCR algorithm's CPU times are strongly dependent on the number of expansion terms, so the time required when zero, first and second order expansions ($l = 0,\ 1,\ 2$) are used is given. Also in the table are the total number of iterations required to reduce the max norm of the residual in GCR and MGCR to 1%. The CPU times in parenthesis are extrapolated. They correspond to calculations that were not possible because of the excessive memory required to store the entire potential coefficient matrix, and our lack of patience.

As is clear from the results, the multipole-accelerated GCR algorithm is very effective for the larger problems, particularly if the expansion order is low. To examine the effect of expansion order on accuracy, in Table 2 we compare the resulting capacitances computed by solving (5) for the 4×4 conductor problem with LU factorization, GCR and MGCR for expansion orders 0, 1, and 2. One row of the 4×4 capacitance matrix[2] is given for the five different solution methods. The row chosen represents the capacitance associated with one of the conductors on the outer edge. Taking the direct method results as exact, the data indicate that MGCR can attain better than 10% accuracy in the diagonal entry of the capacitance matrix with only a zero order ($l = 0$) expansion. To achieve reasonable relative accuracy in the smallest coupling capacitance, C_{14}, which is fifty times smaller than the diagonal entry, C_{11}, the 2^{nd}-order expansion is required. In that case MGCR produces results nearly identical to GCR indicating that further increases in accuracy would require tightening the iterative loop tolerance as well as increasing the expansion

[2]In the 4×4 conductor example the lengths have been normalized so that the conductors are each five meters long, one meter high and one meter wide, and all interconductor spaces are one meter. The capacitances are given in picofarads by scaling the program results by $4\pi\epsilon_0 = 111.27$ pF/m.

	Test Problem				
	2×2	3×3	4×4	5×5	6×6
panels	792	1620	2736	4140	5832
direct time	275	2700	12969	44345	(141603)
GCR time	121	570	2115	4881	(14877)
MGCR time ($l = 2$)	55	218	378	790	1412
MGCR time ($l = 1$)	29	108	245	436	775
MGCR time ($l = 0$)	9	48	98	216	356
GCR iters	48	78	120	150	(180)
MGCR iters ($l = 2$)	48	82	120	150	180
MGCR iters ($l = 1$)	54	88	120	150	180
MGCR iters ($l = 0$)	58	90	120	150	180

Table 1: Comparison of Extraction Methods, CPU Times in I.B.M. 6000 Seconds

Solution Method	Capacitance Matrix Entry (pF)							
	C_{11}	C_{12}	C_{13}	C_{14}	C_{15}	C_{16}	C_{17}	C_{18}
direct	404.6	−137.0	−12.04	−7.910	−48.42	−40.09	−40.09	−48.42
GCR	404.2	−137.2	−11.64	−8.083	−48.37	−39.93	−39.93	−48.37
MGCR ($l = 2$)	405.2	−137.8	−11.91	−8.079	−48.36	−40.09	−40.01	−48.45
MGCR ($l = 1$)	406.6	−139.7	−12.36	−6.676	−48.48	−40.45	−40.27	−48.46
MGCR ($l = 0$)	394.5	−124.0	−0.175	−2.471	−52.15	−43.39	−43.08	−52.92

Table 2: Comparison of Extraction Methods, 4×4 Conductor Problem Capacitances

order.

As mentioned above, computing the potentials given a new charge vector using the multipole algorithm just involves applying mapping matrices to the changing charges or multipole and local coefficients. Viewed in this way, the multipole algorithm can be compared more precisely to explicitly computing the matrix-vector product Pw by counting the number of multiply-add operations required in each case. This is a way of comparing MGCR to GCR that eliminates the effects of implementation differences. The number of multiply-add operations required by the four iterative methods of Tables 1 and 2 for each Pw product is plotted in Fig. 6, as a function of problem size. As seen in the plot, all the MGCR methods require fewer operations than GCR for each Pw product when the number of panels, n, exceeds approximately 1000. Furthermore, the cost of a Pw calculation grows as n^2 for GCR but roughly as n for the MGCR methods, with the expansion order affecting only the slope.

