A Markov Chain Based Hierarchical Algorithm for Fabric-Aware Capacitance Extraction

A Markov Chain Based Hierarchical Algorithm for Fabric-Aware Capacitance Extraction

Tarek El-Moselhy, Ibrahim (Abe) M. Elfadel, and Luca Daniel

***Abstract*—In this paper, we propose a hierarchical algorithm to compute the 3-D capacitances of a large number of topologically different layout configurations that are all assembled from the same basic layout motifs. Our algorithm uses the boundary element method in order to compute a Markov transition matrix (MTM) for each motif. The individual motifs are connected together by building a large Markov chain. Such Markov chain can be simulated extremely efficiently using Monte Carlo simulations (e.g., random walks). The main practical advantage of the proposed algorithm is its ability to extract the capacitance of a large number of layout configurations in a complexity that is basically independent of the number of configurations. For instance, in a large 3-D layout example, the capacitance calculation of 1000 different configurations assembled from the same motifs is accomplished in the time required to solve independently two configurations, i.e., a 500× speedup.**

***Index Terms*—Integrated circuit interconnections, interconnected systems, large scale integration, Markov processes, Monte Carlo methods, parameter extraction, parasitic capacitance.**

I. Introduction

RECENT integrated circuit technology nodes (32 nm and beyond) rely on regular patterns, uniform gridded designs and restricted design rules, in order to overcome many of the manufacturing-process related complications [1]–[3]. The main idea in such fabric-based design methodology, is to force designers to use a relatively small number of patterns (design fabrics, or design motifs), which are litho-friendly and preoptimized from a manufacturing prospective, in order to build arbitrary designs. As an example the reader is referred to Figs. 1 and 2, which represent two different configurations constructed from the *same* six motifs. Notice that with "configuration" we mean a structure constructed by a particular arrangement of the motifs. In this paper we propose a new hierarchical capacitance extraction algorithm specifically targeted to a large number of topologically-different structures which are constructed from a given set of motifs.

Manuscript received January 19, 2010; revised October 27, 2010; accepted October 28, 2010. Date of publication December 17, 2010; date of current version January 07, 2011. This work was supported by Mentor Graphics, AMD, IBM, the Semiconductor Corporations and the Interconnect Focus Center, one of five research centers funded under the Focus Center Research Program, a DARPA and Semiconductor Research Corporation program. This work was recommended for publication by Associate Editor D. Jiao upon evaluation of the reviewers comments.

T. El-Moselhy and L. Daniel are with the Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: tmoselhy@mit.edu; dluca@mit.edu).

I. Elfadel is with the IBM Corporation, Yorktown Heights, NY 10598 USA (e-mail: elfadel@us.ibm.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TADVP.2010.2091504

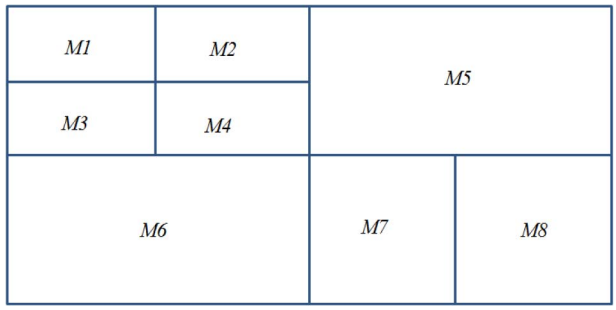

Fig. 1. Hypothetical layout pattern composed of a set of motifs. The arrangement of these motifs constitutes one configuration.

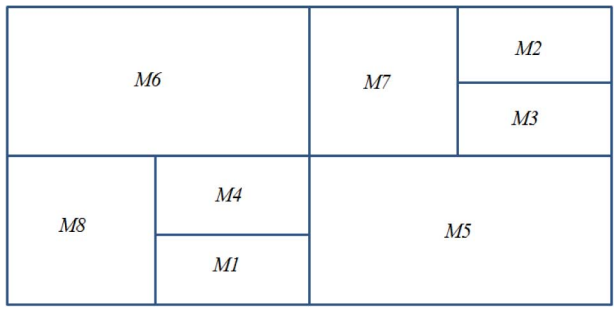

Fig. 2. An alternative layout configuration made of the same set of motifs defined in Fig. 1.

Hierarchical algorithms rely on first subdividing the geometry into smaller subdomains, then resolving the local interactions by computing each subdomain independently and finally connecting the different subdomains together by resolving the global interactions. Fabric-based design methodologies do not require an initial "subdivision" operation but rather an "assembly" operation of predefined motifs. Nonetheless, hierarchical methods can still be the ideal choice to handle fabric-based designs, since each design motif can represent a subdomain. Unfortunately, in the majority of the standard hierarchical algorithms (e.g., [4]–[6]) resolving the global interactions constitutes the bottleneck of the computation, since it relies on assembling and solving a large linear system of equations. Such operation must be repeated for each configuration, hence standard hierarchical algorithms are not suitable for the efficient extraction of a large number of different configurations.

Recently, we have proposed a hierarchical technique [7] which efficiently resolves the global interaction by using Markov chain theory. In fact, in such algorithm resolving the global interactions is a lot less expensive than resolving the local interactions. The complexity of resolving the local

TABLE I
CLASSIFICATION OF HIERARCHICAL ALGORITHMS

		Resolving local interactions	
		BEM	FRW
Resolving global interactions	Linear Systems	[4], [5], [6]	-
	Markov Chain	this paper	[7]

interactions is primarily related to the geometrical manipulations required by the standard floating random walk (FRW) algorithm. On the other hand, resolving the global interactions requires almost negligible complexity, since it involves only scalar computations as opposed to geometrical manipulations or large linear system solves.

Before proceeding with the details of this paper, we emphasize that just using "fast" or variation-aware extraction algorithms is not suitable for the task at hand. "Fast" algorithms (e.g., [8]–[11], etc.) rely on reducing the complexity of a *single* solve from $O(N^2)$ to $O(N)$. Unfortunately, in their present implementations, they do not reuse any of the computations between topologically-different configurations. Consequently, the complexity of such algorithms scales linearly with the total number of structures, which is computationally inefficient if such a number is large. Variation-aware extraction algorithms (e.g., [12]–[17]) typically handle problems, in which the topology is fixed, while the dimensions (width, thickness, or center–center distance) or shape (surface roughness) of the conductors vary. Hence, such methods do not account for variations which result in topological changes.

In this paper, the algorithm proposed in [7] is generalized. In particular, we demonstrate that the two steps corresponding to resolving the local and global interactions are totally independent (see Table I). Consequently, the particular technique best suited for computing each of the two steps can be independently chosen. For instance, it is well known that discretization based methods [such as boundary element method (BEM) or finite element method (FEM)] are significantly more efficient than discretization-free methods (such as FRW) for small/medium size structures. Consequently, in this paper we propose to use the BEM to compute the Markov transition matrices (MTMs) for each motif, unlike [7] in which the local interactions are resolved using the FRW. It is also known that for large systems, such as those that represent the global interactions, Monte Carlo based methods are significantly more efficient than linear system based methods. Consequently, we choose to resolve the global interactions by first building a large Markov chain from the computed MTMs. We then simulate very efficiently the resulting Markov chain using random walks. The efficiency of the resulting algorithm is near-optimal for fabric-aware extraction. In other words, we can compute the capacitance of a large number of configurations, constructed from the same set of motifs, in a complexity that is practically independent of the number of configurations.

