A Time-Domain Layered Finite Element Reduction Recovery (LAFE-RR) Method for High-Frequency VLSI Design

Houle Gan and Dan Jiao, *Senior Member, IEEE*

***Abstract*—A fast and high-capacity electromagnetic solution, time-domain layered finite element reduction recovery (LAFE-RR) method, is proposed for high-frequency modeling and simulation of large-scale on-chip circuits. This method rigorously reduces the matrix of a multilayer system to that of a single-layer system, regardless of the problem size. More importantly, the matrix reduction is achieved analytically, and hence the CPU and memory overheads are minimal. The recovery of solutions in all other layers involves only forward and backward substitution of matrices of single-layer size. The memory cost is also modest-requiring only the memory needed for the factorization of two sparse matrices of half-layer size. The superior performance applies to any arbitrarily shaped multilayer structure. Numerical and experimental results are presented to demonstrate the accuracy, efficiency, and capacity of the proposed method.**

***Index Terms*—Electromagnetics, finite element methods, high frequency, on-chip circuits, time domain analysis.**

I. Introduction

As ON-CHIP circuits have scaled into the deep submicron regime (and the nanometer regime), full-wave electromagnetics-based analysis has increasingly become essential for the following reasons:

- *Reduced feature sizes*. At the 45 nm processing technology node and beyond, the IC industry will have to print features that are several times less than the wavelength of light (193nm) being used in optical lithography. In this regime, the wave nature of light is manifest. In order to ensure IC can be manufactured as designed, it is crucial to model light as a wave rather than as a ray approximation.
- *Increased clock frequency*. Currently the clock frequency of microprocessors is in the gigahertz regime. Since it is necessary to analyze the chip response to harmonics 5 times the clock frequency, it is expected that interconnects at high frequencies would have to be analyzed with certain electromagnetic effects incorporated. The importance of electromagnetic (EM) analysis at tens of GHz was quantitatively demonstrated, via simulation and real silicon measurements in [1], [7].
- *The transition from single core to multicore*. Active power management of multicore processors requires large-scale EM analysis of the global power supply network in order to accurately model current variations and transient power drops and ground bounce.
- *Increased level of integration*. Integrating RF, analog, and digital circuitry on a same chip leads often to undesired coupling and sometimes to system failure. For high-frequency mixed-signal design, an electromagnetic solution is required to overcome the fundamental limits of circuit-based solutions.

Since the advent of computational electromagnetics (CEM) in the 1960s, numerous fast algorithms have been developed [2]. They have been widely applied to microwave engineering, antenna analysis, scattering analysis, wireless communications, and optoelectronics. They have also been applied to board and packaging problems. However, traditional CEM technology has been found not amenable for very-large-scale on-chip design problems [3]. This is mainly because VLSI design demands very large-scale electromagnetic solutions, which cannot be offered by many current CEM techniques. In addition, the unique modeling challenges of on-chip problems [3] further complicate electromagnetic analysis.

In view of the importance and challenges of full-wave electromagnetic analysis in high-frequency VLSI design, in recent years, CEM techniques have been developed for on-chip problems [4]–[13]. They can be categorized into two classes: partial differential equation (PDE) based and integral equation (IE) based methods. A representative of the former method is the finite difference time domain (FDTD) method. Two-dimensional FDTD approaches have been developed for full-wave modeling of on-chip transmission line structures. For 3-D on-chip structures, an FDTD solver was recognized as computationally expensive. In contrast, the partial element equivalent circuit (PEEC) method is an IE-based method [4], [5], [9]. PEEC was first utilized to solve quasi-static problems. It was then extended to full-wave analysis. Recently, surface-based PEEC methods have also been developed which significantly reduce the number of unknowns involved in a volume IE-based PEEC method [5], [9]. Since IE-based methods incorporate dense matrices, a direct solver requires $O(N^3)$ operations and $O(N^2)$ storage in dealing with N number of unknowns, which is computationally expensive. The surface-based PEEC is under way to be accelerated by fast algorithms such as the fast multipole method, FFT-based method, adaptive integral method, and fast QR-based methods. Another surface IE-based

Manuscript received April 20, 2007; revised July 31, 2007. This work was supported by the Office of Naval Research under award N00014-06-1-0716 and by a grant from Intel Corporation

The authors are with the School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN 47907.

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/TAP.2007.910473

formulation was developed in [6]. A fast pre-corrected FFT scheme was formulated to accelerate the iterative solution of the dense matrix equation. The multi-layered dielectric has not yet been included. IE -based methods have also been developed in [8], [10], [11] for the application to on-chip problems. It is expected that an IE-based solution that takes all the on-chip intricacies into consideration, accelerated by the fast algorithms, will soon be completed.