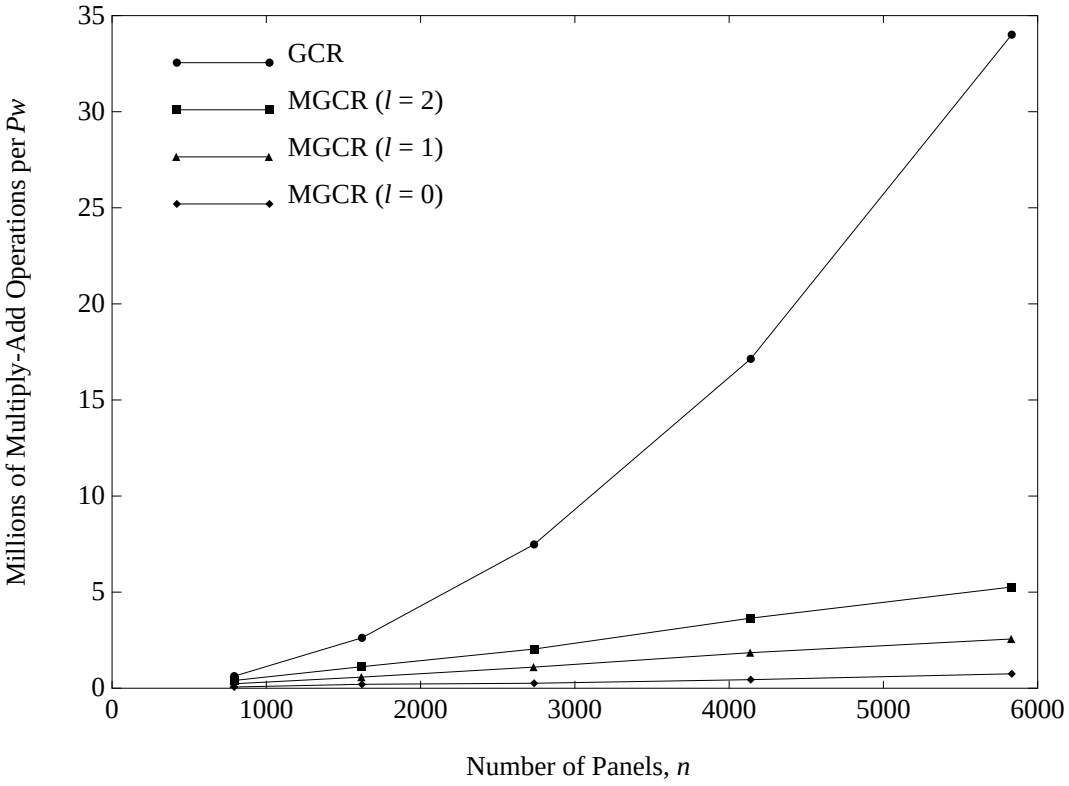

Figure 6: Multiply-Add operations required for a single Pw calculation as a function of problem size.

6 Conclusions and Acknowledgments

The above results indicate that the multipole-accelerated GCR (MGCR) algorithm is faster than both *LU* factorization and GCR alone for problems with more than 1000 panels. Capacitance extraction problems with nearly 6000 panels can be solved using MGCR in about twenty-five minutes on a workstation, roughly ten times faster than GCR, with comparable accuracy. Furthermore, the multipole algorithm provides an approach for trading accuracy for CPU time through reducing the expansion order; if low accuracy is tolerable, another factor of three or four speed improvement can be obtained. Finally, MGCR does not require explicit storage of the entire potential coefficient matrix as do GCR and *LU* factorization, resulting in significantly smaller memory requirements. Combining these features makes it feasible to perform capacitance extraction of complicated structures as the "inner loop" of a design optimization procedure.

Additional work is under way to improve the efficiency of the multipole algorithm, and we are currently working on making the algorithm adaptive. Future research includes investigating using block iterative techniques to reduce the number of iterations required and extending the approach to solving problems with piecewise-constant dielectrics and ground planes. In addition, we will extend our program to handle arbitrary triangular and quadrilateral panels so that more esoteric structures can be discretized.

