EFFICIENT CAPACITANCE EXTRACTION OF 3D STRUCTURES USING GENERALIZED PRE-CORRECTED FFT METHODS *

J. R. PHILLIPS AND J. K. WHITE
DEPARTMENT OF ELECTRICAL ENGINEERING AND COMPUTER SCIENCE
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
CAMBRIDGE, MA 02139
PHONE: (617) 253-2543 FAX: (617) 258-7859

1. Introduction. In the design of high performance integrated circuits and integrated circuit packaging, there are many cases where accurate analysis of the electromagnetic characteristics of complicated three-dimensional structures is important for determining final circuit speeds and functionality. The method of moments, a common technique for discretizing the potential-integral equations describing the electromagnetic properties of these structure, generates large, dense matrix problems. Recently developed fast algorithms for solving these dense matrix problems have been based on multipole-accelerated iterative methods [7, 6, 5], in which the fast multipole algorithm [3, 1] is used to rapidly compute the matrix-vector products in a Krylov-subspace based iterative method. In this paper, we describe an FFT-based algorithm which is avoids the high memory overhead associated with the multipole algorithms, and for reasonably homogeneous problems requires $O(n)$ memory and $O(n \log n)$ computation time.

2. Formulation. As an example problem, consider capacitance extraction for a system of conductors embedded in a homogenous dielectric medium. The capacitance matrix of the system is determined by setting the potential of each conductor in turn to unity, then determining the charge density on the other conductors in the system. Assuming a single layer charge density σ, σ must satisfy the first-kind integral equation

$$\psi(x) = \int_{surfaces} \sigma(x') \frac{1}{4\pi\epsilon_0 \|x - x'\|} da', \tag{1}$$

where $\psi(x)$ is the known conductor surface potential, da' is the incremental conductor surface area, x, $x' \in \mathbf{R}^3$, and $\|x\|$ is the usual Euclidean length of x.

A standard approach to numerically solving (1) for σ is to use a piece-wise constant collocation scheme, where σ is represented by a set of panels, each carrying uniform charge density. The result is a dense linear system,

$$Pq = \overline{p} \tag{2}$$

where $P \in \mathbf{R}^{n \times n}$, q is the vector of panel charges, $\overline{p} \in \mathbf{R}^n$ is the vector of known panel potentials, and

$$P_{ij} = \frac{1}{a_j} \int_{panel_j} \frac{1}{4\pi\epsilon_0 \|x_i - x'\|} da', \tag{3}$$

where x_i is the center of the i-th panel and a_j is the area of the j-th panel.

3. Grid-based Approach. If Gaussian elimination is used to solve (2), the number of operations is order n^3. Clearly, this approach becomes computationally intractable if the number of panels exceeds several hundred. Instead, consider solving the linear system (2) using a conjugate-residual style iterative method like GMRES. The dominant costs of such an algorithm are in calculating the n^2 entries of P using (3) before the iterations begin, and performing n^2 operations to compute the dense matrix-vector product on each iteration.

To develop a faster approach to computing the matrix-vector product, after discretizing the problem into n panels, consider subdividing the problem domain into an array of small cubes so that each small cube contains only a few panels. Several sparsification techniques for P are based on the idea of directly computing only those portions of Pq associated with interactions between panels in neighboring cubes. The rest of Pq is then somehow approximated to accelerate the computation [3].

One approach to computing distant interactions is to exploit the fact that evaluation points distant from a cube can be accurately computed by representing the given cube's charge distribution using a small number of weighted point charges[2]. Pq can then be approximated in four steps: (1) project the panel charges onto a uniform grid of point charges, (2) compute the grid potentials due to grid charges, (3) interpolate the grid potentials onto the panels, (4) directly compute nearby interactions. This process is summarized in Figure 1.

There are several possible approaches to computing the grid charge. The implementation described in this paper requires that the potential of the grid charges representing a panel charge exactly match the potential of the

* This work was supported by ARPA contracts N00014-91-J-1698 and N00174-93-C-0035, as well as grants from the Semiconductor Research Corporation (SJ-558), IBM and Digital Equipment Corporation.