However, ultra large scale IC design results in numerical problems of ultra large scale, requiring billions of parameters to describe them accurately. To solve N number of parameters, the optimal computational complexity one can hope for is linear complexity $O(N)$. However, even $O(N)$ is inadequate in practice since N is too large in practical integrated systems. Therefore, it is important to develop high-capacity electromagnetic solutions that can overcome the $O(N)$ barrier to achieve $O(M)$ with $M \ll N$. In addition, it is desirable that the solutions be obtained in a *rigorous* fashion. Along this line of thought, a time-domain layered finite element reduction recovery (TD-LAFE-RR) method is proposed herein to solve large-scale IC design problems at high frequencies. This method rigorously reduces the matrix of a multilayer system to that of a single-layer system regardless of the original problem size. More importantly, the matrix reduction is achieved analytically, and hence the CPU and memory overheads are minimal. In addition, the reduced system matrix preserves the sparsity of the original system matrix. Compared to the layered finite-element method developed in [12], the proposed method further improves the modeling capacity and performance since the matrix reduction is achieved analytically and the reduced system matrix is sparse. In addition, since the method is developed in the time domain, the method permits nonlinear modeling and broadband simulation within one simulation.

The remainder of this paper is as follows. In Section II, the problem statement is presented. In Section III, the time-domain finite-element framework is introduced. Section IV delineates the proposed time-domain LAFE-RR method. In Section V, numerical and experimental results demonstrate the capacity, efficiency, and accuracy of the proposed method.

II. Problem Statement

Consider the 3-D integrated circuits in Fig. 1. Generally, these circuits are multilayered structures. They are embedded in a multilayer dielectric media backed by silicon, GaAs, InP, SiC, or other semiconductor substrates. A Manhattan-type integrated circuit as shown in Fig. 1(a) and (b) is layered in any of the x-, y-, and z-directions.

Inside these circuits, the electric field $\mathbf{E}$ satisfies the second-order vector wave equation

$$\nabla \times [\mu_r^{-1} \nabla \times \mathbf{E}(\mathbf{r},t)] + \mu_0 \varepsilon \partial_t^2 \mathbf{E}(\mathbf{r},t) + \mu_0 \sigma \partial_t \mathbf{E}(\mathbf{r},t) = -\mu_0 \partial_t \mathbf{J}(\mathbf{r},t) \quad \text{in } V \tag{1}$$

subject to certain boundary conditions. In (1), μ, ε, σ are permeability, permittivity, and conductivity, respectively; $\mathbf{J}$ is the current source; V is the computational domain that encloses the circuit. Due to the computational complexity of on-chip circuits, the resultant numerical system is generally prohibitively large even for small circuits. In this work, a new time-domain finite-element method is proposed to address the challenges associated with the large problem size.

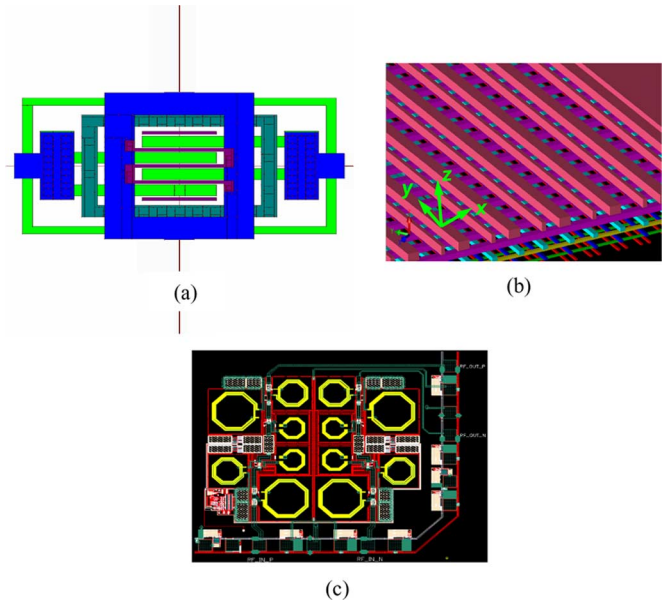


Fig. 1. Illustration of 3-D IC problems. (a) A RF CMOS device. (b) Global on-chip interconnects. (c) A RF IC.

III. Time-Domain Finite-Element Framework

Compared to other time-domain CEM methods such as FDTD-based methods and time-domain IE-based methods, a time-domain finite-element method (TDFEM) [14]–[21] deals with sparse matrices as well as possessing increased capability in handling irregular geometries. In this section, the basic numerical scheme of the TDFEM is outlined.

