Multilevel Integral Equation Methods for the Extraction of Substrate Coupling Parameters in Mixed-Signal IC's

Mike Chou Jacob White
Department of EECS, Massachusetts Institute of Technology
Cambridge, MA. 02139

Abstract—

The extraction of substrate coupling resistances can be formulated as a first-kind integral equation, which requires only discretization of the two-dimensional contacts. However, the result is a dense matrix problem which is too expensive to store or to factor directly. Instead, we present a novel, multigrid iterative method which converges more rapidly than previously applied Krylov-subspace methods. At each level in the multigrid hierarchy, we avoid dense matrix-vector multiplication by using moment-matching approximations and a sparsification algorithm based on eigendecomposition. Results on realistic examples demonstrate that the combined approach is up to an order of magnitude faster than a Krylov-subspace method with sparsification, and orders of magnitude faster than not using sparsification at all.

I. Introduction

The design of single chip mixed-signal systems is now an active area of research, driven by the relentless quest for high-level integration and cost reduction. A major challenge for mixed-signal design tools is the accurate modeling of the parasitic noise coupling through the common substrate between the high-speed digital and high-precision analog components [22], [11], [15]. Fast switching logic components inject current into the substrate, causing voltage fluctuations which can affect the operation of sensitive analog circuitry through the body effect. Since the bulk substrate behaves resistively up to a frequency of a few gigahertz [6], [23], it is sufficient to solve Laplace's equation inside the substrate with proper boundary and interface conditions. Examples of this approach [11], [22], [13], [20], [25] includes Finite Element (FEM) and Finite Difference (FD) methods. Although the resulting linear systems are *sparse*, such methods are impractical for complex layouts because the number of unknowns resulting from three-dimensional volume-meshing of the entire substrate is too large.

Integral equation based techniques such as the Boundary Element Method (BEM) have been applied with some success to the modeling of substrate coupling [19], [5], [24]. Since only the two-dimensional substrate contacts are discretized, BEM methods dramatically reduce the size of the linear system to be solved. However, they produce *dense* matrices, which are too expensive to store or factor directly. To address this difficulty, heuristic partitioning schemes [7], [19] were proposed to sparsify the matrix *inverse* by setting direct admittances to contacts outside a user-defined region to zero. This approach requires too much user intervention and, more importantly, results in errors that are difficult to control and quantify.

Iterative schemes, combined with a sparsification algorithm to compute dense matrix-vector products efficiently, can be effective for solving large BEM systems. In [24], GMRES [17], a Krylov-subspace based iterative method, was combined with a fast multipole [8] algorithm for substrate resistance extraction. However, accuracy is compromised since the multipole algorithm cannot handle substrate edge effects. A more serious difficulty is that Krylov-subspace iterative methods converge slowly when applied to large BEM systems, which tend to be ill-conditioned [18], [21], [12]. Hundreds of iterations may be required per solution for large problems.

Multigrid methods, or more generally, multilevel methods, are known to be the most efficient iterative techniques in the solution of elliptic partial differential equations (PDE's) [1], [9], [2] due to their fast convergence. However, multilevel methods are not well-developed for first-kind integral equations [12] defined over complicated geometries, as is our case here. In this paper, we address this void by developing a multigrid iterative solver, and then integrating it with sparsification algorithms specially tuned to accurately account for substrate edge effects. Section II summarizes the BEM formulation for substrate coupling resistance extraction. Our new multilevel algorithm is presented in Section III. Computational results are given in Section IV, where comparisons to Krylov-subpace iterative methods are also made. Finally we give our conclusions and acknowledgements in Section V.

II. Background and Previous Work

In the electrostatic approximation [6], [24], the substrate is modeled as a stratified medium composed of several homogeneous layers characterized by their conductivities shown in Figure 1. Three contacts are shown in gray. For this work, the substrate backplane is assumed to be grounded electrically.

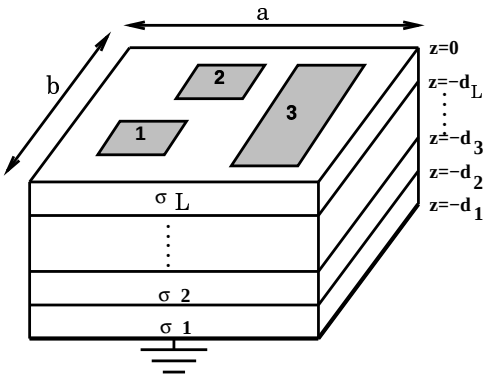

Fig. 1. 3D substrate profile.

Since the problem is linear, an integral equation defined over S, the collection of two-dimensional substrate contacts, can be written

$$\phi(\mathbf{r}) = \int_S \rho_S(\mathbf{r}')G(\mathbf{r};\mathbf{r}')da', \qquad \mathbf{r} \in S, \tag{1}$$

where $\mathbf{r}, \mathbf{r}'$ are points on S, ρ_S is the current density on S, and $G(\mathbf{r};\mathbf{r}')$ is the *Green's function*, which must also satisfy the appropriate boundary and interface conditions. This is a *first-kind* integral equation [12] which forms the basis for the boundary-element method (BEM) used in [19], [7], [24].

