A Fast Multipole Algorithm for Capacitance Extraction of Complex 3-D Geometries

K. Nabors, J. White

Research Laboratory of Electronics
Dept. of Electrical Engineering and Computer Science
Massachusetts Institute of Technology
Cambridge, MA 02139

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. The method is an acceleration of the standard integral equation approach for multiconductor capacitance extraction. These integral equation methods are slow because they lead to dense matrix problems which are typically solved with some form of Gaussian elimination. This implies the computation grows like n^3, where n is the number of tiles needed to accuracy discretize the conductor surface charges. In this paper we present a preconditioned conjugate-gradient iterative algorithm with a multipole approximation to compute the iterates. This reduces the complexity so that accurate multiconductor capacitance calculations grow as nm where m is the number of conductors.

1 Introduction

In the design of high performance integrated circuits, there are many cases where accurate estimates of the capacitances of complicated three dimensional structures are important for determining final circuit speeds or functionality. Two examples are complicated three-dimensional dynamic memory cells and the three-dimensional chip carriers commonly used in mainframe computers. 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, the technique that is typically used in three dimensions is the integral equation approach[ruehli73,rao84,ning88]. In this approach, the surfaces or edges of all the conductors are broken into small tiles. It is assumed that on each tile i, a charge, q_i, is uniformly or linearly distributed. The potential on each tile is then computed by summing the contributions to the potential from all the tiles using Laplace's equation Green's functions. In this way a matrix of potential coefficients, P, relating the set of n tile potentials and the set of n tile charges is constructed, and must be solved to compute capacitances. Typically, Guassian elimination or Cholesky factorization is used to solve the equation, in which case the number of operations is order n^3. Clearly, this approach becomes computationally intractable if the number of unknowns exceeds several hundred, and this limits the size of the problem that can be analyzed to one with a few conductors.

In this paper we present an algorithm for computing capacitance whose complexity grows as mn, where m is the number of conductors. Our algorithm, which is really the pasting together of three well-known algorithms[rohklin86], 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 symmetric matrix. Then, in Section 3, a preconditioned conjugate-gradient algorithm is described, and it is shown to reduce the complexity of the calculation to order mn^2. In Section 4, it is shown that the conjugate-gradient algorithm only requires the evaluation of a potential field from a charge distribution, and this can be computed in order n time using a multipole algorithm. In Section 5, some preliminary experimental results are given, and we present our conclusions and acknowledgments.

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 i^{th} column of C can be calculated by solving for the total charges on each of the conductors when the i^{th} conductor is at

CH2671-6/89/0000-0134 $1.00 © 1989 IEEE

unit potential, and all the other conductors are at zero potential. Then the charge on conductor j, $\hat{q}_j$, is equal to C_{ij}.

There are a variety of approaches for numerically computing the conductor charges given a set of conductor potentials, and we will focus on integral equation methods[ruehli73,rao84,ning88], as they are efficient when applied to problems with ideal conductors in a uniform dielectric medium. The 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 potential is related to the discretized surface charge through integrals of a Green's functions.

Let the surfaces of a collection of m conductors in free space be discretized into a total of n tiles. The potential at the center of the i^{th} tile would be the sum of the contributions to the potential from the charge distribution on every tile. That is,

$$p_i = \sum_{j=1}^{n} \int_{tile_j} \frac{q_j(r)}{| r - \tilde{r}_i |} da \qquad (1)$$

where $\tilde{r}_i$ is the position of the center of tile i, r is the position on the surface of tile j, p_i is the potential at $\tilde{r}_i$, $q_j(r)$ is the position dependent charge density on the surface of the j^{th} tile, and $| r |$ denotes the Euclidian length of r. Note that the integral in (1) is the free space Green's function multiplied by the charge density, integrated over the surface of the j^{th} tile, and that as the distance between tile i and tile j becomes large compared to the surface area of tile j, the integral reduces to $\frac{q_j}{|\tilde{r}_j - \tilde{r}_i|}$ where q_j is the total charge on tile j.

There are several approaches to simplifying (1), the simplest is the "point-matching" approximation in which it is assumed that the charge is distributed uniformly on the tile surface[rao84]. In that case (1) can be simplified to

$$p_i = \sum_{j=1}^{n} \frac{q_j}{a_j} \int_{tile_j} \frac{1}{| r - \tilde{r}_i |} da \qquad (2)$$

where q_j is the total charge on tile j, and a_j is the surface area of tile j. When applied to the collection of n tiles, a dense linear system results,

$$Pq = p \qquad (3)$$

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

$$P_{ij} = P_{ji} = \frac{1}{2}\left[\frac{1}{a_j}\int_{tile_j} \frac{1}{| r - \tilde{r}_i |} da + \frac{1}{a_i}\int_{tile_i} \frac{1}{| r - \tilde{r}_j |} da\right] . (4)$$