The solution to the boundary value problem defined by (1) and its boundary conditions can be obtained by seeking the stationary point of the functional

$$\begin{aligned} F[\mathbf{E}(\mathbf{r},t)] &= \frac{1}{2} \iiint_V \{\mu_r^{-1} [\nabla \times \mathbf{E}(\mathbf{r},t)] \cdot [\nabla \times \mathbf{E}(\mathbf{r},t)] \\ &\quad + \mu_0 \varepsilon \partial_t^2 \mathbf{E}(\mathbf{r},t) \cdot \mathbf{E}(\mathbf{r},t) \\ &\quad + \mu_0 \sigma \partial_t \mathbf{E}(\mathbf{r},t) \cdot \mathbf{E}(\mathbf{r},t)\} \, dV \\ &\quad + \iiint_V \mu_0 \partial_t \mathbf{J}(\mathbf{r},t) \cdot \mathbf{E}(\mathbf{r},t) \, dV \\ &\quad - \frac{1}{2} \iint_S \mathbf{E}(\mathbf{r},t) \cdot P(\mathbf{E}(\mathbf{r},t)) \, dS. \end{aligned} \tag{2}$$

In (2), S is the surface that encloses computational domain V, and $P(\mathbf{E})$ is the absorbing boundary condition imposed on S to absorb the outgoing wave [12]. To discretize (2), the computational domain is divided into small elements. In each element, $\mathbf{E}(\mathbf{r},t)$ is expanded as

$$\mathbf{E}(\mathbf{r},t) = \sum_{i=1}^{m} u_i(t) \mathbf{N}_i^e(\mathbf{r}) \tag{3}$$

where m denotes the total number of expansion terms, and $\mathbf{N}_i(\mathbf{r})$ and u_i are the vector expansion functions and corresponding expansion coefficients, respectively. Substituting (3) into (2), taking the partial derivative of the functional with respect to the expansion coefficients u_i, and setting the resulting equation to zero yields the following system of ordinary differential equations [16]:

$$\mathbf{T}\frac{d^2u}{dt^2}+\mathbf{R}\frac{du}{dt}+\mathbf{S}u+w=j \tag{4}$$

in which $\mathbf{T},\mathbf{R}$, and $\mathbf{S}$ are square matrices, and u, w, and j are column vectors. Their elements are given by

$$\begin{aligned}\mathbf{T}_{ij}&=\mu_0\varepsilon\langle\mathbf{N}_i,\mathbf{N}_j\rangle_V \quad \mathbf{R}_{ij}=\mu_0\sigma\langle\mathbf{N}_i,\mathbf{N}_j\rangle_V\\ \mathbf{S}_{ij}&=\mu_r^{-1}\langle\nabla\times\mathbf{N}_i,\nabla\times\mathbf{N}_j\rangle_V\\ w_i&=-\langle\mathbf{N}_i,P\rangle_S \quad j_i=-\mu_0\langle\mathbf{N}_i,\partial_t\mathbf{J}\rangle_V\end{aligned} \tag{5}$$

where $\langle .,.\rangle_V$ and $\langle .,.\rangle_S$ denote volume and surface integration, respectively.

Adopting a central difference scheme to approximate the first- and second-order time derivatives in (4), we obtain

$$\begin{aligned}\mathbf{P}u^{n+1}=(2\mathbf{T}-\Delta t^2\mathbf{S})u^n+[0.5\Delta t\mathbf{R}-\mathbf{T}]u^{n-1}\\ -\Delta t^2w^n+j^n\end{aligned} \tag{6}$$

in which

$$\mathbf{P}=\mathbf{T}+0.5\Delta t\mathbf{R} \tag{7}$$

and Δt represents the time step. Clearly, u^{n+1} (the field value at the $(n+1)-$th time step) can be solved in a time marching fashion from the solution of u at previous time steps. The dimension of $\mathbf{P}$ can be very large for realistic on-chip problems, which constitutes a computational challenge. An efficient and high-capacity solution, time-domain LAFE-RR method, is proposed to solve this problem in the following section.

IV. Proposed Time-Domain Layered Finite-Element Reduction Recovery (LAFE-RR) Method

In the description that follows, the number of layers is denoted by L, the total number of unknowns is denoted by N, the number of volume unknowns per layer is N_V, and the number of top/bottom surface unknowns is N_s. The stack-growth direction is defined as z. The layer-growth direction can be chosen the same as the stack-growth direction; it can also be chosen from x and y.