To numerically solve (1), the domain S discretized into N disjoint, rectangular *panels* $\{p_i\}$ such that $S = \bigcup_{i=1}^{N} p_i$. An example of panel discretization for a three-contact layout is given in Figure 2. In the Galerkin scheme [10], the current density on each panel p_i is assumed to be uniform. Then N linear equations are constructed by evaluating the *average* potential $\phi(\mathbf{r})$ over each panel p_i. The result is a discretized version of (1)

$$v = Pq, \tag{2}$$

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

DAC 98, June 15-19, 1998, San Francisco, CA USA
ISBN 1-58113-049-x/98/06…$5.00

where q and v are length-N vectors with q_i denoting the total current on panel i and v_j denoting the average potential on panel j. P is an $N \times N$ matrix given by

$$P_{ij} = \frac{1}{a_i a_j} \int_{p_i} \int_{p_j} G(\mathbf{r}; \mathbf{r}') da da' \tag{3}$$

where a_i and a_j are the areas of panels i and j respectively. P is often called the *coefficient-of-potential* matrix. We note here that P is *dense* since current injected into any panel i produces a non-zero potential at every other panel j.

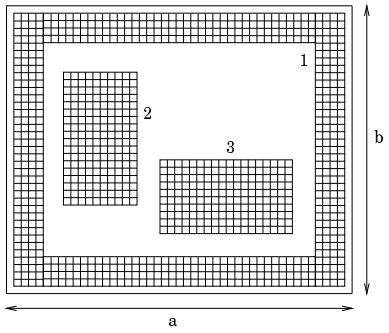

Fig. 2. Example of contact discretization.

For the extraction of substrate coupling resistances, the voltages at the contacts, each corresponding to a circuit node, are specified. We are required to solve for the detailed current distribution, which can be summed over each contact to yield the current flow into each node. This is analogous to the capacitance extraction problem [14]. The linear system (2) is to be solved for q given v.

The Green's function $G(\mathbf{r}; \mathbf{r}')$ for the bounded substrate with grounded backplane was shown in [6] to be a double infinite series of cosines in x and y. By truncating $G(\mathbf{r}; \mathbf{r}')$ to a finite $M \times M$ series and substituting it into (3), it was also shown in [6] that each entry P_{ij} can be constructed from linear combinations of appropriate terms from a two-dimensional $M \times M$ array $\{F_{lm}\}$, which is computed once and for all with a Type-1 Discrete Cosine Transform (DCT). Although this allows individual entries of P to be computed, direct *solution* of (2) still requires $\mathcal{O}(N^3)$ CPU time and $\mathcal{O}(N^2)$ memory since P is dense. This limits the size of the problem to a few hundred panels. In the next section, we develop an efficient *iterative* solver based on multiresolution analysis.

III. Matrix-Free, Multilevel Solution of First-Kind Integral Equations

The efficiency of multigrid iterative methods is a direct result of the fact that convergence rate is independent of discretization, and hence problem size. This is to be contrasted with Krylov-subspace iterative methods, whose convergence rates deteriorate with increasing mesh refinement, or equivalently, worse matrix conditioning. In this section, we develop a multigrid method for the discretized first-kind integral equation (2), as well as sparsification techniques necessary to avoid dense matrix-vector multiplication at each level in the multilevel algorithm.

To best present the method, we first describe the simpler case of a uniformly discretized contact that covers the entire substrate. We then describe the modifications needed for many irregularly shaped contacts.

A. Basic Multigrid Algorithm for Uniform Grids

In this section, we assume that the integral equation is defined over the *entire* substrate $\Omega = [0, a] \times [0, b]$

$$\phi(\mathbf{r}) = \int_{\Omega} \rho_S(\mathbf{r}') G(\mathbf{r}; \mathbf{r}') da', \qquad \mathbf{r} \in \Omega, \tag{4}$$

and that Ω is discretized into a uniform array of $M \times M$ panels. We assume further that M is a power of two, *i.e.* $M = 2^l$ for integer l. We refer to this discrete BEM system as a level l, or *fine-grid*, representation of (4) $P_{\{l\}} \cdot q_{\{l\}} = v_{\{l\}}$. The number of panel unknowns, and hence the size of the linear system, is then $N_l = M^2$. Suppose we also discretize (4) using a coarser, uniform $(M/2) \times (M/2)$ array of panels, yielding a discrete linear system of size $N_{l-1} = N_l/4$. This results in a level $(l-1)$, or *coarse-grid*, representation $P_{\{l-1\}} \cdot q_{\{l-1\}} = v_{\{l-1\}}$. See Figure 3 for the two discretizations.

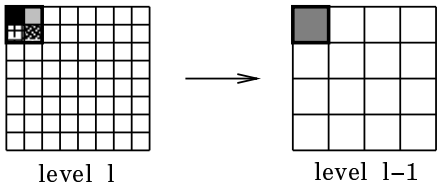

Fig. 3. Two-level Representation and Restriction for Uniform Grid Problem.

Algorithm 1 (Two-Grid Method (TGM))

Set $k \equiv 1, q_{\{l\}}^{(1)} \equiv 0$.
Repeat {
  Fine-Grid Smoothing:
    Solve $D_{\{l\}} \cdot \Delta q_{\{l\}}^* = -P_{\{l\}} \cdot q_{\{l\}}^{(k)} + v_{\{l\}}$
      for $\Delta q_{\{l\}}^*$.
    Compute intermediate guess $q_{\{l\}}^* = q_{\{l\}}^{(k)} + \Delta q_{\{l\}}^*$.
  Compute residual $u_{\{l\}} = P_{\{l\}} \cdot q_{\{l\}}^* - v_{\{l\}}$.
  Project to coarse grid $u_{\{l-1\}} = r u_{\{l\}}$.
  Coarse-Grid Correction:
    Solve for $\Delta q_{\{l-1\}}$ in $P_{\{l-1\}} \cdot \Delta q_{\{l-1\}} = u_{\{l-1\}}$.
    Project to fine grid $\Delta q_{\{l\}} = p(\Delta q_{\{l-1\}})$.
    Update intermediate guess $q_{\{l\}}^{(k+1)} = q_{\{l\}}^* - \Delta q_{\{l\}}$.
  Set $k = k + 1$.
} **Until** residual norm $\|u_{\{l\}}\| < \varepsilon$.