The authors would like to thank David Ling and Albert Ruehli of the I.B.M. T. J. Watson Research Center for the many discussions and their help in understanding the integral equation method, and Albert Ruehli for the suggestion that led to the approach presented here. In addition, we would like to acknowledge the helpful discussions with Tom Simon and the members of the M.I.T. custom integrated circuits group.

A Multipole Algorithm Formulas

This appendix presents the multipole algorithm expansion, shift and conversion formulas used in the capacitance extraction algorithm implementation. The formulas used are equivalent to those in the original multipole algorithm formulation of [2, 3] but avoid complex arithmetic. They are obtained by combining complex conjugates in the original formulas to obtain expressions in the style of [7]. Section A.1 defines the real valued coefficients and the spherical harmonics which together are used to form the multipole formulas of Section A.2.

A.1 Formula Components

Each multipole or local expansion term involves a coefficient multiplying a spherical harmonic. When a real coefficient expansion is used this fact is obscured by the combination of complex conjugates. However, since the real coefficient expansions are just reorganizations of the complex coefficient formulas, the same coefficients and spherical harmonics appear in slightly different form.

A.1.1 Real Valued Expansion Coefficients

Given a multipole or local expansion coefficient, G_n^m, the corresponding real valued coefficients are defined as

$$\bar{G}_n^m \triangleq \begin{cases} 2\sqrt{\frac{(n+|m|)!}{(n-|m|)!}}\,\mathrm{Re}\{G_n^m\}, & |m|>0,\ |m|\le n; \\ G_n^m, & m=0,\ m\le n; \\ 0, & \text{otherwise;} \end{cases} \tag{21}$$

$$\tilde{G}_n^m \triangleq \begin{cases} -2\sqrt{\frac{(n+m)!}{(n-m)!}}\,\mathrm{Im}\{G_n^m\}, & m>0,\ m\le n; \\ 2\sqrt{\frac{(n+|m|)!}{(n-|m|)!}}\,\mathrm{Im}\{G_n^m\}, & m<0,\ |m|\le n; \\ 0, & \text{otherwise.} \end{cases} \tag{22}$$

All the multipole algorithm formulas are converted to real coefficients using these substitutions for the complex coefficients.

A.1.2 Spherical Harmonics

The functions

$$Y_n^m(\theta,\phi) \triangleq \sqrt{\frac{(n-|m|)!}{(n+|m|)!}}P_n^{|m|}(\cos\theta)e^{im\phi}, \tag{23}$$

are called surface spherical harmonics. A surface spherical harmonic is part of a solution to Laplace's equation obtained by separation of variables. Here, as in [8], a surface spherical harmonic is the product of the elevation (θ) and azimuth (ϕ) components of the solution. Unlike the usual definition, however, a normalization constant is omitted following [2, 3]. The complete solution is a product of Y_n^m and a power of r, the radial coordinate. The result, for example

$$\frac{1}{r^{n+1}}Y_n^m(\theta,\phi), \tag{24}$$

is called a spherical harmonic.

The function $P_n^m(\cos\theta)$ is the associated Legendre function of the first kind with degree n and order m. These functions are defined only when n is a non-negative integer and m is an integer such that $-n \le m \le n$. For convenience any $P_n^m(\cos\theta)$ whose indices do not satisfy these restrictions is taken to be zero.

The recursion

$$\begin{aligned}(n-m)&P_n^m(\cos\alpha) \\ &= (2n-1)\,\cos\alpha\,P_{n-1}^m(\cos\alpha) - (n+m-1)P_{n-2}^m(\cos\alpha),\end{aligned} \tag{25}$$

valid for $0 \le m \le n-2$, and the formulas

$$P_m^m(\cos\alpha) = \frac{(2m)!}{2^m m!}(-\sin\alpha)^m, \qquad 0 \le m, \tag{26}$$

$$P_{m+1}^m(\cos\alpha) = (2m+1)\cos\alpha P_m^m(\cos\alpha), \qquad 0 \le m, \tag{27}$$

can be used to recursively evaluate the Legendre functions [7, 10].