A. Discretization

First, the computational domain is discretized into triangular prism elements. Conductors are also discretized to accurately capture the fields inside conductors. In each prism element, the electric field is expanded into prism vector bases [22] as shown in Fig. 2. These functions can be written as

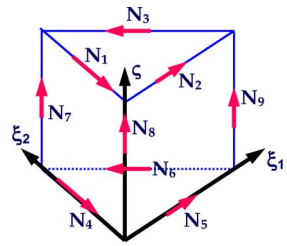


Fig. 2. Illustration of prism vector basis functions.

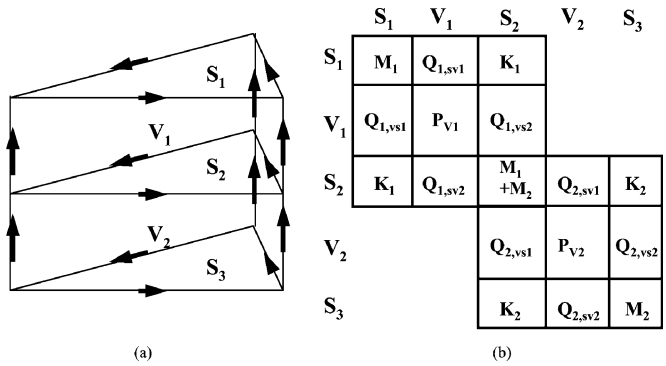


Fig. 3. (a) Unknown ordering scheme. (b) Matrix pattern.

$$\begin{aligned}\mathbf{N}_1&=\zeta(\xi_1\nabla\xi_2-\xi_2\nabla\xi_1)=\varsigma\mathbf{W}_1\\ \mathbf{N}_2&=\zeta(\xi_2\nabla\xi_3-\xi_3\nabla\xi_2)=\varsigma\mathbf{W}_2\\ \mathbf{N}_3&=\zeta(\xi_3\nabla\xi_1-\xi_1\nabla\xi_3)=\varsigma\mathbf{W}_3\\ \mathbf{N}_4&=(1-\zeta)(\xi_1\nabla\xi_2-\xi_2\nabla\xi_1)\\ \mathbf{N}_5&=(1-\zeta)(\xi_2\nabla\xi_3-\xi_3\nabla\xi_2)\\ \mathbf{N}_6&=(1-\zeta)(\xi_3\nabla\xi_1-\xi_1\nabla\xi_3)\\ \mathbf{N}_7&=\xi_1\nabla\zeta,\mathbf{N}_8=\xi_2\nabla\zeta,\mathbf{N}_9=\xi_3\nabla\zeta.\end{aligned} \tag{8}$$

In (8), ξ_1,ξ_2, and ξ_3 are area coordinates; ς is 0 at the bottom plane, and 1 at the top plane; $\mathbf{W}_i$ are the well-known edge basis functions.

The unknowns are ordered layer-by-layer starting from surface unknowns of S_1, and proceeding to volume unknowns of V_1, surface unknowns of S_2, volume unknowns of V_2, and surface unknowns of S_3, as shown in Fig. 3(a). Proceeding in this way, a banded matrix is generated, as illustrated in Fig. 3(b). Clearly, it is less computationally expensive to solve a banded matrix than to solve a general matrix. However, when the number of unknowns is large, the solution remains computationally intensive. To solve this problem, an algorithm is proposed to analytically reduce the 3-D layered matrix system to a single-layer one.

B. Analytical Reduction of the 3-D System Matrix

First, the 3-D layered system matrix is reduced to a 2-D layered one. To form a matrix system that only involves 2-D

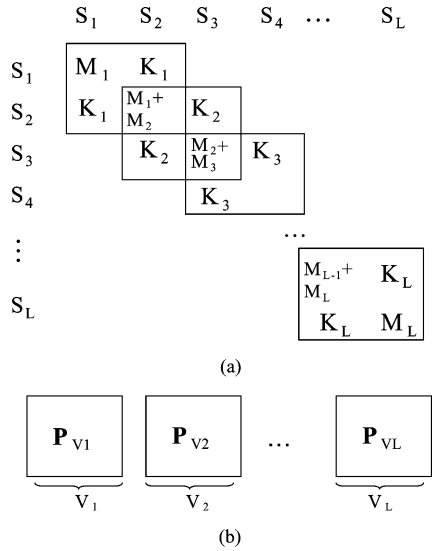


Fig. 4. Decomposition of the original 3-D layered system matrix. (a) A sparse matrix formed by surface unknowns in all layers. (b) L decoupled sparse matrices formed in each layer by volume unknowns.