The rest of the paper is organized as follows. In Section II we develop our new hierarchical algorithm (BEM-MTM). In Section III we develop a fabric-aware BEM-MTM algorithm for 3-D capacitance extraction of a large number of configurations constructed by different recompositions of a set of motifs. Finally, in Section IV we show a variety of examples validating our algorithm.

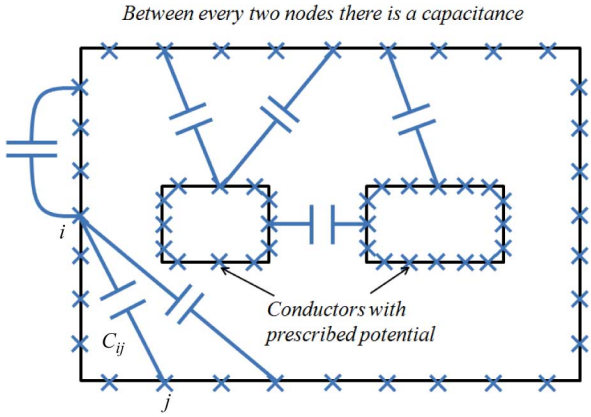

Fig. 3. The capacitance matrix of a single motif describes the connectivity between the nodes.

II. MAIN ALGORITHM: BEM-MTM

A. Computing Local Interactions Within Motifs

In our hierarchical algorithm, the local interactions within each motif are captured by computing a Markov transition matrix (MTM) for each motif. Such a MTM represents the probability of moving from any point on the surface of the conductors inside the motif or any point on the boundary of the motif, to any point on the surface of conductors inside the motif or on the boundary of the motif. There are a variety of ways to define the MTM. In particular, in [7] the MTM is defined as the relation between the *potential* at the points on the surface of the conductors inside of the motif, or on the boundary of the motif and the *potential* at the same points. In this paper we define a more general MTM, which relates the *charge* at the points on the surface of the conductors inside of the motif or on the boundary of the motif to the *potential* at the same points. Such a MTM can therefore be interpreted as a capacitance matrix (see Fig. 3)

$$\mathbf{q}_i = \sum_{j=1}^{N} \mathcal{C}_{ij}\mathbf{v}_j \tag{1}$$

where $\mathbf{q}_i$ is the charge at point i, $\mathbf{v}_j$ is the potential at point j and $\mathcal{C}_{ij}$ is the capacitance between i and j. Notice that $\mathbf{v}_j$ is the *absolute potential relative to infinity*. Equation (1) can be rewritten as

$$\mathbf{v}_i\mathcal{C}_{ii} - \mathbf{q}_i = -\sum_{j=1, j\neq i}^{N} \mathcal{C}_{ij}\mathbf{v}_j. \tag{2}$$

Equation (2) can be written in matrix form

$$\mathrm{diag}\{\mathcal{C}\}\mathbf{v} - \mathbf{q} = -\Delta\mathcal{C}\mathbf{v} \tag{3}$$

where $\mathrm{diag}\{\mathcal{C}\}$ is a diagonal matrix whose diagonal elements are equal to those of $\mathcal{C}$ and $\Delta\mathcal{C} = \mathcal{C} - \mathrm{diag}\{\mathcal{C}\}$.

Lemma II.1: The matrix $-(\mathrm{diag}\{\mathcal{C}\})^{-1}\Delta\mathcal{C}$ is a MTM, in which the element i, j represents the probability of moving from the node i to the node j inside of the same motif.

Proof: In order to prove that $-(\mathrm{diag}\{\mathcal{C}\})^{-1}\Delta\mathcal{C}$ is a properly defined MTM we need to show that it satisfies the following three properties.

- The diagonal element of each row is equal to zero.
- All elements of each row are strictly nonnegative
- The sum of any row is equal to 1.

The first statement is trivially true, since $\Delta\mathcal{C}$ has zero diagonal elements. The second statement follows from the fact that the capacitance matrix has positive diagonal elements and negative off-diagonal elements [18]. The third statement follows from the fact that in any closed domain, where no field is allowed to escape, the sum of any row of the capacitance matrix is zero [18]. Consequently,

$$\mathcal{C}_{ii} = -\sum_{j=1,j\neq i}^{N} \mathcal{C}_{ij}. \tag{4}$$

The sum of any row (e.g., the ith row) of $-(\mathrm{diag}\{\mathcal{C}\})^{-1}\Delta\mathcal{C}$ is

$$-\frac{1}{\mathcal{C}_{ii}} \sum_{j=1,j\neq i}^{N} \mathcal{C}_{ij}. \tag{5}$$

Substituting (4) in (5)

$$-\frac{1}{-\sum\limits_{j=1,j\neq i}^{N} \mathcal{C}_{ij}} \sum_{j=1,j\neq i}^{N} \mathcal{C}_{ij} = 1. \tag{6}$$

■

The motif MTM or equivalently the capacitance matrix in a closed domain (see Fig. 3) can be computed using any standard extraction algorithm, such as FRW, BEM, or FEM. In this paper, the MTM is computed using the BEM, since the motifs are typically small to medium size problems for which the BEM is very efficient. For ease of presentation, we assume that the medium is homogeneous with permittivity ε. For examples on how to apply the BEM in media with complex dielectric configurations, the reader is referred to [4], [5], [9]. Starting from the integral equation formulation

$$\frac{1}{2}\phi(r) = \int_{\Omega} \rho_V(r)G(r,r')d^3r' - \oint_{\Gamma} \varepsilon G(r,r')\nabla_n\phi(r')d^2r' + \oint_{\Gamma} \varepsilon\phi(r')\nabla_n G(r,r')d^2r' \tag{7}$$

where Ω is the computational domain, Γ is the union of the boundary of the motif and the surfaces of the conductors, $\hat{n}$ is the unit vector normal to Γ, $\nabla_n = \hat{n}\cdot\nabla$, $G(r,r') = 1/4\pi\varepsilon\|r-r'\|$ is the free space Green's function, $\rho_V(r)$ is the volumetric charge density, and $\phi(r)$ is the electric potential.