Solving the fine-grid problem by direct matrix factorization is impractical for large N_l since $P_{\{l\}}$ is dense. However, it may be possible to factor the smaller matrix $P_{\{l-1\}}$ corresponding to the coarse-grid problem, since $N_{l-1} = N_l/4$. This motivates our development of a *two-grid method* (TGM), in which the problem is solved *iteratively* at level l with the help of direct solution at level $(l-1)$. The two principal algorithmic components, analogous to TGM for PDE's [1], [9], [2], are the *smoothing* operator and the *intergrid transfer*, or restriction-prolongation, operators. In our TGM iteration for solving $P_{\{l\}}\, q_{\{l\}} = v_{\{l\}}$, the *error* in the k-th iterate, $q_{\{l\}}^{(k)}$, is *smoothed* by carefully solving a series of local problems. This first stage is typically called "fine-grid smoothing", and results in an intermediate guess $q_{\{l\}}^*$. Next, we compute the *residual* $u_{\{l\}} = P_{\{l\}} \cdot q_{\{l\}}^* - v_{\{l\}}$ and project it onto the coarse grid via $u_{\{l-1\}} = r u_{\{l\}}$, where r is a *restriction* operator. Then we solve explicitly the coarse-grid problem $P_{\{l-1\}} \cdot (\Delta q_{\{l-1\}}) = u_{\{l-1\}}$ for $\Delta q_{\{l-1\}}$, and project the result onto the fine grid via $\Delta q_{\{l\}} = p(\Delta q_{\{l-1\}})$, where p is a *prolongation* operator. Finally, the intermediate guess on the fine grid is updated to yield the $(k+1)$-st iterate $q_{\{l\}}^{(k+1)} = q_{\{l\}}^* - \Delta q_{\{l\}}$. This second stage is termed "coarse-grid correction" and is responsible for "long-range" interactions. The fine-grid smoothing/coarse-grid correction cycle is repeated until the norm of the residual $u_{\{l\}}$ is below some tolerance. TGM is summarized in Algorithm 1, where the matrix D is described below.

To derive the smoothing operator, we first make the important observation that the BEM matrix P is derived from a Green's function $G(\mathbf{r}; \mathbf{r}')$ that is sharply peaked as $\mathbf{r} \to \mathbf{r}'$, but is *smooth* otherwise,

i.e. $|\mathbf{r}-\mathbf{r}'| > d$ for some distance d. We seek an *operator splitting* at level l

$$P_{\{l\}} = D_{\{l\}} + S_{\{l\}}, \tag{5}$$

such that $D_{\{l\}}$ captures the short-range, sharply-peaked portion of $P_{\{l\}}$, and $S_{\{l\}}$ captures the long-range, smooth portion of $P_{\{l\}}$. See Figure 4 for a rough depiction.

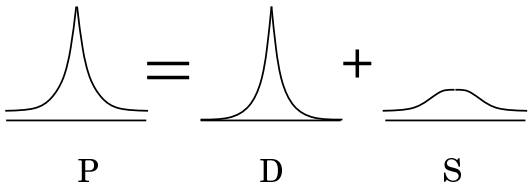

Fig. 4. Operator Splitting $P = D + S$.

Given (5), we define the smoothing operator as the result of solving

$$D_{\{l\}} \cdot q^*_{\{l\}} = -S_{\{l\}} \cdot q^{(k)}_{\{l\}} + v_{\{l\}} \tag{6}$$

for the vector $q^*_{\{l\}}$. Equation (6) defines a *fixed-point* iteration [21], since the condition $q^{(k)}_{\{l\}} = q_{\{l\}}$, where $q_{\{l\}}$ is the exact solution of the fine-grid problem, would lead to $q^*_{\{l\}} = q_{\{l\}}$. Since it is necessary that the above smoothing step be done cheaply, we require that $D_{\{l\}}$ be easy to invert, or that $D^{-1}_{\{l\}}$ has a *sparse* representation, since

$$q^*_{\{l\}} = D^{-1}_{\{l\}} \cdot \left(-S_{\{l\}} \cdot q^{(k)}_{\{l\}} + v_{\{l\}} \right). \tag{7}$$

Here, we construct directly a sparse matrix $D^{-1}_{\{l\}}$ based on the *preconditioner* developed in Fastcap [14], a BEM-based capacitance extraction program. Our particular implementation is outlined as follows. For each panel i, a local coefficient-of-potential matrix P_{loc} involving only panel i and its immediate neighbors is constructed. For the uniform grid problem, the size of P_{loc} is at most 9×9. This small matrix is easily inverted to yield P^{-1}_{loc}, whose elements from the *row* corresponding to panel i are then extracted and *stamped* into corresponding locations in the i-th *row* of $D^{-1}_{\{l\}}$. Hence, the matrix $D^{-1}_{\{l\}}$ contains at most 9 non-zero entries per row and is *sparse*. We recall that the panel-to-panel interaction coefficient can be computed inexpensively from a $M \times M$ DCT array described in Section II.