surface unknowns in each layer, volume unknowns are eliminated. Since the number of volume unknowns is LN_V, at least $O(LN_V)$ operations and storage are required to eliminate them. Here, the unknowns can be eliminated analytically without any CPU and memory cost. This is achieved by utilizing the orthogonality between volume unknowns and surface unknowns. As shown in Fig. 2, the volume vector bases $\boldsymbol{N}_7, \boldsymbol{N}_8$, and $\boldsymbol{N}_9$ are perpendicular to surface vector bases $\boldsymbol{N}_i (i = 1-6)$, and hence the coupling between volume unknowns and surface unknowns is zero in matrices $\mathbf{T}$ and $\mathbf{R}$. As a result, volume unknowns and surface unknowns are decoupled in $\mathbf{P}$, i.e., all the $\mathbf{Q}$ matrices in Fig. 3(b) vanish, and hence volume unknowns are analytically eliminated from the final matrix equation. As a result, the final matrix is decomposed into one sparse matrix that involves only 2-D surface unknowns in all layers as shown in Fig. 4(a), which is denoted by $\mathbf{P}_s$, and L decoupled small sparse matrices formed by volume unknowns in each layer, $\mathbf{P}_{Vl} (l = 1, 2, \ldots, L)$, as shown in Fig. 4(b). Hence, matrix equation (6) is decomposed into

$$\begin{aligned} \mathbf{P}_s x_s &= b_s \\ \mathbf{P}_{Vl} x_{Vl} &= b_{Vl} \quad l = 1, 2, \ldots, L. \end{aligned} \tag{9}$$

Note that this does not lead to the conclusion that there is no physical coupling between volume and surface unknowns. In fact, the surface unknowns influence the volume ones by altering their right hand sides b_s at each time step through the stiffness matrix $\mathbf{S}$ in (6); similarly, the volume unknowns influence the surface ones by changing their right hand sides b_{Vl} at each time step.

Next, a procedure is proposed to analytically reduce the 2-D layered system matrix to a single-layer one. Denoting the layer to be reduced to as i, a top-down-bottom-up procedure is performed to eliminate other layers and project their contributions to layer i. For the bottom-up elimination, as shown in Fig. 5, the finite-element sub-matrix in the Lth layer is first formed. It is

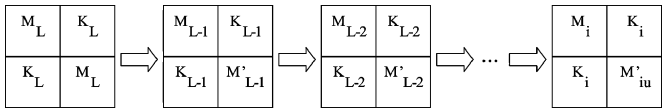


Fig. 5. Illustration of the bottom-up elimination.

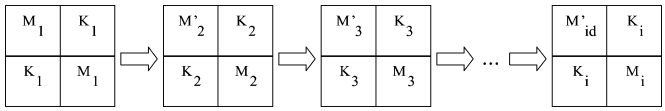


Fig. 6. Illustration of the top-down elimination.

then eliminated and its contribution is projected to the $(L-1)$th layer. Next, the modified $(L-1)$th layer is projected to the $(L-2)$th layer. This procedure is continued until reaching the ith layer. From left to right, Fig. 5 lists the sub-matrix in the $L-$th layer, the modified sub-matrix in the $(L-1)$th layer, …, and the modified sub-matrix in the ith layer. Mathematically, it can be represented as

$$\begin{aligned} \mathbf{M}'_{L-1} &= \mathbf{M}_{L-1} + \mathbf{M}_L - \mathbf{K}_L \mathbf{M}_L^{-1} \mathbf{K}_L \\ \mathbf{M}'_{L-2} &= \mathbf{M}_{L-2} + \mathbf{M}_{L-1} - \mathbf{K}_{L-1} \mathbf{M}_{L-1}^{'-1} \mathbf{K}_{L-1} \\ &\cdots \\ \mathbf{M}'_{iu} &= \mathbf{M}_i + \mathbf{M}_{i+1} - \mathbf{K}_{i+1} \mathbf{M}_{i+1}^{'-1} \mathbf{M}_{i+1}. \end{aligned} \tag{10}$$

For the top-down elimination, as shown in Fig. 6, first the finite-element sub-matrix in the first layer is formed. It is then eliminated and its contribution is projected down to the second layer. Next, the contribution from the modified second layer is projected down to the third layer. This procedure is continued until reaching the ith layer. Mathematically it can be written as

$$\begin{aligned} \mathbf{M}'_2 &= \mathbf{M}_1 + \mathbf{M}_2 - \mathbf{K}_1 \mathbf{M}_1^{-1} \mathbf{K}_1 \\ \mathbf{M}'_3 &= \mathbf{M}_2 + \mathbf{M}_3 - \mathbf{K}_2 \mathbf{M}_2^{'-1} \mathbf{K}_2 \\ &\cdots \\ \mathbf{M}'_{id} &= \mathbf{M}_{i-1} + \mathbf{M}_i - \mathbf{K}_{i-1} \mathbf{M}_{i-1}^{'-1} \mathbf{M}_{i-1}. \end{aligned} \tag{11}$$