A.2 Real Coefficient Multipole Algorithm Formulas

Using the real valued coefficients and the spherical harmonics of the previous section, the multipole algorithm formulas used in the capacitance extraction algorithm are obtained. The resulting real coefficient formulas eliminate the need for complex arithmetic and square root calculations.

A.2.1 Multipole Expansion (Q2M, M2P)

The order l multipole expansion approximation to the potential, ψ, at the point (r, θ, ϕ) is

$$\psi(r,\theta,\phi) \approx \sum_{n=0}^{l} \sum_{m=-n}^{n} \frac{M_n^m}{r^{n+1}} Y_n^m(\theta,\phi). \tag{28}$$

Applying the definition of the surface spherical harmonic, Y_n^m, gives

$$\psi(r,\theta,\phi) \approx \sum_{n=0}^{l} \sum_{m=-n}^{n} \frac{M_n^m}{r^{n+1}} \sqrt{\frac{(n-|m|)!}{(n+|m|)!}} P_n^{|m|}(\cos\theta) e^{im\phi}. \tag{29}$$

Substituting the real coefficients using (21) and (22) yields the real coefficient multipole expansion,

$$\psi(r,\theta,\phi) \approx \sum_{n=0}^{l} \frac{1}{r^{n+1}} \sum_{m=0}^{n} \frac{(n-m)!}{(n+m)!} P_n^m(\cos\theta) \left[\bar{M}_n^m \cos(m\phi) + \tilde{M}_n^m \sin(m\phi) \right]. \tag{30}$$

The complex coefficient local expansion conversion is nearly identical.

A multipole expansion is constructed from k charges with strengths q_i and positions $(\rho_i, \alpha_i, \beta_i)$, $i = 1, \ldots, k$, using

$$M_n^m \triangleq \sum_{i=1}^{k} q_i \, \rho_i^n \, Y_n^{-m}(\alpha_i, \beta_i). \tag{31}$$

Substituting (21) and (22) gives expressions for the real multipole coefficients corresponding to a set of k charges,

$$\bar{M}_n^m = \begin{cases} 2\sum_{i=1}^{k} q_i \, \rho_i^n \, P_n^{|m|}(\cos\alpha_i) \cos(m\beta_i), & |m| > 0, \ |m| \le n; \\ \sum_{i=1}^{k} q_i \, \rho_i^n \, P_n^0(\cos\alpha_i), & m = 0, \ m \le n; \\ 0, & \text{otherwise}; \end{cases} \tag{32}$$

$$\tilde{M}_n^m = \begin{cases} 2\sum_{i=1}^{k} q_i \, \rho_i^n \, P_n^{|m|}(\cos\alpha_i) \sin(m\beta_i), & |m| > 0, \ |m| \le n; \\ 0, & \text{otherwise}. \end{cases} \tag{33}$$

A.2.2 Local Expansion (Q2L, L2P)

The order l local expansion approximation to the potential, ψ, at the point (r, θ, ϕ) is

$$\psi(r,\theta,\phi) \approx \sum_{n=0}^{l} \sum_{m=-n}^{n} L_n^m Y_n^m(\theta,\phi) r^n, \tag{34}$$

or

$$\psi(r,\theta,\phi) \approx \sum_{n=0}^{l} \sum_{m=-n}^{n} L_n^m \sqrt{\frac{(n-|m|)!}{(n+|m|)!}} P_n^{|m|}(\cos\theta) e^{im\phi} r^n. \tag{35}$$

Substituting the real coefficients using (21) and (22) yields the real coefficient local expansion,

$$\psi(r,\theta,\phi) \approx \sum_{n=0}^{l} r^n \sum_{m=0}^{n} \frac{(n-m)!}{(n+m)!} P_n^m(\cos\theta) \left[\bar{L}_n^m \cos(m\phi) + \tilde{L}_n^m \sin(m\phi)\right]. \tag{36}$$

A local expansion is constructed from k charges with strengths q_i and positions $(\rho_i, \alpha_i, \beta_i)$, $i = 1, \ldots, k$, using

$$L_n^m \triangleq \sum_{i=1}^{k} \frac{q_i}{\rho_i^{n+1}} Y_n^{-m}(\alpha_i, \beta_i). \tag{37}$$