Because we do not construct $D_{\{l\}}$ directly, it may seem at first glance that the matrix $S_{\{l\}} \equiv P_{\{l\}} - D_{\{l\}}$ is difficult to obtain. But by subtracting $(D_{\{l\}} \cdot q^{(k)}_{\{l\}})$ from (6) and then multiplying through by $D^{-1}_{\{l\}}$, the result

$$\begin{aligned} \Delta q^*_{\{l\}} &= D^{-1}_{\{l\}} \cdot \left(-P_{\{l\}} \cdot q^{(k)}_{\{l\}} + v_{\{l\}} \right) \\ q^*_{\{l\}} &= q^{(k)}_{\{l\}} + \Delta q^*_{\{l\}}. \end{aligned} \tag{8}$$

can be used to compute $q^*_{\{l\}}$. This requires only operators $D^{-1}_{\{l\}}$ and $P_{\{l\}}$ which are readily available.

In addition to the smoothing operator, we require transfer operators r and p between the two grids. They are trivial in the case of uniform grids, where a coarse-grid panel, called a *parent*, is composed of four fine-grid panels, called *kids*. See Figure 3. Recall that in the Galerkin formulation, q_i is the net current on panel i, and v_j the average potential on panel j. For the restriction operator r mapping from level l to level $(l-1)$, the net current on a parent is simply the *sum* of the currents on the four kids, and the average potential on a parent is the average of the four kid potentials. The prolongation operator p mapping from level $(l-1)$ to level l is defined as the *adjoint*, or transpose, of the restriction r [2].

The *multigrid method* (MGM) is the generalization of the two-grid method to an arbitrary number of levels, and this is done exactly as in the standard multigrid literature. Instead of solving the coarse-grid problem explicitly at level $(l-1)$, which may still be too expensive, we apply a similar smoothing-correction cycle at level $(l-1)$. In the same manner, the correction cycle at level $(l-1)$ becomes a smoothing-correction cycle at level $(l-2)$, and so on. The integral equation (4) is now discretized at all levels $\{l_{min}, \ldots, l_{max}\}$. Only at the coarsest level $(l = l_{min})$ is the system $P_{\{l\}}\ q_{\{l\}} = v_{\{l\}}$ solved explicitly. Hence, each multigrid iteration is best described as a *recursive* function call.

B. Hierarchical Basis Functions for Complicated Domains

For integral equations defined on an arbitrarily shaped region S, we recall that the linear system (2) results from a Galerkin discretization based on *constant strength* panels. Figure 2 shows an example of discretization using square panels of equal size. We define here the *characteristic function*, $\mathcal{X}_i(\mathbf{r})$,associated with each panel p_i

$$\mathcal{X}_i(\mathbf{r}) = \begin{cases} 1/a_i & \text{if } \mathbf{r} \in p_i \\ 0 & \text{otherwise} \end{cases} \tag{9}$$

where a_i is the area of p_i. The Galerkin coefficients P_{ij} given in (3) is equivalent to the definition

$$P_{ij} = \int_S \int_S G(\mathbf{r};\mathbf{r}') \cdot \mathcal{X}_i(\mathbf{r}) \cdot \mathcal{X}_j(\mathbf{r}') da da' \tag{10}$$

where the integrations are now over the entire surface S.

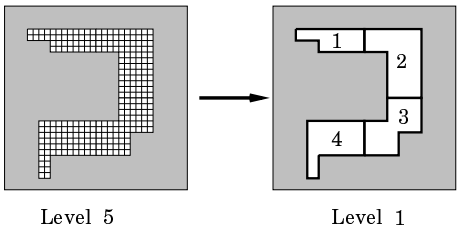

Fig. 5. Hierarchical basis functions.

The explicit use of characteristic functions allows us to construct coarser-level representations of the integral equation defined over complicated geometries. For simplicity, let us assume that at the finest level l_{max}, each panel coincides with a *cell* on the regular $M \times M$ grid, where $M = 2^{l_{max}}$. Of course, not all of the M^2 cells are occupied by panels. A panel at a *coarser* level $l = \{(l_{max} - 1), \ldots, 0\}$ is *defined* in the following manner. At each level l, the substrate is represented by a regular $2^l \times 2^l$ array of level-l cells. For each non-empty level-l cell, a level-l panel is defined as the *union* of all finest-level panels within that cell. The k-th panel at level l is denoted by $p^{\{l\}}_k$, and is associated with a characteristic function $\mathcal{X}^{\{l\}}_k$ defined similarly as in (9). Given the set of hierarchical basis functions $\{\mathcal{X}^{\{l\}}_k\}$, we can now easily define discrete representations of the integral equation $P^{\{l\}} \cdot q^{\{l\}} = v^{\{l\}}$ at each level l, where the Galerkin matrix elements $P^{\{l\}}_{ij}$ are computed as

$$P^{\{l\}}_{ij} = \int_S \int_S G(\mathbf{r};\mathbf{r}') \cdot \mathcal{X}^{\{l\}}_i(\mathbf{r}) \cdot \mathcal{X}^{\{l\}}_j(\mathbf{r}') da da' \tag{11}$$

Figure 5 shows a finest-grid discretization at level 5 being mapped into a coarse representation with only four panels at level 1. An efficient, hierarchical algorithm for the actual computation of the integrals in (11) is described in the next section.