Note that q and p are the vectors of *tile* charges and potentials rather than the *conductor* charge and potential vectors, $\hat{q}$ and $\hat{p}$ mentioned above. In (4), the potential coefficients, P_{ij}, have been "symmetrized" by averaging for several reasons: the physical system is symmetric, the symmetrized equations have been shown to produce more accurate results for a given discretization, and a symmetric matrix problem is more easily solved. The dense linear system of (3) can be solved, typically by some form of symmetric Gaussian elimination, to compute tile charges from a given set of tile potentials. To compute the j^{th} column of the capacitance matrix, (3) must be solved for q, given a p vector whose entries p_i are set equal to one if tile i is on the j^{th} conductor, and zero otherwise. Then the ij^{th} term of the capacitance matrix is computed by summing all the charges on the j^{th} conductor, i.e. $C_{ij} = \sum_{k \in Conductor_j} q_k$.

3 Using Preconditioned Conjugate-Gradient

In order to solve for a complete $m \times m$ capacitance matrix, the $n \times n$ symmetric matrix of potential coefficients, P, must be factored once, usually into $P = LL^T$, 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 tiles into which the conductor surfaces are cut, m is necessarily much less than n. Therefore, the n^3 factorization dominates for large problems.

This suggests that iterative methods might be more efficient than direct factorization for solving the m charge distribution problems. In particular, as the matrix is symmetric and positive definite, the conjugate-gradient (CG) algorithm is a natural choice[golub83]. Unfortunately, the CG algorithm can converge slowly when applied to the matrix of potential coefficients, particularly when the problem contains widely separated pairs of very closely spaced tiles. To accelerate the convergence of CG, an attempt is made to factor most of the part of the problem associated with the closely spaced tiles directly. To accomplish this, the smallest cube containing the entire problem is uniformly divided into a large number of cubes, typically into as close to $\frac{n}{10}$ cubes as possible. The piece of the potential coefficient matrix associated with the tile interactions inside a cube is then factored directly and used as a preconditioner to accelerate the CG algorithm. If the p and q vector in (3) are reordered so that tiles contained in a given cube are ordered contiguously, the potential coefficients representing the interaction between tiles in a given cube will be blocks on the diagonal of P. That is, $P = P_{intracube} + P_{intercube}$ where $P_{intracube}$ is a block diagonal matrix.

The CG capacitance extraction algorithm with the $P_{intracube}$ preconditioner (PCG) is as follows:

Algorithm 1: Preconditioned CG capacitance extraction algorithm

```
Setup Phase.
  Divide all the conductors into a total of $n$ tiles.
  Divide the tiles into cubes, and reorder to
    make $P_{intracube}$ block diagonal.
Compute the Potential Coefficient Matrix.
  for $i = 1$ to $i = n$
    for $j = 1$ to $j = n$
      Compute $P_{ij}$ from (4).
  Factor $P_{intracube}$.
Loop Through all the Conductors.
  for $k = 1$ to $m$
    if tile $i$ is on conductor $k$, set $p_i = 1$.
    else $p_i = 0$.
    Use PCG to solve $Pq = p$.
    for $l = 1$ to $m$ $C_{kl} = \sum_{k \in conductor_l} q_k$.

Preconditioned CG (PCG).
The Setup.
  $r = p$, $q = 0$.
Conjugate-Gradient Loop.
Repeat
  Solve $P_{intracube} z = r$.
  if the first iteration $\beta = 0$.
  else $\beta = z^T r / (z^T r)_{prev}$.
  $x = z + \beta x$.
  $y = Px$.
  $\alpha = \frac{z^T r}{x^T A x}$.
  $q = q + \alpha x$.
  $r = r - \alpha y$.
Until Converged
```

4 Acceleration with a Multipole Algorithm

As can be seen from examining the computation in Algorithm 1, m problems must be solved iteratively, and the major cost is computing the matrix P, and in each iteration forming the product Px, both of which are order n^2. This implies that computing the capacitance matrix with Algorithm 1 is order mn^2, and may not be much more efficient than direct factorization if the ratio of tiles to conductors is low.

An approach for reducing the cost of forming P and computing Px in the CG algorithm can be derived by recalling that if x is thought of as a charge distribution, Px is the potential due to that charge distribution. To see how this helps simplify the computation Px, consider two widely separated cubes, each with k tiles. Computing the contributions to the potentials at the center of each of the tiles in the first cube due to the k tile charges in the second cube from (4) requires k^2 calculations. If all the charges in the second cube are positive, then the k potential contributions to the first cube can be computed approximately in k operations. This is done by assuming the charges in the second cube contribute to potential in the first cube like a point charge equal to the sum of the charges in the second cube located at a "center of mass". Note that the accuracy of the approximation improves as the separation between cubes increases.