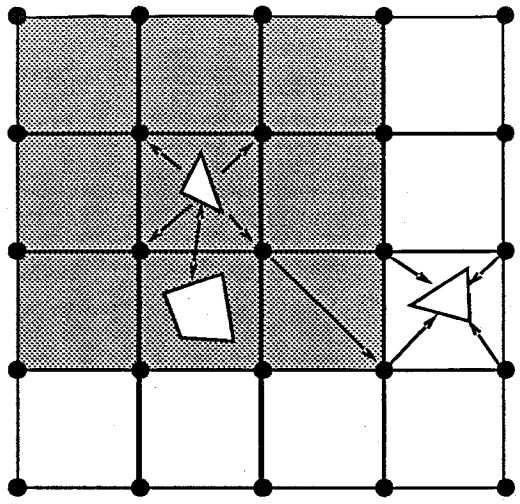

FIG. 1. *2-D Pictorial representation of the four steps of the precorrected-FFT algorithm. Interactions with nearby panels (in the grey area) are computed directly, interactions between distant panels are computed using the grid.*

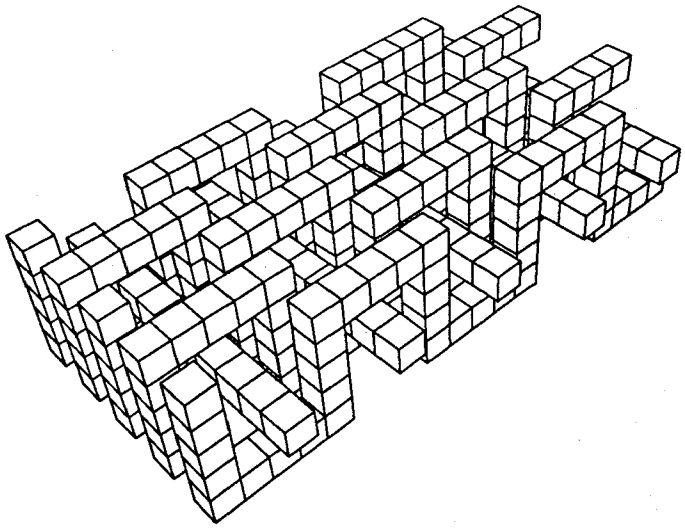

FIG. 2. *A typical capacitance extraction problem*

panel charge at collocation points chosen on the surface of the near-neighbor cubes surrounding the cell enclosing the panel. Additionally, the sum of grid charges is required to be equal to the panel charge, which is equivalent to matching the order zero multipole expansion coefficient for the panel charge distribution. Empirical results indicate that for a $3 \times 3 \times 3$ array of grid charges per unit cell, the grid charge representation is slightly more accurate outside near-neighbor cells than a second-order multipole expansion outside *next*-near-neighbor cells.

When the grid charges have been determined, their potentials at the grid points must be computed. The potential $\psi(\boldsymbol{x})$ at a point $\boldsymbol{x} = (x, y, z)$ is the sum of the potentials from all the grid charges $q(\boldsymbol{x}')$,

$$\psi(\boldsymbol{x}) = \sum_{\boldsymbol{x}'} g(\boldsymbol{x}|\boldsymbol{x}')q(\boldsymbol{x}') \,. \tag{4}$$

The free-space Green function $g(\boldsymbol{x}|\boldsymbol{x}') = g(x - x', y - y', z - z')$ depends only on the relative difference between the points $\boldsymbol{x}$ and $\boldsymbol{x}'$. Therefore, because of the regular grid, the computation of the grid-charge potentials at the grid points is a three-dimensional discrete convolution. This convolution can be rapidly computed by using the Fast Fourier Transform, requiring $O(N \log N)$ operations.

Once the grid potentials have been computed, they can be interpolated to the panels in each cube using the transpose of the operator which computes grid charges from panel charges.

For reasonably homogeneous panel distributions, and assuming a constant number of panels per cell, this algorithm requires $O(N \log N)$ steps and $O(N)$ storage.

4. Pre-correction. In the computation of panel potentials due to grid charges, the portions of Pq associated with neighboring cube interactions have already been computed, though this close interaction has been poorly approximated in the projection/interpolation. Before computing a better approximation, it is necessary to remove the contribution of the inaccurate approximation. It is possible to construct a "precorrected" direct interaction operator, $P^{cor}_{a,b}$, which consists of the direct interaction operator for neighboring cells, with the errors introduced by the grid-charges exactly subtracted out. When used in conjunction with the grid charge representation, $P^{cor}_{a,b}$ results in exact calculation of the interactions between panels which are close. Assuming that the Pq product will be computed many times in the inner loop of an iterative algorithm, $P^{cor}_{a,b}$ will be expensive to initially compute, but will cost no more to subsequently apply than $P_{a,b}$.