C. Sparsification via Eigendecomposition and Moment-Matching

In this section, we first describe how to compute the matrix-vector product $P^{\{l_{max}\}} \cdot q^{\{l_{max}\}}$ efficiently at the finest level via

an *eigendecomposition* technique. Then we propose a moment-matching algorithm which is used in conjuction with eigendecomposition to compute $P^{\{l\}} \cdot q^{\{l\}}$ efficiently at at all coarser levels $l = \{(l_{max} - 1), \ldots, 0\}$.

We first define a *surface eigenfunction* $\varphi(\mathbf{r})$ as an eigenfunction of the integral operator defined over the entire substrate surface $\Omega = [0, a] \times [0, b]$

$$\lambda \cdot \varphi(\mathbf{r}) = \int_{\Omega} \varphi(\mathbf{r}') G(\mathbf{r}; \mathbf{r}') da', \qquad \mathbf{r} \in \Omega. \tag{12}$$

It has been shown [4] that the surface eigenfunctions are

$$\varphi_{ij}(x, y) = \cos\left(\frac{i\pi x}{a}\right) \cos\left(\frac{j\pi y}{b}\right) \ , \tag{13}$$

where the *eigenvalues* λ_{ij} are given in [7], [6]. Although the eigenfunctions in (13) also appear in [7], [6], they were used only to construct the panel-to-panel interaction coefficients P_{ij}. The dense matrix-vector multiplication $P \cdot q$ still requires $\mathcal{O}(N_p^2)$ operations in [7], where N_p is the number of panels. In contrast, we use the eigenfunctions to *expand* the global current density $J(x, y)$ and show that the $P \cdot q$ product can be computed in $\mathcal{O}(2 \cdot M^2 \cdot \log_2(M))$ operations using the DCT. If we assume that all panels are minimum sized cells on the $M \times M$ substrate grid, and that 10% of the cells are occupied by actual panels (*i.e.* $N_p = (0.1)M^2$), then the cost of computing the dense $P \cdot q$ product directly is $\mathcal{O}(0.01 \times M^4)$. At $M = 128$, eigendecomposition is already an order of magnitude faster than direct multiplication.

We define a prototype characteristic function centered about the origin in one dimension

$$\Theta_a(x) = \begin{cases} M/a & \text{if } |x| \leq a/2M \\ 0 & \text{otherwise} \end{cases} \ , \tag{14}$$

It is then clear that the panel characteristic functions $\mathcal{X}_i(x, y)$ can be obtained by combining and shifting $\Theta_a(x)$ and $\Theta_b(y)$. The global current density resulting from a Galerkin panel discretization is then

$$J(x, y) = \sum_{i=0}^{M-1} \sum_{j=0}^{M-1} q_{ij} \cdot \Theta_a\left(x - \frac{(i + 1/2)\ a}{M}\right) \cdot \Theta_b\left(y - \frac{(j + 1/2)\ b}{M}\right). \tag{15}$$

If we can expand $J(x, y)$ in (15) in terms of the eigenfunctions

$$J(x, y) = \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} a_{ij}\ \varphi_{ij}(x, y) \ , \tag{16}$$

i.e. if the coefficients $\{a_{ij}\}$ can be quickly computed, then (12) immediately leads to

$$\Phi(x, y) = \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} \lambda_{ij}\ a_{ij}\ \varphi_{ij}(x, y). \tag{17}$$

Given the potential everywhere (17), the *average* panel potential $\overline{\Phi}_{pq}$ at the (p, q) position in the cell array is

$$\overline{\Phi}_{pq} = \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} C_{ij} \lambda_{ij} a_{ij} \cos\left(\frac{(p + 1/2)\pi i}{M}\right) \cdot \cos\left(\frac{(q + 1/2)\pi j}{N}\right) \ , \tag{18}$$

where C_{ij} are normalization coefficients. It can be shown [4] that if (16) and (18) are truncated to a finite series, the $M \times M$ coefficients $\{a_{ij}\}$ can be computed via a forward Type-2 DCT, and then the $M \times M$ array $\{\overline{\Phi}_{pq}\}$ via an inverse Type-2 DCT.

To calculate a single Galerkin coefficient $P_{ij}^{\{l\}}$ between two *coarse-grid* panels defined by (11), it is sufficient to perform a double summation of the panel-to-panel coefficients $P_{ij}^{\{l_{max}\}}$ at the finest level. However, this leads to an $\mathcal{O}(N_p^2)$ algorithm. Instead, we make the observation that when two coarse panels $p_i^{\{l\}}$ and $p_j^{\{l\}}$ are "well-separated", their interaction coefficient $P_{ij}^{\{l\}}$ can be computed approximately by leaving out much of the detail in the characteristic functions $\mathcal{X}_i^{\{l\}}$ and $\mathcal{X}_j^{\{l\}}$. Similar ideas have been used extensively in multipole-accelerated algorithms[16], [14], [24]. For the multipole approximation used in [24], it was necessary to assume a substrate Green's function which has translational invariance and which can be fitted to a sum of polynomials in $(1/r)$, where $r = |\mathbf{r} - \mathbf{r}'|$. In contrast, we develop here a fast moment-matching method which can be used in combination with the DCT to accelerate the coarse-grid computations *and* account properly for all the substrate boundary effects.

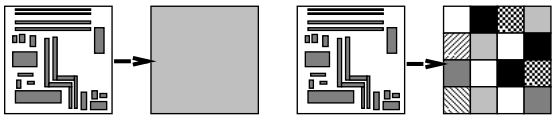

Fig. 6. Moment-matched representation of coarse panel.