The system of equations in (10) and (11) involve matrix inverse and matrix-matrix multiplication in each layer, which can be computationally intensive. However, they can be performed *analytically* because $\mathbf{M}$ and $\mathbf{K}$ are assembled from their elemental contributions as

$$\begin{aligned} \mathbf{M}^e_{l,ij} &= \frac{h_l}{3} \left(\mu_0 \varepsilon_l^e + 0.5 \Delta t \mu_0 \sigma_l^e \right) \iint_{\Omega^e} \mathbf{W}_i \cdot \mathbf{W}_j d\Omega \\ \mathbf{K}^e_{l,ij} &= \frac{h_l}{6} \left(\mu_0 \varepsilon_l^e + 0.5 \Delta t \mu_0 \sigma_l^e \right) \iint_{\Omega^e} \mathbf{W}_i \cdot \mathbf{W}_j d\Omega \\ & \qquad\qquad l = 1, 2, \ldots L \end{aligned} \tag{12}$$

in which subscript l is the layer index, Ω^e denotes each triangular element, $\mathbf{W}_i$ is the edge vector basis function as shown in (8), h_l is the thickness of layer l, ε_l^e is the permittivity of layer l in element e, and σ_l^e is the conductivity of layer l in element e. If the layer-growth direction is chosen as x or y, different layers would feature the same permittivity configuration. If the

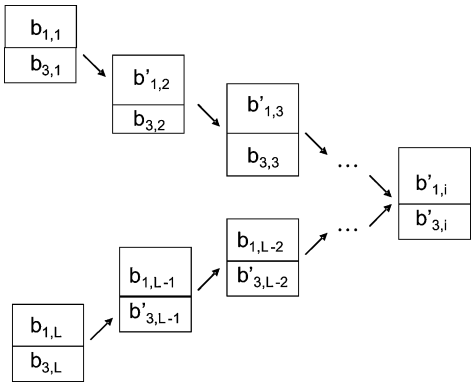


Fig. 7. Updates in the right hand side.

layer-growth direction is chosen as z, then the elements in each layer share the same permittivity. In either case, $\mathbf{M}$ and $\mathbf{K}$ are linearly proportional to permittivity as well as layer thickness. If different layers have different conductivity configuration and the conductivity in each layer is not a constant, the term associated with loss, i.e., matrix $\mathbf{R}$ in (4), is moved to the right hand side. Hence, $\mathbf{M}_l(l = 1, 2, \ldots, L)$ are linearly proportional to each other, and $\mathbf{K}_l(l = 1, 2, \ldots, L)$ are linearly proportional to each other. Moreover, $\mathbf{M}_l(l = 1, 2, \ldots, L)$ are also linearly proportional to $\mathbf{K}_l(l = 1, 2, \ldots, L)$. Therefore, $\mathbf{M}_l(l = 1, 2, \ldots, L)$ of primed quantities are also linearly proportional to $\mathbf{K}_l(l = 1, 2, \ldots, L)$ as can be seen from (10) and (11). As a result, the reduction shown in Figs. 5 and 6, which is mathematically represented by (10) and (11), is achieved without any numerical calculation. To be more specific, if $\mathbf{M}_1$, which is $\mathbf{M}$ in the first layer, is taken as the reference, then $\mathbf{M}'_{id}$ and $\mathbf{M}'_{iu}$ can be obtained instantly by scaling $\mathbf{M}_1$ by a certain coefficient made of permittivity and layer thickness in different layers. This also shows that the reduced single-layer matrix preserves the sparsity of the original system matrix.

The right hand sides of each layer are also updated during the reduction process as illustrated in Fig. 7. As an example, the bottom-up path can be recursively written as

$$\begin{aligned} b'_{3,L-1} &= b_{3,L-1} - \mathbf{K}_L \mathbf{M}_L^{-1} b_{3,L} \\ b'_{3,L-2} &= b_{3,L-2} - \mathbf{K}_{L-1} \mathbf{M}_{L-1}^{'-1} b'_{3,L-1} \\ &\cdots \\ b'_{3,i} &= b_{3,i} - \mathbf{K}_{i+1} \mathbf{M}_{i+1}^{'-1} b'_{3,i+1}. \end{aligned} \tag{13}$$

Again there is no need to perform matrix inversion and matrix vector multiplication in (13) because $\mathbf{M}$ and $\mathbf{K}$ are linearly proportional to each other, and $\mathbf{M}$ of primed quantities and $\mathbf{K}$ are also linearly proportional to each other.