Using the fact that there are no volumetric charges $\rho_V(r) = 0$ and that the surface charge density $\rho(r') = -\varepsilon\nabla_n\phi(r')$, we obtain

$$\frac{1}{2}\phi(r) = \oint_{\Gamma} G(r,r')\rho(r')d^2r' + \oint_{\Gamma} \varepsilon\phi(r')\nabla_n G(r,r')d^2r'. \tag{8}$$

Equation (8) is solved for the unknown potential $\phi(r)$ and unknown surface charge density $\rho(r)$, which are defined *everywhere* on Γ. For simplicity, both the charge density and the potential are discretized using piecewise constant basis functions supported on rectangular panels. Then, collocation testing is used in order to obtain a linear systems of equations of the form

$$\left(\frac{1}{2}\mathbf{I} - \mathbf{M}_n\right)\mathbf{v} = \mathbf{M}\mathbf{q} \tag{9}$$

where $\mathbf{v}$ is a vector of the potential coefficients and $\mathbf{q}$ is a vector of the charge coefficients (notice the use of charges not charge densities), and

$$\mathbf{M}_n(i,j) = \fint_{S_j} \varepsilon\nabla_n G(r_i,r')d^2r' \tag{10}$$

$$\mathbf{M}(i,j) = \int_{S_j} \frac{G(r_i,r')}{a_i}d^2r' \tag{11}$$

where $\fint$ refers to the principal value of the integral and a_i refers to the area of panel i. Notice that both integrals are available in analytical closed form. Using (9) we obtain

$$\mathbf{q} = \mathbf{M}^{-1}\left(\frac{1}{2}\mathbf{I} - \mathbf{M}_n\right)\mathbf{v} \tag{12}$$

which leads to the following motif MTM

$$\mathcal{C} = \mathbf{M}^{-1}\left(\frac{1}{2}\mathbf{I} - \mathbf{M}_n\right). \tag{13}$$

B. Resolving the Global Interactions

When motifs are connected together, two different boundary conditions must be satisfied, namely, the continuity of the potential, and the continuity of the displacement field. Notice that there is no charge accumulation at the interface between motifs. When connecting motifs, each discretization panel can belong to at most two motifs. Writing (2) at two neighboring motifs (see Fig. 4), and summing the resulting expressions we obtain

$$\mathbf{v}_i^{(1)}\mathcal{C}_{ii}^{(1)} + \mathbf{q}_i^{(1)} + \mathbf{v}_i^{(2)}\mathcal{C}_{ii}^{(2)} + \mathbf{q}_i^{(2)} = -\sum_{j=1,j\neq i}^{N} \mathcal{C}_{ij}^{(1)}\mathbf{v}_j^{(1)} - \sum_{j=1,j\neq i}^{N} \mathcal{C}_{ij}^{(2)}\mathbf{v}_j^{(2)} \tag{14}$$

where the superscripts (1) and (2) refer to local quantities of the first motif and of the second motif, respectively. We have assumed without loss of generality that the common interface point is given the local index i in both neighboring motifs. Consequently, at the boundary point i: $\mathbf{v}_i^{(1)} = \mathbf{v}_i^{(2)} = \mathbf{v}_i^{(g)}$ and $\mathbf{q}_i^{(1)} + \mathbf{q}_i^{(2)} = 0$ we obtain

$$\mathbf{v}_i^{(g)}\left(\mathcal{C}_{ii}^{(1)} + \mathcal{C}_{ii}^{(2)}\right) = -\sum_{j=1,j\neq i}^{N} \mathcal{C}_{ij}^{(1)}\mathbf{v}_j^{(1)} - \sum_{j=1,j\neq i}^{N} \mathcal{C}_{ij}^{(2)}\mathbf{v}_j^{(2)} \tag{15}$$

which leads to

$$\mathbf{v}_i^{(g)} = \sum_{j=1,j\neq i}^{N} \frac{-\mathbf{C}_{ij}^{(1)}}{\left(\mathcal{C}_{ii}^{(1)} + \mathcal{C}_{ii}^{(2)}\right)}\mathbf{v}_j^{(1)} + \sum_{j=1,j\neq i}^{N} \frac{-\mathcal{C}_{ij}^{(2)}}{\left(\mathcal{C}_{ii}^{(1)} + \mathcal{C}_{ii}^{(2)}\right)}\mathbf{v}_j^{(2)}. \tag{16}$$

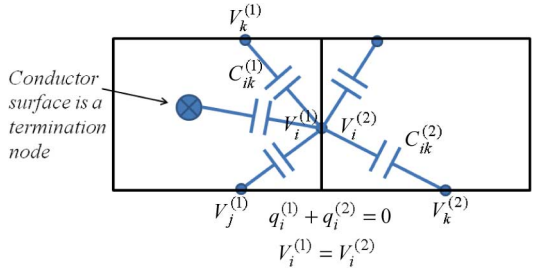

Fig. 4. Two different motifs connected together. At point i the boundary conditions are satisfied by combining the capacitance networks of both motifs.

Equation (16) should be understood in a more general (electrical circuits) context. The potential V_i is a weighted sum of the potential of *all* the nodes (in the global configuration) connected to the node i. The weights are the ratio of the capacitance on the link from i to j to the total capacitance connected to node i

$$\mathbf{v}_i^{(g)} = \sum_{j=1, j\neq i}^{N_i} \frac{-\mathcal{C}_{ij}^{(g)} \mathbf{v}_j^{(g)}}{\mathcal{C}_{ii}^{(g)}} \tag{17}$$

where $\mathcal{C}_{ii}^{(g)} = \mathcal{C}_{ii}^{(1)} + \mathcal{C}_{ii}^{(2)}$ and the summation is over all the N_i points that are connected to point i in the *global* network, i.e., after connecting the motifs. The weights $-\mathcal{C}_{ij}^{(g)}/\mathcal{C}_{ii}^{(g)}$, which represent the contribution of the potential at j to the potential at i, are interpreted as the probability of making a transition from i to j. Consequently, such a network of capacitances can be interpreted as a Markov chain and it will be simulated in the next section using Monte Carlo methods (e.g., random walks) [7], [20]–[22].

C. Complete BEM-MTM Algorithm for Capacitance Extraction

Section II-A defined local interactions using motif MTMs that only incidentally could be interpreted as intermediate capacitance matrices. These capacitance matrices represent the capacitance between the actual conductors present in the structure and additional nonphysical (virtual) motif boundaries, which are introduced when motifs are assembled together to form a configuration. Section II-B presented the equations necessary for resolving the global interactions between motifs. In this section, we show that, by combining the relations in Sections II-A and II-B, random walk methods can be used in order to reduce the large capacitance matrix generated when motifs are connected together (which includes capacitances between the conductors and the motif boundaries) to the desired smaller capacitance matrix between only the physical conductors in the configuration.

To extract the capacitance between conductors I and J, the total charge Q_I on the surface of the target conductor I as a consequence of a unit potential on the surface of the source conductor J is computed

$$Q_I(\mathbf{v}_J = 1) = \int_{S_I} \rho(r) dS = \sum_{i \in S_I} \mathbf{q}_i \tag{18}$$

where the summation is over all the panels discretizing the surface of conductor I. Consequently, the objective is to compute each individual $\mathbf{q}_i$, which is achieved by resorting to (2). Furthermore, since in any standard mutual capacitance extraction setup the voltage at the target wire is set to 0, i.e., $\mathbf{v}_i = 0$, (2) is reduced to

$$\mathbf{q}_i = (-\mathcal{C}_{ii}) \frac{\sum_{j=1, j\neq i}^{N} -\mathcal{C}_{ij} \mathbf{v}_j}{\mathcal{C}_{ii}}. \tag{19}$$