If we are interested in the potential at a panel $p_j^{\{l\}}$ far from $p_i^{\{l\}}$, it may suffice to approximate $\mathcal{X}_i^{\{l\}}$ by distributing the current uniformly over the entire level-l *cell* as shown in Figure (6) on the left. This idea can be refined by matching higher-order moments of the characteristic function $\mathcal{X}_i^{\{l\}}$ with a regular $2^k \times 2^k$ array of characteristic functions associated with the cells at level $(l + k)$. For the choice $k = 2$, each coarse panel is associated with a set of $4 \times 4 = 16$ coefficients, as shown in Figure (6) on the right. If we let $\overline{\mathcal{X}}_i^{\{l\}}$ be the approximate characteristic funtion generated from the 4×4 cell array, the moment matching conditions require that the moments $Q_{\alpha\beta}^{(\gamma)}$ and $\overline{Q}_{\alpha\beta}^{(\gamma)}$ defined by

$$Q_{\alpha\beta}^{(\gamma)} = \int \mathcal{X}_i^{\{l\}} x^{\alpha} y^{\beta} dx dy, \quad \overline{Q}_{\alpha\beta}^{(\gamma)} = \int \overline{\mathcal{X}}_i^{\{l\}} x^{\alpha} y^{\beta} dx dy, \tag{19}$$

match exactly up to a certain order γ. The Cartesian moments defined in (19) for each order γ requires that $\alpha + \beta = \gamma$ and that α takes on all values $\{\ 0, \ldots, \gamma\ \}$. It can be shown that the if the Cartesian moments match up to order γ, then the *difference* between the the potentials generated by $\overline{\mathcal{X}}_i^{\{l\}}$ and $\mathcal{X}_i^{\{l\}}$ is of order $(1/r^{\gamma+1})$, but the proof is too long to state here.

Since each coarse-grid panel $p_i^{\{l\}}$ is now associated with 16 *geometric coefficients*, and since these coefficients correspond to cell currents at level $(l + k)$, it is possible to compute the $P^{\{l\}} q^{\{l\}}$ product in a matrix-free manner by using the DCT on a $2^{l+k} \times 2^{l+k}$ grid, similar to the finest level computation. However, since the nearest-neighbor panel interactions are not well approximated by moment matching, they are computed directly.

IV. Computational Results

We present numerical experiments comparing two iterative methods for solving (2): our new multigrid (MG) algorithm and the standard Generalized Minimal RESidual algorithm (GMRES [17]) without preconditioning. Since (2) results from a first-kind integral operator (1), the smallest eigenvalues of the matrix $P_{\{l\}}$ approach zero with increasing mesh refinement [12] and $P_{\{l\}}$ becomes more *ill-conditioned.* It is well-known that Krylov-subspace based iterative methods such as GMRES or CG (Conjugate Gradient) converge slowly for ill-conditioned linear systems [21]. Although it is

possible to appply *preconditioning* to accelerate GMRES convergence as was done in [14], an increasing number of iterations is still required for finer discretizations. We demonstrate that the multigrid algorithm resolves this difficulty by retaining a *constant* convergence rate per iteration, independent of mesh refinement, and hence problem size. Thus, for a given relative error tolerance, the number of multigrid iterations required is *fixed*.

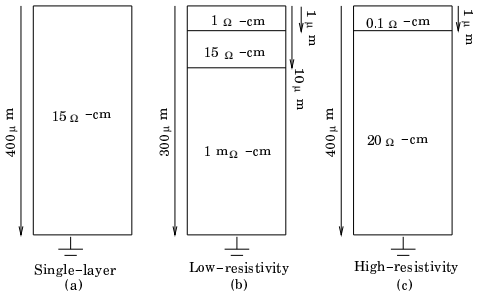

Fig. 7. Example substrate profiles.

In Figure 7, we display the three vertical substrate profiles used in this section: the single-layer substrate, the low-resistivity substrate, and the high-resistivity substrate. The lateral dimensions of the substrate is assumed to be 1mm $\times$ 1mm (or 1000 μm $\times$ 1000 μm).

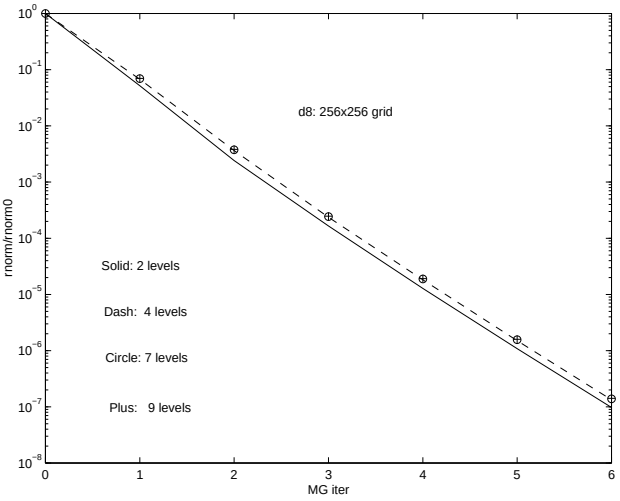

Fig. 8. Two-grid vs. Multigrid convergence.