5. Modified Green functions for planar geometries. In solving potential integral equations, it is sometimes useful to reduce the size of the problem by formulating a Green function which accounts for the special geometry of a system, thereby removing part of the problem domain from consideration. For an arbitrary Green function $g(\boldsymbol{x}|\boldsymbol{x}')$ the approach described above will fail since the grid-potential calculation cannot be expressed as a discrete convolution. An important special case where the pre-corrected FFT approach can still be applied occurs in a system where the Green function is modified by the presence of planar structure, for example, ground planes or planar dielectric interfaces. To illustrate the problem and its solution, we consider the case of conducting bodies

Example	Speed	Memory	Product	Density	Panels/cell
micromotor	0.71	0.81	0.58	0.87	10.4
cube	1.14	0.31	0.35	0.35	4.3
woven bus	0.82	0.42	0.34	0.78	17.3
bus crossing	0.59	0.26	0.15	0.47	7.3
via	2.26	0.37	0.84	0.33	2.9
DRAM cell	0.88	0.73	0.64	0.77	33.2

TABLE 1

Performance of FFT-based to multipole codes, free-space Green function. "Speed" is ratio of matrix-vector product time of pre-corrected FFT method to fast multipole based method, "memory" the ratio of required storage, "product" the speed-memory ratio product. Density is the fraction of grid cells which contain charge.

over a single ground plane located at $z = 0$ and extending to infinity in the x and y directions. The extension to the cases of multiple ground planes, dielectric interfaces, or combinations thereof is straightforward.

By the method of images, the potential at (x, y, z) due to a single charge at (x', y', z') is

$$\psi(x, y, z) = g(x - x', y - y', z - z') - g(x - x', y - y', z + z') \tag{5}$$

where g is the free-space Green function. The difficulty for the precorrected-FFT method is that the second term depends on $z + z'$, a general difficulty for problems with planar interfaces. The matrix mapping grid charges to grid potentials is the sum of a matrix with block-Toeplitz structure, corresponding to the first term of Eq. 5, and a matrix with block-Hankel structure, corresponding to the second term of Eq. 5. The Toeplitz-like part of the matrix corresponds to the discrete convolution with the free-space Green function, an can be treated directly with the FFT as described above. Because a Hankel matrix is related to a Toeplitz matrix via a permutation matrix[4] which is simple to compute, multiplication by a Hankel matrix may also be done in $O(N \log N)$ time via the FFT. Furthermore, the permutation matrix may be represented in Fourier space so that multiplication of a vector by the sum of a Hankel and Toeplitz matrix can be performed using a single forward and inverse FFT pair. Thus, at each iteration of GMRES, the only additional computation required to incorporate a modified Green function is multiplication in Fourier space by a diagonal matrix and a permutation matrix, which requires negligible additional computation time.

6. Algorithm Performance. Table 1 shows comparisons of the precorrected-FFT algorithm with the multipole-based code FASTCAP[6] on a variety of realistic examples drawn from engineering problems (a typical example, the woven bus, is shown in Fig. 2). The precorrected-FFT method can be as much as 40% faster and can use as little as one fourth the memory of FASTCAP[6], a multipole-based capacitance extraction code. We have also shown a figure-of-merit defined as the product of the memory- and speed- advantages of the precorrected-FFT method. It is important to consider this figure because it is often possible to trade memory for speed by changing the size of the grid used in the precorrected-FFT method (this is true of the "via" structure, for example). In terms of the speed/memory product, the precorrected-FFT method was found to be superior to FASTCAP for all the examples shown, in some cases by more than a factor of six. As is clear from the density of cells containing charge, and the mean number of panels per cell, performance of the precorrected-FFT method is best when the panel distribution is most homogeneous. It appears that for many realistic examples, the panel distribution is sufficiently homogeneous to make the precorrected-FFT method competitive with fast multipole algorithms.

Preliminary results indicate that substitution of the ground-plane Green function for the free-space Green function results in an increase of 50% in initialization time for the capacitance extraction program, an increase of 10-20% in required memory, and virtually no increase in time required to compute a matrix-vector vector product.

REFERENCES

[1] C. R. ANDERSON, *An implementation of the fast multipole method without multipoles*, SIAM J. Sci. Statist. Comput., 13 (1992), pp. 923–947.

[2] L. BERMAN, *Grid-multipole calculations*, Tech. Report RC 19068(83210), IBM Research Report, 1993.

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

[4] R. A. HORN AND C. R. JOHNSON, *Matrix Analysis*, Cambridge University Press, Cambridge, 1985.

[5] V. JANDHYALA, E. MICHIELSSEN, AND R. MITRA, *Multipole-accelerated capacitance computation for 3-d structures in a stratified dielectric medium using a closed form green's function*, tech. report, University of Illinois, 1994.

[6] K. NABORS AND J. WHITE, *Fastcap: A multipole accelerated 3-D capacitance extraction program*, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 10 (1991), pp. 1447–1459.

[7] V. ROHKLIN, *Rapid solution of integral equation of classical potential theory*, J. Comput. Phys., 60 (1985), pp. 187–207.