Notice that the term $-\mathcal{C}_{ij}/\mathcal{C}_{ii}$ represents the probability of making a transition from the starting panel i to some other panel j *inside or on the boundary* of the same motif. Consequently, the summation is computed using random walk methods by choosing a point on the motif boundary or on a conductor surface based on the probabilities $-\mathcal{C}_{ij}/\mathcal{C}_{ii}$. If the chosen point is on a panel with prescribed potential $\mathbf{v}_j$, then the path is assigned the random variable $-\mathcal{C}_{ii}\mathbf{v}_j$ and is terminated. Otherwise, the chosen point is on the boundary of the motif, and the path needs to be completed until reaching a conductor anywhere in the configuration. To complete the path, equation (17) is similarly computed using random walks. In such computation the weighting functions $w_{ij} = -\mathcal{C}_{ij}^{(g)}/\mathcal{C}_{ii}^{(g)}$ are interpreted as the probability of making a transition from the point i to the point j *in the global network*. This is implemented by drawing randomly a number r from a uniform distribution between 0 and 1. The point j is chosen such that $\sum_{k=1}^{j-1} w_{ik} < r$ and $\sum_{k=1}^{j} w_{ik} > r$. If point j is on a conductor surface then the random variable $-\mathcal{C}_{ii}$ [see (19)] is assigned to the path and the path is terminated. The process of generating paths is then repeated a large number of times. The unknown charge $\mathbf{q}_i$ is finally computed using $\mathbf{q}_i = -\mathcal{C}_{ii}\mathbb{E}[\mathbf{v}_j]$, where $\mathbb{E}[]$ is the expectation operator

$$\mathbf{q}_i = -\mathcal{C}_{ii} \frac{\#\text{of paths from } I \text{ to } J}{\text{total } \#\text{ paths starting from } I}. \tag{20}$$

Important Note: Since in the capacitance extraction setup only one conductor is assigned a nonzero potential at a time, only paths terminating at that particular conductor will have a nonzero contribution to the random variables. Consequently, the capacitance from conductor I to all other conductors can be computed simultaneously. If the target conductor is the same as the source conductor (i.e., we want to extract the self capacitance of a conductor) then v_i in (2) is equal to 1 and the value of $\mathcal{C}_{ii}$ is added to q_i.

Algorithm 1 summarizes our complete algorithm for computing the Ith row of the capacitance matrix. Fig. 5 shows two different possible paths.

D. Theoretical Analysis of BEM-MTM

Many standard "absorbing Markov chain" theorems and well known results can be exploited to certify properties of our BEM-MTM algorithm once we show how it is possible to appropriately construct a large MTM $\mathcal{T}$ for the entire configuration

$$\mathcal{T} = \begin{bmatrix} \mathbf{Q} & \mathbf{R} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \tag{21}$$

Algorithm 1 BEM-MTM for a given configuration

repeat
  choose randomly a point i on the the surface of the target conductor I.
  using (19) take a step starting from point i
  if reached a point on the interface between motifs **then**
    repeat
      make a transition using the capacitance matrices of both motifs defining the interface (see (17))
    until a conductor or a configuration boundary are reached
  end if
  if terminated on a conductor J **then**
    add $-\mathcal{C}_{ii}$ to capacitance $\mathbf{C}(I,J)$
  else { terminated on configuration boundary}
    add $-\mathcal{C}_{ii}$ to stray capacitance $\mathbf{C}(I,0)$
  end if
  $\mathbf{C}(I,I) \leftarrow \mathbf{C}(I,I) + \mathcal{C}_{ii}$
until convergence achieved
divide $\mathbf{C}(I,:)$ by the total number of walks

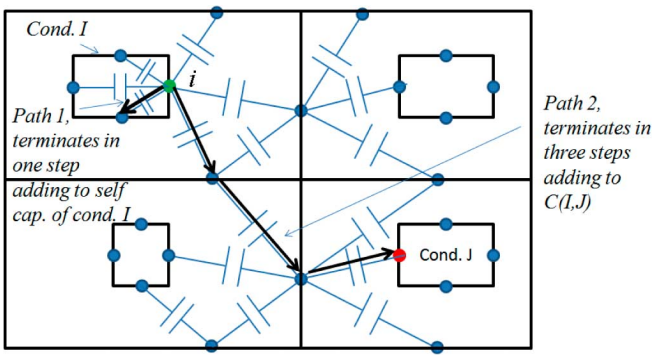

Fig. 5. Two paths starting from point i. The first terminates in one step on conductor I, the second terminates in three steps on conductor J in a different motif.

where $\mathbf{Q}$ is the transition probability matrix between "transient states" (in our case any point on the interface between motifs as shown in Fig. 6); $\mathbf{R}$ is the transition probability matrix from the transient states to the "absorbing states" (in our case all the conductors and any point on the external configuration boundary). Matrices $\mathbf{0}$ and $\mathbf{I}$ simply define the behavior of the absorbing states: once an absorbing state is reached the probability to remain in it is one, and consequently the probability to transition to any other state is zero. The upper part $[\mathbf{Q}\ \mathrm{R}]$ of the MTM $\mathcal{T}$ can be related to the individual MTMs of each motif using the Law of Total Probability

$$[\mathbf{Q} \quad \mathbf{R}](i,j) = \frac{-C_{ij}^{(g)}}{C_{ii}^{(g)}} \tag{22}$$

where $|C_{ij}|$ is the total capacitance between the points i and j in the global capacitance network; and C_{ii} is the total capacitance connected to node i. Notice that there is no capacitance between i and j if they do not belong to the same motif, there is a single capacitance if they belong to the same motif but not

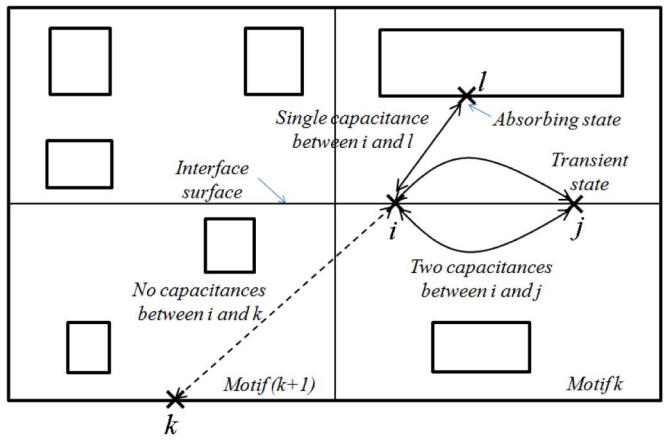

Fig. 6. Transition to interfaces (transient states) contribute to the $\mathbf{Q}$ matrix and transition to conductors and configuration boundaries (absorbing states) contribute to $\mathbf{R}$.

the same interface, and there are two different capacitances if i and j belong to a common interface.

Having cast our BEM-MTM algorithm as an absorbing Markov chain problem it is easy to rigorously answer many legitimate questions using the literature available on that topic [23]. For instance, the following theorem can be used to prove the termination of each BEM-MTM "path" in a finite number of transitions, and to even provide a precise estimate on the average number of transitions before termination.

Theorem II.1: Assume that BEM-MTM starts at a point i on an interface between motifs (i.e., a transient state), then the average length of the walk on interfaces, or expected number of transitions before reaching a conductor or the configuration boundary (i.e., an absorbing state), is finite and is given by the row sum of the ith row in the "fundamental matrix" $\mathbf{N} = (\mathbf{I} - \mathbf{Q})^{-1}$.