The efficiency of multigrid algorithms in general arises from the fact that the *smoothing operator* at each level reduces the corresponding error components by the *same* numeric ratio [2]. This is demonstrated for the single-layer substrate example discretized with a 256 $\times$ 256 grid, and with a cell occupancy factor of one-half. The normalized residual, $\|P_{\{l\}}q_{\{l\}}^{(k)} - v_{\{l\}}\|/\|v_{\{l\}}\|$, is plotted versus the MG iteration count in Figure 8 for the two-grid method TGM (2 levels) and the multigrid method MGM (9 levels). The same convergence rate of about an order of magnitude per iteration is observed for multigrid methods of varying *depths*. We note that MGM requires only the *application* of operators P_l at various levels (and the solution of a scalar equation at the coarsest level $l = 0$), whereas TGM requires *solution* of the system at level $(l-1)$. Since the size of the linear system decreases *geometrically* with the level index, the cost of an MGM iteration is a constant multiple of an operator application at the finest level, or equivalently, a single GMRES iteration. Using the algorithms developed in Section III, we have observed this factor to be three to four in our implementation.

The crucial feature of multigrid schemes is that the convergence rate is independent of discretization, and hence problem size. We perform our next experiment with the single-layer substrate on five

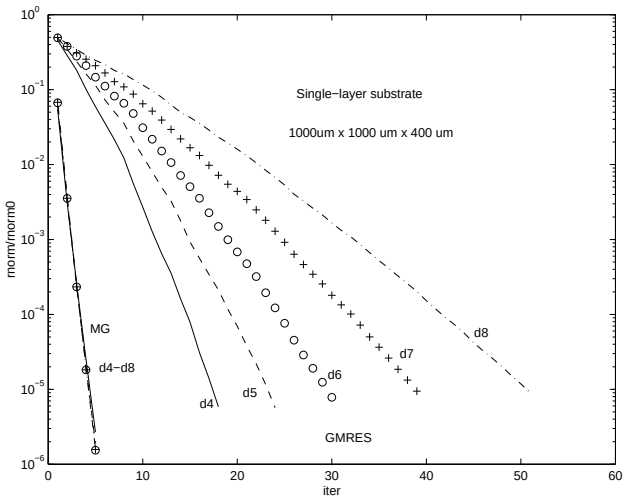

Fig. 9. Effect of mesh refinement on convergence.

test layouts of increasing mesh refinement, labeled $d = 4$ through $d = 8$. The $d = 4$ layout is discretized on a 16 $\times$ 16 grid, and the $d = 8$ layout is discretized on a 256 $\times$ 256 grid. The multigrid method with maximum depth ($l_{min} = 0, l_{max} = d$) is applied to solve each problem. The observed MG convergence rate is indeed independent of mesh size, as shown by the residual versus iteration plot in Figure 9. Also displayed are the GMRES convergence rates, which deteriorate with increasing mesh refinement as expected. Since the cost of a single MG iteration is a constant multiple of that of a GMRES iteration (three to four in our case), it is clear that MG is superior to GMRES, especially for large problems requiring fine discretization.

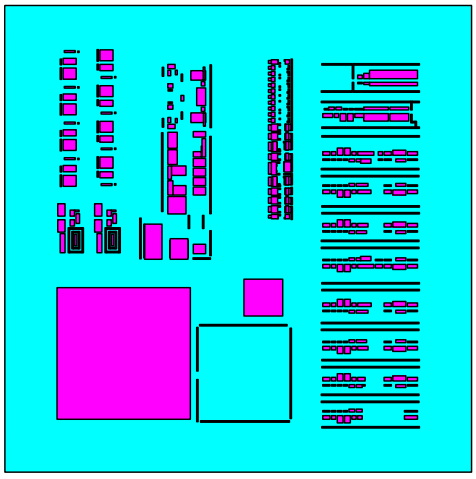

Fig. 10. PLL active area layout.

To show that the multigrid approach can be applied to realistic problems, we perform substrate parameter extraction on a a Phase Lock Loop (PLL) frequency synthesizer circuit [3] on a 1mm $\times$ 1mm chip. There are 478 substrate contacts defined by the active layer mask shown in Figure 10. Discretized with the help of a 1024 $\times$ 1024 grid, the total number of panels, or minimum-size cells, is $N_l = 183905$. This corresponds to roughly 20% of the chip area. The resolution thus achieved is about 1 micron.

The GMRES algorithm with sparsification via eigendecomposition [4] is used as a timing benchmark. We demonstrate the efficiency of MG versus GMRES by comparing the CPU times required to extract the 478 $\times$ 478 substrate conductance matrix. The substrate profile is assumed to be (a) single-layer, (b) low-resistivity, or (c) high-resistivity shown in Figure 7. We require convergence

to a tolerance of 1e−3 in the relative residual norm for each of the 478 solves. The timing results are summarized in Table I. It is seen from the total extraction time that MG is faster than GMRES by almost an order of magnitude for the low-resistivity and high-resistivity substrates. More significant gains will be seen for even larger problems requiring finer meshes.

	T_{setup}	T_{iter}	N_{iters}	T_{solve}	T_{total}
MG (a)	317s	28.5s	3	85.5s	11.4h
GMRES (a)	11.2s	7.32s	50	366s	48.6h
MG (b)	343s	28.4s	4	114s	15.2h
GMRES (b)	29.9s	8.1s	95	771s	102h
MG (c)	333s	28.0s	6	168s	22.3h
GMRES (c)	23.0s	8.4s	180	1512s	201h

TABLE I
COMPUTATIONAL COST FOR PLL SUBSTRATE EXTRACTION.

A possible limitation of the multigrid algorithm is that it resolves only ill-conditioning caused by mesh refinement. It is less effective in dealing with ill-conditioning caused by the apparent loss of the groundplane, as seen in a slowdown of MG convergence for the high-resistivity case (c). To maintain the optimal MG convergence rate, it is necessary to *explicitly solve* the problem at some level $l_{min} > 0$.