Substituting (21) and (22) gives expressions for the real multipole coefficients corresponding to a set of charges,

$$\bar{L}_n^m = \begin{cases} 2\sum_{i=1}^{k} \frac{q_i}{\rho_i^{n+1}} P_n^{|m|}(\cos\alpha_i) \cos(m\beta_i), & |m| > 0,\ |m| \le n; \\ \sum_{i=1}^{k} \frac{q_i}{\rho_i^{n+1}} P_n^0(\cos\alpha_i), & m = 0,\ m \le n; \\ 0, & \text{otherwise}; \end{cases} \tag{38}$$

$$\tilde{L}_n^m = \begin{cases} 2\sum_{i=1}^{k} \frac{q_i}{\rho_i^{n+1}} P_n^{|m|}(\cos\alpha_i) \sin(m\beta_i), & |m| > 0,\ |m| \le n; \\ 0, & \text{otherwise}. \end{cases} \tag{39}$$

A.2.3 Multipole Expansion Shift (M2M)

Consider a multipole expansion about the point (ρ, α, β). The potential at a given point results when its coordinates relative to (ρ, α, β) are substituted into the expansion. If the expansion about (ρ, α, β) has coefficients O_n^m, then the coefficients of a shifted multipole expansion about the origin, N_j^k, are given by

$$N_j^k = \sum_{n=0}^{j} \sum_{m=-n}^{n} \frac{\sqrt{(j+k)!(j-k)!}\, i^{|k|-|m|-|k-m|}\, Y_n^{-m}(\alpha,\beta)\, O_{j-n}^{k-m}\, \rho^n}{\sqrt{(j-n+k-m)!\,(j-n-k+m)!\,(n+m)!\,(n-m)!}}. \tag{40}$$

Substituting for the surface spherical harmonics using (23) and for the complex coefficients with (21) and (22) gives the real coefficient multipole expansion shift formulas for $j \ge k \ge 0$,

$$\bar{N}_j^k = (j+k)! \sum_{n=0}^{j} \rho^n \sum_{m=0}^{n} f_M(m,k) \frac{P_n^m(\cos\alpha)}{(n+m)!} \cdot \left\{ \frac{i^{k-m-|k-m|}}{(j-n+|k-m|)!} \left[\bar{O}_{j-n}^{k-m} \cos(m\beta) - \tilde{O}_{j-n}^{k-m} \sin(m\beta)\right] \right.$$

$$+\frac{(-1)^m}{(j-n+k+m)!}\left[\bar{O}_{j-n}^{k+m}\cos(m\beta)+\tilde{O}_{j-n}^{k+m}\sin(m\beta)\right]\Bigg\}; \tag{41}$$

$$\tilde{N}_j^k = (j+k)!\sum_{n=0}^{j}\rho^n\sum_{m=0}^{n} f_M(m,k)\frac{P_n^m(\cos\alpha)}{(n+m)!}$$
$$\cdot\left\{\frac{i^{k-m-|k-m|}}{(j-n+|k-m|)!}\left[\bar{O}_{j-n}^{k-m}\sin(m\beta)+\tilde{O}_{j-n}^{k-m}\cos(m\beta)\right]\right.$$
$$\left.+\frac{(-1)^m}{(j-n+k+m)!}\left[-\bar{O}_{j-n}^{k+m}\sin(m\beta)+\tilde{O}_{j-n}^{k+m}\cos(m\beta)\right]\right\}. \tag{42}$$

Here

$$f_M(m,k)\triangleq\begin{cases}1, & m\neq 0,\ k\neq 0;\\ 1/2, & m=0,\ k\neq 0;\\ 1/2, & m\neq 0,\ k=0;\\ 1/2, & m=0,\ k=0.\end{cases} \tag{43}$$

A.2.4 Multipole to Local Expansion Conversion (M2L)

An order l multipole expansion about the point (ρ,α,β), with coefficients O_n^m, can be converted to an order l local expansion about the origin, with coefficients N_j^k, using

$$N_j^k=\sum_{n=0}^{l}\sum_{m=-n}^{n}\frac{\sqrt{(j+n+m-k)!\,(j+n-m+k)!}\;i^{|k-m|}\;Y_{j+n}^{m-k}(\alpha,\beta)\,O_n^m}{\sqrt{(n+m)!\,(n-m)!\,(j+k)!\,(j-k)!}\;(-1)^n\;i^{|k|+|m|}\;\rho^{j+n+1}}. \tag{44}$$