Proof: The proof follows directly from Theorem 3.3.5 in [23]. ■

E. Computational Complexity

Time: Assume that the structure was assembled using N_M different kind of motifs and that each motif k is discretized using a total of n_k panels (on boundary and conductor surfaces). The complexity of computing the kth motif MTMs is $O(n_k^3)$. Notice that the inverted matrix is exactly the capacitance matrix of the first type Fredholm formulation, which means that we can use fast algorithms to compute n_k matrix solves. However, in general the independent motifs are sufficiently require fast algorithms. All the motifs are independent. Consequently, the computation is embarrassingly parallelizable and the total complexity for computing all motifs is bounded by $\sum_k^{N_M} O(n_k^3/N_{\mathrm{procc}})$, where N_{procc} is the total number of processors.

The second part of the algorithm consists of resolving the global interaction. For that we need to initiate a large number of random walks. The complexity is linear in the number of walks $O(N_w)$. This part of the algorithm is also embarrassingly parallelizable, consequently the complexity is $O(N_w/N_{\mathrm{procc}})$. The main computational advantage of our algorithm is that the complexity constant associated with this part of the computation

is very small, since there are absolutely no geometrical manipulations involved. Instead, all transitions, are implemented by purely numeric manipulations, and are consequently extremely cheap.

Memory: The memory required for resolving the local interactions is $O(\max_k n_k^2)$, which is the memory required for the computation of the MTM of the motif discretized using the largest number of panels [see (9)]. On the other hand, resolving the global interactions requires storing the MTMs of all motifs for a total memory of only $O(\sum_k^{N_M} n_k)$, since in general the MTMs are very sparse (an example will be presented in the result section). Typically, a total of 100 transition matrices, corresponding to 100 different kind of motifs, can be stored in 2 GB of RAM. Notice that the actual number of instantiations of such motifs in the design configurations can be orders of magnitude larger than the number of kinds of motifs without any significant increase of the memory requirement.

F. Comparison Versus Other Hierarchical Algorithms

Compared to the HFRW [7] (which is a FRM-MTM algorithm), our new algorithm avoids using the computationally expensive FRW to compute the MTM of each motif. Such computations are expensive since they involve geometrical manipulations. As mentioned previously, our algorithm does not involve any of the FRW geometrical manipulations. Similar to the HFRW algorithm, our BEM-MTM uses the MTMs to resolve the global interactions. On the other hand, the main advantage of using the HFRW is that it can handle edge-defined variations extremely efficiently, whereas the BEM-MTM can only handle topological variations.

The main difference between our algorithm and the standard BEM hierarchical methods ([4], [5]) is that our algorithm resolves the global interactions without assembling/solving any expensive linear systems. Instead, our algorithm resolves the global interactions using random walks to simulate the Markov Chains. For large structures, such random walk methods are significantly more efficient than solving large linear systems of equations. In addition, BEM-MTM fully exploits the local nature of the capacitance extraction problem. Moreover, the BEM-MTM can efficiently extract the capacitance of a very large number of topologically different configurations constructed from the same set of motifs (see Section III).

Compared to approximate hierarchical algorithms [6], [11] and window based techniques [24], [25], which all rely on some form of approximation to achieve computational efficiency, our BEM-MTM algorithm is exact. In general, one of the main advantages of our BEM-MTM is that it exploits the local nature of the capacitance problem to achieve computational efficiency without the need to introduce approximations. In particular, the random walk simulation of the Markov chain exploits the fact that most of the capacitance of a wire is determined by the nearby interactions, since paths connecting a conductor to its immediate neighbors are typically very short and are computed extremely efficiently.

G. Comparison Versus the Standard FRW

There is apparent similarity between the proposed algorithm and the standard FRW [26], since the latter relies on first determining a transition domain, then making a transition from inside of the transition domain to its boundary, and repeating such process until the path terminates at a conductor surface with known potential. However, there exist a variety of fundamental differences between the two algorithms. Foremost, in the FRW the notion of transition domains is restricted. Indeed, in the standard FRW [26] a transition domain is a domain that includes no conductors and must have homogeneous media. In the generalized FRW [16] the transition domains are allowed to include nonhomogeneous media, however, they are not allowed to include any charged conductors. Such constrains typically result in small transition domains. On the other hand, in the BEM-MTM the notion of transition domain is generalized to be basically any closed domain. Indeed, by this new definition, the transition domain is now allowed to include complex dielectric structures, floating potential metal fills and even charged conductors. Such generality allows for declaring a whole predefined motif as a single transition domain.

Another difference is that due to the large constrains on what is defined as a transition domain in the FRW, the transition probability matrix is available in closed form and does not need to be numerically computed. Even in the generalized FRW, since the domain is relatively constrained and small in size, computing the transition matrix numerically is a trivial task. In our algorithm since the transition domains are large and general, the MTMs must be computed numerically. However, since the domains are predetermined by the set of motifs available in the fabric-aware design methodology, the MTMs can be precomputed only once for a given technology and then reused for all walks in all design configurations that use those motifs.

The most important difference is that in the FRW the transition domains are overlapping and are constructed online along the way of the path. Most of the computational time is spent on the geometrical manipulations required to determine the dimensions of the transition domains. In the BEM-MTM algorithm, the transition domains (which actually coincide with the design motifs) are non-overlapping, and are predetermined (rather than recomputed at each step). Therefore, the BEM-MTM algorithm does not require any geometrical manipulations. Indeed, in the BEM-MTM the walk, which starts and terminates at conductor surfaces and proceeds only on interfaces between motifs, is fully implemented using only numerical manipulations without the need for determining any transition domain or any other geometrical manipulations.

To summarize, our algorithm could be interpreted as an extremely generalized FRW in which the domains are predetermined, nonoverlapping, and large. The transition probability matrix is computed once by using a discretization-based solution to a given PDE. In terms of implementation, all FRW-related geometrical manipulations are removed and only the underlying Markov chain is very efficiently simulated using random walks.

III. Fabric-Aware Extraction

The most important advantage, and in general the main motivation behind using the BEM-MTM algorithm, is that it offers a very efficient algorithm to extract the capacitance of a very

Algorithm 2 BEM-MTM for all configurations

1: **for** each motif kind **do**
2: compute the MTM
3: **end for**
4: **for** each desired capacitance matrix row $\mathbf{C}(I,:)$ **do**
5: **repeat**
6: Make a random walk step starting from conductor I
7: **if** step reached a point on the interface between motifs **then**
8: **for** each configuration **do**
9: use MTMs to walk on interfaces and terminate on a conductor or configuration boundary
10: add $-\mathcal{C}_{ii}$ to appropriate capacitance
11: **end for**
12: **else** {step reached a conductor}
13: add $-\mathcal{C}_{ii}$ to appropriate capacitance
14: **end if**
15: $\mathbf{C}(I,I) \leftarrow \mathbf{C}(I,I) + \mathcal{C}_{ii}$
16: **until** convergence achieved
17: divide $\mathbf{C}(I,:)$ by the total number of walks
18: **end for**

large number of configurations constructed by different arrangements of a given set of motifs. Indeed, since in our BEM-MTM resolving the global interaction is cheaper than computing the local interactions, we can afford to repeat such a computation for a very large number of different configurations without significantly increasing the complexity of our algorithm. Furthermore, the majority of paths (corresponding to the self term and the strong coupling terms) terminate in the motif containing the target conductor and do not even have to be continued. Consequently, such paths are computed only once and are shared among all different configurations without the need for any recomputations. This in turn explains our observation that the BEM-MTM exploits the local nature of the capacitance extraction problem.