After the aforementioned reduction process, the original system matrix equation is rigorously reduced to the following matrix equation that only involves the surface unknowns (x_1 and x_3) in layer i

$$\begin{pmatrix} \mathrm{M}'_{id} & \mathrm{K_i} \\ \mathrm{K_i} & \mathrm{M}'_{iu} \end{pmatrix} \begin{pmatrix} x_{1,\mathrm{i}} \\ x_{3,\mathrm{i}} \end{pmatrix} = \begin{pmatrix} \mathrm{b}'_{1,\mathrm{i}} \\ \mathrm{b}'_{3,\mathrm{i}} \end{pmatrix} \tag{14}$$

which can be solved readily. In (14), $\mathbf{M}'_{id}$ incorporates the effects from layers above i, and $\mathbf{M}'_{iu}$ carries the contributions from those below. Due to the analytical reduction process, the final reduced matrix is obtained instantly without the need of any computation.

Equation (14) can be further reduced to

$$\mathbf{M}''_{i,d} x_{1,i} = b''_{1,i} \tag{15}$$

where $\mathbf{M}''_{i,d} = \mathbf{M}'_{i,d} - \mathbf{K}_i (\mathbf{M}'_{i,u})^{-1} \mathbf{K}_i$ and $b''_{1,i} = b'_{1,i} - \mathbf{K}_i \mathbf{M}_{iu}^{'-1} b'_{3,i}$. As in the previous case, since $\mathbf{M}$ of primed quantities and $\mathbf{K}$ are linearly proportional to each other, there is no need to perform matrix inverse, matrix-matrix multiplication, and matrix-vector multiplication to obtain (15).

C. Recovery of the Solutions in All Other Layers

Once the top surface unknowns in the ith layer are known from (15), the surface unknowns in other layers can be calculated as

$$\begin{aligned} x_{1,l} &= \mathbf{M}_l^{'-1} [b'_{1,l} - \mathbf{K}_l x_{3,l}], \quad l = i-1, i-2, \ldots, 1 \\ x_{3,l} &= \mathbf{M}_l^{'-1} [b'_{3,l} - \mathbf{K}_l x_{1,l}], \quad l = i, i+1, 2, \ldots, L-1 \\ x_{3,l} &= \mathbf{M}_l^{-1} [b_{3,l} - \mathbf{K}_l x_{1,l}], \quad l = L. \end{aligned} \tag{16}$$

Due to the linear dependence of $\mathbf{M}$ and $\mathbf{K}$, the cost of computing (16) is reduced to the evaluation of $\mathbf{M}_l^{'-1} b$ in each layer. Since $\mathbf{M}'_l(l = 1, 2, \ldots, L)$ are linearly proportional to each other, their inverse only needs to be conducted once. The dimension of $\mathbf{M}'_l$ is N_s.

The volume unknowns in layer i can be calculated as

$$x_{Vl} = \mathbf{P}_{Vl}^{-1} b_{Vl} \quad l = 1, 2, \ldots, L \tag{17}$$

where matrix $\mathbf{P}_{Vl}$ is assembled from their elemental contributions as

$$\mathbf{P}^e_{Vl,ij} = h_l (\mu_0 \varepsilon_l^e) \iint_{\Omega^e} \xi_i \cdot \xi_j d\Omega \quad l = 1, 2, \ldots, L. \tag{18}$$

Clearly, $\mathbf{P}_{Vl}$ in different layers are also linearly proportional to each other. Hence its inverse only needs to be done once irrespective of the number of layers. The dimension of $\mathbf{P}_{Vl}$ is N_V.

D. Performance Analysis

The numerical steps of the proposed method can be summarized as follows.

Step 1: Reduce the original matrix equation to a single-layer matrix equation as shown in (15);

Step 2: Pre-calculate the LU factorization of $\mathbf{M}$ in (12) and $\mathbf{P}_V$ in (18) for the time marching process;
Beginning of the Time Marching

Step 3: Solve (15) and obtain the solution of top surface unknowns in that single layer;

Step 4: Recover solutions of surface unknowns in all other layers using (16), and recover solutions of volume ones in all layers using (17);

Step 5: Construct the new right hand side of (6) for the next time step. Go back to step 3.
End of the Time Marching.

The computational cost of each step is as follows.