There are a collection of algorithms based on the above idea, often referred to as multipole algorithms[rohklin86,katzenelson88,zhao87]. The details of the multipole algorithm we used are well described in[greengard87], and only a very basic outline will be given here. In general, the potential, ψ, due to a cube of point charges at a location outside the radius of the cube is given by the *multipole* expansion,

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

where r, θ and ϕ are the spherical coordinates of the evaluation location, $Y_n^m(\theta,\phi)$ is the spherical harmonic, and M_n^m is the multipole coefficient, which can be computed from the charge in the cube from

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

where ρ_i, α_i, and β_i are the spherical coordinates of the i^{th} charge. If the evaluation location is well outside the cube, then the potential can be accurately computed using just a few terms of the multipole expansion.

Consider a collection of cubes containing charges and one cube, well separated from the others, containing several locations at which the potential must be evaluated. It is possible to combine all the multipole expansions for the cubes containing charges into a single local expansion from which the potential at the evaluation points in the cube can be computed quickly. The local expansion is given by

$$\psi(r,\theta,\phi) = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} L_n^m Y_n^m(\theta,\phi) r^n \tag{7}$$

where r, θ and ϕ are the spherical coordinates of the evaluation location, and L_n^m are the local expansion coefficients, which are computed from the combination of multipole expansions for the cubes containing charges. Good accuracy can be achieved with a few terms of the local expansion.

Truncated multipole and local expansions can be used to compute n potentials at n evaluation points in order n operations, provided the charges and evaluation points are reasonably separated. To ensure adequate separation and avoid excess calculation, careful hierarchical shifting and combining of both the multipole and local expansions is necessary, as is well described in [greengard87].

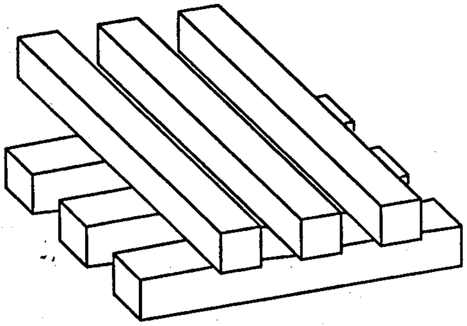

Figure 1: Bus Structure Example with Six Conductors

	2 Cond.	4 Cond.	6 Cond.
tiles	216	720	1512
cubes	64	64	64
direct time	67	2520	20160
PCG time	65	653	2613
PCG iters	7	10	12
MPCG iters	8	11	14
MPCG rel. err.	0.002	0.001	0.002

Table 1: Comparison of Extraction Methods

With the computation organized in this manner, the multipole algorithm can be used to compute most of Px in Algorithm 1, except the part due to interactions between tiles in a given cube, and the tiles of each cube's nearest neighbors. This implies that in Algorithm 1, if the multipole algorithm is used to compute Px, most of P need not be formed explicitly. Note also that the part that must be computed explicitly includes $P_{intracube}$, therefore the multipole accelerated PCG algorithm can still use $P_{intracube}$ as a preconditioner. Finally, note that using the multipole algorithm to compute Px implies that both n^2 steps of Algorithm 1, forming all of P and computing Px, can be removed.

5 Results and Conclusions

The multipole accelerated PCG algorithm was implemented and tested on a simple bus structure (Figure 1), with 2, 4, and 6 conductors. In Table 1 we report the total number of tiles resulting from the conductor surface discretization, the number of cubes into which space was divided, the time to compute capacitance using direct factorization and PCG, the number of iterations to achieve convergence with PCG and multipole accelerated PCG (MPCG), and the relative error introduced by the multipole approximation.

Much additional work is under way to improve the efficiency of our MPCG-based capacitance extraction program, and CPU time comparisons for an efficient implementation will be presented at the conference. Future research includes extending the approach to piecewise-constant dielectrics and problems with ground planes.

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 that led to the approach presented here, as well as their help along the way. In addition we would like to acknowledge the helpful discussions with Jacob Katzenelson, and finally we thank the many members of the MIT Custom Integrated Circuits group for their help and encouragement.

References

[golub83] G. Golub and C. Van Loan, *Matrix Computations*, Johns Hopkins University Press, Baltimore, Maryland, 1983.

[greengard87] L. Greengard, V. Rohklin, "A Fast Algorithm for Particle Simulations," J. Comp. Phys., Vol 73, pp. 325–348, 1987.

[katzenelson88] J. Katzenelson, *Computational Structure of the N-body Problem*, Mass. Inst. of Tech., Artificial Intelligence Laboratory, AI Memo 1042, April 1988.

[ning88] Z.-Q. Ning and P. M. Dewilde, "SPIDER: Capacitance Modeling for VLSI Interconnections," *IEEE Transactions on Computer-Aided Design*, vol. CAD-7, No. 12, December 1988.

[rao84] S. Rao, T. Sarkar, R. Harrington, "The Electrostatic Field of Conducting Bodies in Multiple Dielectric Media," *IEEE Transactions on Microwave Theory and Techniques*, vol. MTT-32, No. 11, November 1984.

[rohklin86] V. Rohklin, "Rapid Solution of Integral Equation of Classical Potential Theory," J. Comput. Phys., Vol. 60, pp. 187–207, 1985.

[ruehli73] 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.

[zhao87] 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.