Algorithm 2 summarizes the steps of our proposed approach for 3-D capacitance extraction of a large number of configurations constructed by different recompositions of a set of motifs. Recall that if we have N_T motif instantiations then there are $O(N_T!)$ possible different configurations.

Since all configurations are constructed from the same set of $N_M \ll N_T$ motifs, the N_M motif MTMs can be precomputed separately as shown in Step 2. The complexity of this part of the algorithm depends linearly on the number of different kinds of motifs $O(N_M)$, and does not depend on the total number of configurations, $O(N_T!)$. Step 6 in Algorithm 2 is independent of the configuration structure, and therefore it is implemented once per motif and reused for all configurations. This step is very efficient since it involves only generating one random variable and choosing a corresponding point on the domain boundary or a conductor surface.

The remaining part of each path depends on the particular recomposition of the motifs, therefore it must be executed separately for each configuration. Since this part of the algorithm does not involve any geometrical manipulations it is extremely cheap. Consequently, the bottleneck of the algorithm is Step 2, and the complexity of our algorithm is almost completely independent of the total number of configurations.

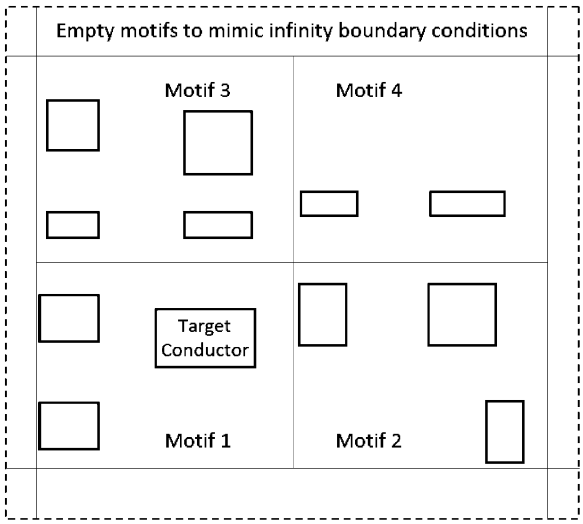

Fig. 7. Cross section in the structure of example IV-A. Motifs' boundaries are identified on the cross-section by solid lines. Empty motifs at the boundary (shown only partially) are used to mimic the infinity boundary condition.

IV. Results

Unless otherwise stated the reported results are obtained using our own Matlab implementations of the corresponding algorithms on a Intel Duo CPU at 2.4 GHz with 2 GB of memory. Unless otherwise stated, all floating random walk related results have been obtained by using a total of one million random walks, i.e., $N_w = 10^6$.

A. Accuracy Validation

The first example validates the accuracy of the proposed algorithm. A cross section of the 3-D geometry for this example is shown in Fig. 7, and is composed of 12 conductors. The geometry was assembled using four different motifs. In addition, empty motifs are added at the boundary (shown only partially in Fig. 7) to mimic the infinity boundary condition. In order to verify the accuracy of our approach, we extracted the capacitance between a target conductor in motif 1 (see Fig. 7) and all the other 11 conductors using both our BEM-MTM and the standard BEM method. Using 2×10^6 paths, the BEM-MTM is $10\times$ slower than the standard BEM and obtained for all extracted capacitances a 0.1% accuracy compared to the BEM method.

B. A Small 3-D Fabric-Aware Extraction Example

In this example we use the same motifs used in Fig. 7 (from the previous example IV-A) to construct a total of $4! = 24$ different configurations, corresponding to all possible different recompositions of the four internal motifs. The capacitance matrices of each of the configurations are computed using both our BEM-MTM and the standard BEM. All the values of the computed capacitances for each configuration are within 0.5% of the values computed using the standard BEM. The total time to compute the capacitance matrices of *all* 24 configurations using our BEM-MTM is about ($12\times$) the time required to compute the capacitance matrix of just *one* configuration using the standard

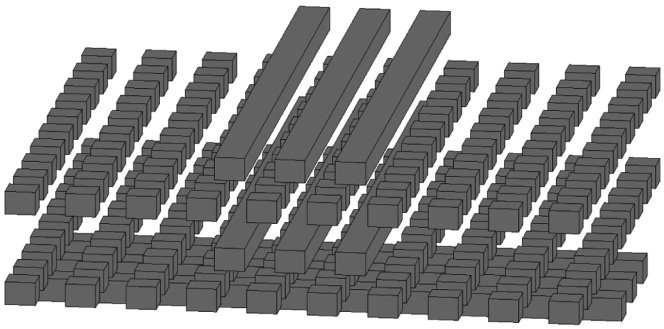

Fig. 8. A five layers, 209 conductor structure. Each layer is treated as a motif.

BEM. This corresponds to a 2× speedup. The reason we did not obtain large speed-up ratios is that this example is small.

C. A Large 3-D Example

In this subsection we demonstrate that the BEM-MTM can treat large 3-D structures very efficiently. The example under consideration is a five layer structure (Fig. 8). Two of such layers contain 100 cubic shaped conductors arranged on a 10×10 grid. The size of the conductors is 100 nm. These small conductors represent "metal fill," i.e., small floating potential conductors inserted in empty regions of the layout to facilitate the planarization. Each of the other three layers contain 3 parallel long wires of dimensions $100\ \mathrm{nm} \times 1400\ \mathrm{nm} \times 100\ \mathrm{nm}$. The wires are separated by 100 nm. Each of the five layers is 300 nm thick and has a unique dielectric constant. Each layer is treated as a motif. We recompose such motifs to construct a total of 120 different configurations. Notice that every configuration includes a total of 209 total conductors. Four different capacitances are extracted from each configuration.

The time required to compute the MTMs of the five motifs is approximately 6 min. The largest MTM is of dimensions 1536 × 1636, and is 95% sparse (Fig. 9). The computational time required to solve all possible 120 configurations is 15 min. Using the HFRW [7] to compute the capacitance of all 120 configurations requires 30 min. Running a standard boundary element method hierarchical algorithm (HBBEM) on all 120 configurations requires about 600 min. Running the standard FRW on the same example would require an estimated 1800 min. Running FastCap (C code) on the same set of 120 configurations with the same 5% accuracy requires a total of 270 min. Hence, our Matlab BEM-MTM is 1.5× faster than HFRW,[1] 20× faster than Matlab HHBEM, 9× faster than C-code FastCap and 60× faster than the standard FRW.

D. A Very Large Example

In this example we extract the capacitances of a 3-D, 10-layer very large structure. Each layer is composed of 20 parallel long conductors and has a unique dielectric constant. Each layer was assembled using 4 motifs, resulting in a total of 40 motifs. On average, each motif is discretized using 4000 unknowns. The time used to compute the complete capacitance matrix for all

[1]Notice that the HFRW requires 15 min to compute the different MTMs and 15 min to do the subsequent extraction. Our BEM-MTMS provides only 2.5× reduction in computational times required to compute the MTMs, since the motifs in this example are very dense and can therefore be efficiently computed using the FRW.