Substituting for the surface spherical harmonics using (23) and for the complex coefficients with (21) and (22) gives the real coefficient multipole to local expansion conversion formulas for $l\geq j\geq k\geq 0$,

$$\bar{N}_j^k=\frac{f_L(k)}{\rho^j(j-k)!}\sum_{n=0}^{l}\frac{(-1)^n}{\rho^{n+1}}\sum_{m=0}^{n}$$
$$P_{j+n}^{|m-k|}(\cos\alpha)\frac{(j+n-|m-k|)!}{(n+m)!}\;i^{|k-m|-k-m}$$
$$\cdot\left\{\bar{O}_n^m\cos[(m-k)\beta]+\tilde{O}_n^m\sin[(m-k)\beta]\right\}$$
$$+P_{j+n}^{m+k}(\cos\alpha)\frac{(j+n-m-k)!}{(n+m)!}$$
$$\cdot\left\{\bar{O}_n^m\cos[(m+k)\beta]+\tilde{O}_n^m\sin[(m+k)\beta]\right\}; \tag{45}$$

$$\tilde{N}_j^k=\frac{1}{\rho^j(j-k)!}\sum_{n=0}^{l}\frac{(-1)^n}{\rho^{n+1}}\sum_{m=0}^{n}$$
$$P_{j+n}^{|m-k|}(\cos\alpha)\;\frac{(j+n-|m-k|)!}{(n+m)!}\;i^{|k-m|-k-m}$$
$$\cdot\left\{-\bar{O}_n^m\sin[(m-k)\beta]+\tilde{O}_n^m\cos[(m-k)\beta]\right\}$$
$$+P_{j+n}^{m+k}(\cos\alpha)\frac{(j+n-m-k)!}{(n+m)!}$$

$$\cdot \left\{ \bar{O}_n^m \sin[(m+k)\beta] - \tilde{O}_n^m \cos[(m+k)\beta] \right\}. \tag{46}$$

Here

$$f_L(k) = \begin{cases} 1, & k \neq 0; \\ 1/2, & k = 0. \end{cases} \tag{47}$$

A.2.5 Local Expansion Shift (L2L)

An order l local expansion about the point (ρ, α, β), with coefficients O_n^m, can be converted to an order l local expansion about the origin, with coefficients N_j^k, using

$$N_j^k = \sum_{n=j}^{l} \sum_{m=-n}^{n} \frac{\sqrt{(n+m)!(n-m)!}\, i^{|m|-|k|-|m-k|}\, Y_{n-j}^{m-k}(\alpha,\beta)\, O_n^m\, \rho^{n-j}}{\sqrt{(n-j+m-k)!(n-j-m+k)!(j+k)!(j-k)!}\,(-1)^{n-j}}. \tag{48}$$

Substituting for the surface spherical harmonics using (23) and for the complex coefficients with (21) and (22) gives the real coefficient local expansion shift formulas for $j \geq k \geq 0$,

$$\begin{aligned} \bar{N}_j^k = {} & \frac{f_L(k)}{(-\rho)^j (j-k)!} \sum_{n=j}^{l} (-\rho)^n \sum_{m=0}^{n} \\ & P_{n-j}^{|m-k|}(\cos\alpha) \frac{i^{m-k-|m-k|}}{(n-j+|m-k|)!(n-m)!} \\ & \cdot \left\{ \bar{O}_n^m \cos[(m-k)\beta] + \tilde{O}_n^m \sin[(m-k)\beta] \right\} \\ & + P_{n-j}^{m+k}(\cos\alpha) \frac{(-1)^k}{(n-j+m+k)!(n-m)!} \\ & \cdot \left\{ \bar{O}_n^m \cos[(m+k)\beta] + \tilde{O}_n^m \sin[(m+k)\beta] \right\}; \end{aligned} \tag{49}$$