V. CONCLUSIONS AND ACKNOWLEDGEMENTS

In this paper we described a multigrid method for solving the first-kind integral formulation of the substrate extraction problem. We believe that the ideas proposed here can be generalized to solving other problems arising from first-kind integral equations defined over complicated surfaces, such as BEM capacitance extraction [14]. The authors wish to thank Ranjit Gharpurey of Texas Instruments and Edoardo Charbon of Cadence Design Systems for helpful discussions and for providing the PLL layout example.

ACKNOWLEDGEMENTS

This work was supported by ARPA, MURI, SRC, and grants from Harris Semiconductor and HP.

REFERENCES

[1] A. Brandt. Multi-level adaptive solutions to boundary-value problems. *Mathematics of Computation*, 31(138):333–390, April 1977.

[2] W. L. Briggs. *A Multigrid Tutorial*. Society for Industrial and Applied Mathematics, Philadelphia, 1987.

[3] E. Charbon, R. Gharpurey, R.G. Mayer, and A. Sangiovanni Vincentelli. Semi-analytical techniques for substrate characterization in the design of mixed-signal ics. In *International Conference on Computer Aided-Design*, pages 455–462, San Jose, CA, November 1996.

[4] J. P. Costa, M. Chou, and L. M. Silveira. Efficient techniques for accurate modeling and simulation of substrate coupling in mixed-signal ic's. In *DATE'98 - Design, Automation and Test in Europe, Exhibition and Conference*, pages 892–898, Paris, France, February 1998.

[5] R. Gharpurey and R.G. Meyer. Modeling and analysis of substrate coupling in integrated circuits. In *IEEE 1995 Custom Integrated Circuits Conference*, pages 125–128, 1995.

[6] Ranjit Gharpurey. *Modeling and Analysis of Substrate Coupling in Integrated Circuits*. PhD thesis, Department of Electrical Engineering and Computer Science, University of California at Berkeley, Berkeley, CA, June 1995.

[7] Ranjit Gharpurey and Robert G. Meyer. Modeling and analysis of substrate coupling in integrated circuits. *IEEE Journal Solid-State Circuits*, 31(3):344–353, March 1996.

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

[9] W. Hackbusch. *Multi-Grid Methods and Applications*. Springer-Verlag, Berlin Heidelberg New York Tokyo, 1985.

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

[11] T. A. Johnson, R.W. Knepper, V. Marcellu, and W. Wang. Chip substrate resistance modeling technique for integrated circuit design. *IEEE Transactions on Computer-Aided Design of Integrated Circuits*, CAD-3(2):126–134, 1984.

[12] R. Kress. *Linear Integral Equations*. Springer-Verlag, 1989.

[13] Sujoy Mitra, R. A. Rutenbar, L. R. Carley, and D. J. Allstot. A methodology for rapid estimation of substrate-coupled switching noise. In *IEEE 1995 Custom Integrated Circuits Conference*, pages 129–132, 1995.

[14] K. Nabors and J. White. Fastcap: A Multipole Accelerated 3-D Capacitance Extraction Program. *IEEE Transactions on Computer-Aided Design*, pages 1447–1459, November 1991.

[15] Bram Nauta and Gian Hoogzaad. How to deal with substrate noise in analog cmos circuits. In *European Conference on Circuit Theory and Design*, pages Late 12:1–6, Budapest, Hungary, September 1997.

[16] V. Rohklin. Rapid solution of integral equation of classical potential theory. *J. Comput. Phys.*, 60:187–207, 1985.

[17] Y. Saad and M. H. Schultz. Gmres: A generalized minimum residual algorithm for solving nonsymmetric linear systems. *SIAM J. Stat. Comp.*, 7:856–869, 1986.

[18] A. H. Schatz, V. Thomée, and W. L. Wendland. *Mathematical Theory of Finite and Boundary Element Methods*. Birkhäuser Verlag, Basel/Boston/Berlin, 1990.

[19] T. Smedes, N. P. van der Meijs, and A. J. van Genderen. Extraction of circuit models for substrate cross-talk. In *International Conference on Computer Aided-Design*, pages 199–206, San Jose, CA, November 1995.

[20] B.R. Stanisic, N.K. Verghese, R.A. Rutenbar, L.R. Carley, and D.J. Allstot. Addressing substrate coupling in mixed-mode ic's: Simulation and power distribution synthesis. *IEEEJSSC*, 29(3):226–238, March 1994.

[21] J. Stoer and R. Bulirsch. *Introduction to Numerical Analysis*. Springer-Verlag, second edition, 1993.

[22] D.K. Su, M.J. Loinaz, S. Masui, and B.A. Wooley. Experimental results and modeling techniques for substrate noise in mixed-signal integrated circuits. *IEEEJSSC*, 28(4):420–430, April 1993.

[23] Nishath Verghese. *Extraction and Simulation Techniques for Substrate-Coupled Noise in Mixed-Signal Integrated Circuits*. PhD thesis, Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA, August 1995.

[24] Nishath K. Verghese, David J. Allstot, and Mark A. Wolfe. Verification techniques for substrate coupling and their application to mixed-signal ic design. *IEEE Journal Solid-State Circuits*, 31(3):354–365, March 1996.

[25] Ivan L. Wemple and Andrew T. Yang. Mixed-signal switching noise analysis using voronoi-tesselation substrate macromodels. In 32^{nd} *ACM/IEEE Design Automation Conference*, pages 439–444, San Francisco, CA, June 1995.