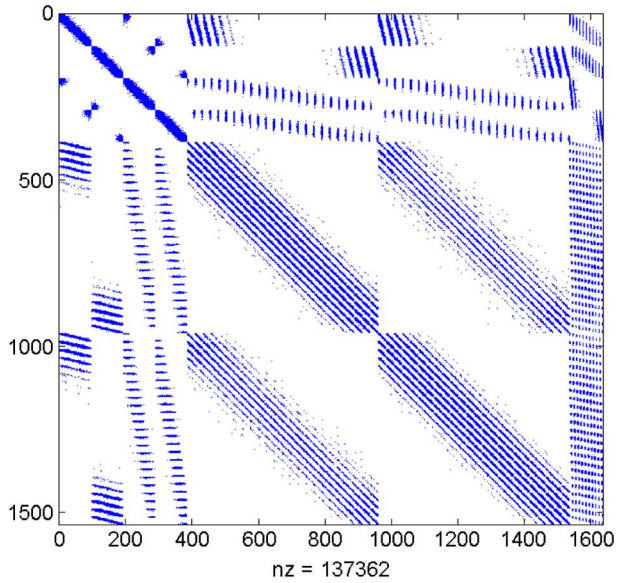

Fig. 9. A typical sparsity pattern of the MTM for a motif including 100 cubic conductors.

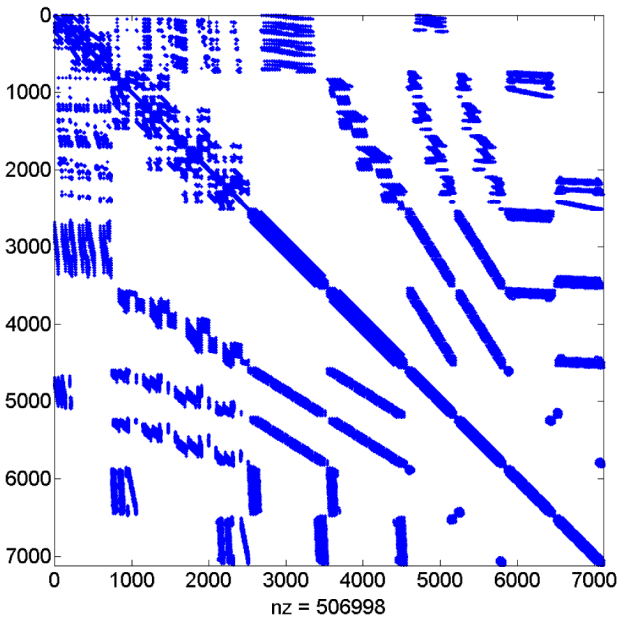

Fig. 10. The MTM corresponding to the largest motif of example IV-D.

motifs is less than 50 min. The average transition matrix is about 99% sparse, which means that storing *all* the transition matrices requires less 0.2 GB of memory. Fig. 10 shows the sparsity pattern corresponding to the MTM of the largest domain. Using the computed MTMs we extract the capacitance of a total of 1000 randomly generated different configurations in 45 min, which corresponds to slightly less than the time required to compute the MTMs. When using the HFRW [7] to compute the capacitance of all 1000 configurations 495 min are required (450 min to compute the MTMs and 45 min to extract the capacitance of the configurations). Hence, our algorithm provides 5× reduction in computational time. Due to large memory requirements (more than 4 GB), this example cannot be solved using our implementation of the standard hierarchical algorithms [4], [5] (which rely on assembling/solving a linear systems of equations).

V. Conclusion

In this paper, we have presented a hierarchical extraction algorithm for computing the 3-D capacitances of a large number of layout configurations that, although topologically different, are all assembled from the same layout motifs. Our BEM-MTM algorithm uses the boundary element method to efficiently compute a MTM for each motif. The motifs are connected together to form a large Markov chain. Such a chain is then simulated very efficiently using random walks. Consequently, our algorithm avoids the main complexity of standard hierarchical algorithms by not relying on assembling/solving linear systems to resolve the global interactions. Furthermore, our approach does not rely on approximations and trade-off between accuracy and computational efficiency. The main advantage of our algorithm is its extreme efficiency in extracting the capacitance matrix of a large number of configurations constructed by different recompositions of the same set of motifs. We have observed that its complexity is almost independent of the number of configurations. In particular, in a large example, the total time required to compute all the capacitance matrices of 1000 different 3-D configurations is only two times the time required for solving a single configuration. This is equivalent to a 500$\times$ speedup.

References

[1] T. Jhaveri, V. Rovner, L. Pileggi, A. J. Strojwas, D. Motiani, V. Kheterpal, K. Y. Tong, T. Hersan, and D. Pandini, "Maximization of layout printability/manufacturability by extreme layout regularity," *J. Micro/Nanolithography, MEMS MOEMS* vol. 6, no. 3, p. 031011, 2007.

[2] L. Liebmann, "DFM, the teenage years," *Design Manufacturability Through Design-Process Integration II*, vol. 6925, p. 692502, 2008.

[3] S. Banerjee, P. Elakkumanan, L. W. Liebmann, J. A. Culp, and M. Orshansky, "Electrically driven optical proximity correction," in *SPIE Conf.*, Apr. 2008, vol. 6925.

[4] W. Yu, Z. Wang, and J. Gu, "Fast capacitance extraction of actual 3-D VLSI interconnects using quasi-multiple medium accelerated bem," *IEEE Trans. Microwave Theory Tech.*, vol. 51, no. 1, pp. 109–119, Jan. 2003.

[5] T. Lu, Z. Wang, and W. Yuc, "Hierarchical block boundary-element method (hbbem): A fast field solver for 3-d capacitance extraction," *IEEE Trans. Microwave Theory Tech.*, vol. 52, no. 1, pp. 10–19, Jan. 2004.

[6] L. J. Jiang, B. J. Rubin, J. D. Morsey, H. T. Hu, and A. Elfadel, "Novel capacitance extraction method using direct boundary integral equation method and hierarchical approach," in *Electrical Performance Electron. Packag.*, Oct. 2006, pp. 331–334.

[7] T. El-Moselhy, I. Elfadel, and L. Daniel, "A hierarchical floating random walk algorithm for fabric-aware 3-D capacitance extraction," in *IEEE/ACM Int. Conf. Computer-Aided Design*, Nov. 2009.

[8] J. R. Phillips, J. K. White, and A. Member, "A precorrected-FFT method for electrostatic analysis of complicated 3-D structures," *IEEE Trans. Computer-Aided Design Integr. Circuits Syst.*, vol. 16, pp. 1059–1072, 1997.

[9] K. Nabors and J. White, "Fastcap: A multipole accelerated 3-D capacitance extraction program," *IEEE Trans. Computer-Aided Design Integr. Circuits Syst.*, vol. 10, no. 11, pp. 1447–1459, Nov. 1991.

[10] W. Hackbusch, "A sparse matrix arithmetic based on $\mathcal{H}$-matrices. Part I: Introduction to $\mathcal{H}$-matrices," *Computing* vol. 62, no. 2, pp. 89–108, 1999.