$$\begin{aligned} \tilde{N}_j^k = {} & \frac{1}{(-\rho)^j (j-k)!} \sum_{n=j}^{l} (-\rho)^n \sum_{m=0}^{n} \\ & P_{n-j}^{|m-k|}(\cos\alpha) \frac{i^{m-k-|m-k|}}{(n-j+|m-k|)!(n-m)!} \\ & \cdot \left\{ -\bar{O}_n^m \sin[(m-k)\beta] + \tilde{O}_n^m \cos[(m-k)\beta] \right\} \\ & + P_{n-j}^{m+k}(\cos\alpha) \frac{(-1)^k}{(n-j+m+k)!(n-m)!} \\ & \cdot \left\{ \bar{O}_n^m \sin[(m+k)\beta] - \tilde{O}_n^m \cos[(m+k)\beta] \right\}. \end{aligned} \tag{50}$$

The function $f_L(k)$ is given by (47).

References

[1] P. Dewilde and Z.-Q. Ning, *Models for Large Integrated Circuits*, Kluwer Academic Publishers, Boston, Massachusetts, 1990.

[2] L. Greengard, *The Rapid Evaluation of Potential Fields in Particle Systems*, M.I.T. Press, Cambridge, Massachusetts, 1988.

[3] L. Greengard and V. Rohklin, *On the Efficient Implementation of the Fast Multipole Algorithm*, Yale University Research Report YALEU/DCS/RR-602, February 1988.

[4] R. Guerrieri and A. Sangiovanni-Vincentelli, "Three-dimensional capacitance evaluation on a Connection Machine," *IEEE Trans. on Computer-Aided Design,* vol. CAD-7, no. 11, pp. 1125–1133, November 1988.

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

[6] J. L. Hess and A. M. O. Smith, "Calculation of potential flow about arbitrary bodies," *Progress in Aerospace Sciences*, vol. 8, pp. 1–138, 1966.

[7] E. W. Hobson, *The Theory of Spherical and Ellipsoidal Harmonics*, Chelsea, New York, 1955.

[8] J. D. Jackson, *Classical Electrodynamics*, J. Wiley and Sons, New York, 1975.

[9] J. Katzenelson, *Computational Structure of the N-body Problem*, Mass. Inst. of Tech., Artificial Intelligence Laboratory, AI Memo 1042, April 1988.

[10] W. Magnus and F. Oberhettinger, *Special Functions of Mathematical Physics*, Chelsea, New York, 1949, pg. 54.

[11] Z.-Q. Ning and P. M. Dewilde, "SPIDER: capacitance modeling for VLSI interconnections," *IEEE Transactions on Computer-Aided Design*, vol. CAD-7, no. 12, pp. 1221–1228, December 1988.

[12] S. Rao, T. Sarkar and R. Harrington, "The Electrostatic Field of Conducting Bodies in Multiple Dielectric Media," *IEEE Transactions on Microwave Theory and Techniques*, vol. MTT-32, no. 11, pp. 1441–1448, November 1984.

[13] V. Rohklin, "Rapid Solution of Integral Equations of Classical Potential Theory," J. Comput. Phys., vol. 60, no. 2, pp. 187–207, 1985.

[14] A. Ruehli and P. A. Brennen, "Efficient capacitance calculations for three-dimensional multiconductor systems," *IEEE Transactions on Microwave Theory and Techniques*, vol. MTT-21, no. 2, pp. 76–82, February 1973.

[15] Y. Saad and M. H. Schultz, "GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems," *SIAM J. Sci. Stat. Comput.*, vol. 7, no. 3, pp. 856–869, July 1986.

[16] A. H. Zemanian, R. P. Tewarson, C. P. Ju, and J. F. Jen, "Three-Dimensional Capacitance Computations for VLSI/ULSI Interconnections," *IEEE Trans. on Computer-Aided Design*, vol. CAD-8, no. 12, pp. 1319–1326, December 1989.

[17] F. Zhao, *An O(N) Algorithm for Three-Dimensional N-body Simulations*, Master's thesis, Mass. Inst. of Tech., Dept. of Elec. Eng. and Comp. Sci., October 1987.