[11] W. Shi, J. Liu, N. Kakani, and T. Yu, "A fast hierarchical algorithm for 3-D capacitance extraction," in *Proc. 35th Annu. Design Automat. Conf.*, New York, 1998, pp. 212–217.

[12] R. Jiang, W. Fu, J. M. Wang, V. Lin, and C. C. P. Chen, "Efficient statistical capacitance variability modeling with orthogonal principle factor analysis," in *Proc. 2005 IEEE/ACM Int. Conf. Computer-Aided Design*, Washington, DC, 2005, pp. 683–690.

[13] Z. Zhu and J. White, "Fastsies: A fast stochastic integral equation solver for modeling the rough surface effect," in *EEE/ACM Int. Conf. Computer-Aided Design 2005*, Nov. 2005, pp. 675–682.

[14] H. Zhu, X. Zeng, W. Cai, and D. Zhou, "A spectral stochastic collocation method for capacitance extraction of interconnects with process variations," in *IEEE Asia Pacific Conf. Circuits Syst.*, Dec. 2006, pp. 1095–1098.

[15] T. El-Moselhy and L. Daniel, "Stochastic integral equation solver for efficient variation-aware interconnect extraction," in *Proc. 45th ACM/IEEE Design Automat. Conf.*, Jun. 2008, pp. 415–420.

[16] T. El-Moselhy, I. Elfadel, and L. Daniel, "A capacitance solver for incremental variation-aware extraction," in *IEEE/ACM Int. Conf Computer-Aided Design*, Nov. 2008, pp. 662–669.

[17] Z. Ye, Z. Zhu, and J. Phillips, "Incremental large-scale electrostatic analysis," *IEEE Trans. Computer-Aided Design Integrated Circuits Syst.*, vol. 28, no. 11, pp. 1641–1653, Nov. 2009.

[18] A. E. Ruehli, *Circuit Analysis, Simulation, and Design*. Amsterdam, The Netherlands: North Holland, 1986.

[19] J. Gu, Z. Wang, and X. Hong, "Hierarchical computation of 3-D interconnect capacitance using direct boundary element method," in *Proc. Asia South Pacific Design Automat. Conf.*, 2000, pp. 447–452.

[20] H. Qian, S. Nassif, and S. Sapatnekar, "Power grid analysis using random walks," *IEEE Trans. Computer-Aided Design Integrated Circuits Syst.*, vol. 24, no. 8, pp. 1204–1224, Aug. 2005.

[21] H. Qian and S. S. Sapatnekar, "Hierarchical random-walk algorithms for power grid analysis," in *Proc. 2004 Asia South Pacific Design Automat. Conf.*, 2004, pp. 499–504.

[22] P.-Y. Huang, C.-K. Lin, and Y.-M. Lee, "Hierarchical power delivery network analysis using Markov chains," in *IEEE Int. SOC Conf.*, Sep. 2007, pp. 283–286.

[23] J. Kemeny and J. Snell, *Finite Markov Chains*, ser. Univ. Series Undergrad, Math,. New York: VanNostrand, 1969.

[24] M. Beattie and L. Pileggi, "Error bounds for capacitance extraction via window techniques," *, IEEE Trans. Computer-Aided Design Integrated Circuits Syst.*, vol. 18, no. 3, pp. 311–321, Mar. 1999.

[25] F. Yu and W. Shi, "A divide-and-conquer algorithm for 3-D capacitance extraction," in *Int. Symp. Quality Electronic Design*, 2004, pp. 253–258.

[26] Y. L. Coz and R. Iverson, "A stochastic algorithm for high speed capacitance extraction in integrated circuits," *Solid-State Electron.* vol. 35, no. 7, pp. 1005–1012, 1992.

Tarek El-Moselhy received the B.Sc. degree in electrical engineering in 2000 and a diploma in mathematics in 2002, then the M.Sc. degree in mathematical engineering, in 2005, all from Cairo University, Cairo, Egypt. He received the Ph.D. degree in electrical engineering from Massachusetts Institute of Technology, Cambridge, in 2010.

He is a postdoctoral associate in the Department of Aeronautics and Astronautics at Massachusetts Institute of Technology (MIT). His research interests include fast algorithms for deterministic and stochastic electromagnetic simulations, stochastic algorithms for uncertainty quantification in high dimensional systems, and stochastic inverse problems with emphasis on Bayesian inference.

Ibrahim (Abe) M. Elfadel received his undergraduate education at the Ecole Centrale des Arts et Manufactures, Paris, France, on a scholarship from the French Government. He received the Ph.D. degree from Massachusetts Institute of Technology, Cambridge, in 1993.

He is a Senior Scientist in Electronic Design Automation at the IBM Systems and Technology Group. Prior to this position, he was a Research Staff Member in Design Automation at IBM Research. He joined IBM in 1996 after stints in the telecommunication industry with KDD Research and Development Laboratories, Saitama, Japan, the medical device industry with Masimo Corporation, Irvine, CA, and as a visiting scientist with the MIT Research Laboratory of Electronics. Between 2000 and 2004, he was on the faculty of Columbia University as an Adjunct Associate Professor of Electrical Engineering. His current research interests are primarily in VLSI layout electrical extraction,

circuit and interconnect analysis, computational electromagnetics, and device compact modeling. His past R&D work includes contributions to statistical image modeling, adaptive signal processing, neural networks, model-order reduction, circuit optimization, and transmission line macromodeling. He is the holder of 45 issued and pending U.S. patents and the author of numerous refereed publications and internal reports.

Dr. Elfadel is the recipient of an IBM Outstanding Technical Achievement Award for his contributions to the automation of VLSI interconnect modeling, analysis, and optimization, and an IBM Research Division Award for his contributions to timing-driven VLSI circuit optimization. He is also the recipient of six IBM Invention Achievement Awards for his contributions to IBM's IP portfolio in his various areas of technical expertise. His other awards include an ACM Technical Leadership Award from the Special Interest Group on Design Automation, and a "Certificate of Recognition for Technology Innovation" from the Society of Technology in Anesthesia. He served on the Technical Program Committees of several conferences, including the International Conference on Computer-Aided Design and the European Design Automation and Test Conference. He also served as the Chair of the Professional Interest Community on Design Automation at IBM Research.

Luca Daniel received the Laurea degree (*summa cum laude*) in electronic engineering from the Universita di Padova, Italy, in 1996, and the Ph.D. degree in electrical engineering from the University of California, Berkeley, in 2003.

He is an Associate Professor in the Electrical Engineering and Computer Science Department of the Massachusetts Institute of Technology (MIT), Cambridge. His research interests include accelerated integral equation solvers and parameterized stable compact dynamical modeling of linear and nonlinear dynamical systems with applications in mixed-signal/RF/mm-wave circuits, power electronics, MEMs, and the human cardiovascular system.

Dr. Daniel received the 1999 IEEE TRANSACTIONS ON POWER ELECTRONICS best paper award, the 2003 ACM Outstanding Ph.D. Dissertation Award in Electronic Design Automation, five best paper awards in international conferences, the 2009 IBM Corporation Faculty Award, and 2010 Early Career Award from the IEEE Council on Electronic Design Automation.