Step 1: Negligible since the reduction is done analytically;

TABLE I
PERFORMANCE ANALYSIS

	Cost of matrix factorization	Cost at each time step
Traditional TDFEM	Memory: $O(N^2)$ CPU: $O(N^3)$	Memory: $O(N^2)$ CPU: $O(N^2)$
LAFE-RR Method	Memory: $O(N^2)/(2L^2)$ CPU: $O(N^3)/(4L^3)$	Memory: $O(N^2)/(2L^2)$ CPU: $O(N^2)/(2L)$
Gain	Memory: $2L^2$ CPU: $4L^3$	Memory: $2L^2$ CPU: $2L$

Step 2: The cost of the LU factorization of two small matrices. The dimension of one matrix is N_s. The dimension of the other is N_V. Both sizes are approximately equal to $N/(2L)$;
Assuming that the LU factorization of a matrix of size N requires $O(N^3)$ operations and $O(N^2)$ storage. The time complexity of step 2 is $O((N/2L)^3 \times 2) = O(N^3)/(4L^3)$ with $O((N/2L)^2 \times 2) = O(N^2)/(2L^2)$ memory consumption. This is a worst-case estimation. Since the single-layer matrix is small and sparse, the factorization often can be performed in linear complexity.
Steps 3 and 4: The cost is L-time backward and forward substitutions of one sparse matrix of size N_s, and one sparse matrix of size N_V.
Assuming that the backward and forward substitutions scale as $O(N^2)$ in both CPU time and memory consumption. The time complexity of steps 3 and 4 is $O((N/2L)^2 \times 2) \times L = O(N^2)/(2L)$ with $O((N/2L)^2 \times 2) = O(N^2)/(2L^2)$ memory consumption. Again, this is the upper bound of the complexity;
Step 5: The cost of a sparse matrix-vector multiplication, which has a linear complexity.

The performance comparison with a traditional time-domain finite element method is given in Table I. Note that the performance gain listed in Table I applies to any arbitrarily shaped multilayer structure. No periodicity or special structure features are needed to obtain the listed performance gain. For on-chip circuits, x or y are often chosen as the layer growth direction to minimize the number of unknowns in a single layer. In this case, the number of layers encountered is an extremely large number. Hence, the performance gain of the proposed method is significant. More importantly, when the total number of unknowns is large, it is impossible to factorize the original matrix with existing computational resources. However, with the proposed method, as long as there is sufficient memory to factorize a single-layer sparse matrix, there exists sufficient memory to solve the entire system.

It should be noted that the comparison in Table I is given for large-scale problems in which N is large. When N is small, the traditional method can perform better than $O(N^3)$ in matrix factorization, and better than $O(N^2)$ in forward and backward substitution. However, the proposed method still offers significant gains in both memory and CPU time. For example, if the traditional method performs the matrix factorization in $O(N^2)$ computational complexity, the improvement of the proposed-method in the matrix factorization is $2L^2$; if the traditional method performs forward and backward substitution in $O(N^{1.5})$ complexity, the improvement of the proposed method at each time step is $(2L)^{0.5}$. In addition, in Table I, it is assumed that the complexity of factorizing a matrix of half-layer size $N/(2L)$ is the same as that of factorizing the original matrix of dimension N. In reality, the former is computationally much less intense than the latter due to the reduced size. Hence, the performance gain of the proposed method can be even better.

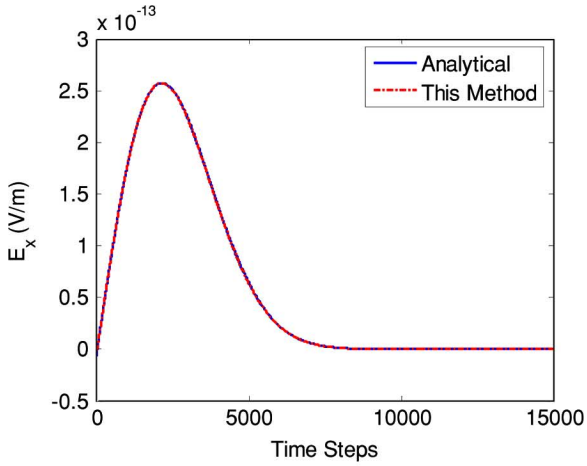
In recent years, the traditional time-domain finite-element method has been accelerated by domain-decomposition techniques. Assuming one subdomain is one layer, for a general layered structure, the domain decomposition technique factorizes the matrix in each subdomain, while the proposed method only needs to factorize a sparse matrix in a single subdomain regardless of the original problem size.

E. Stability Analysis

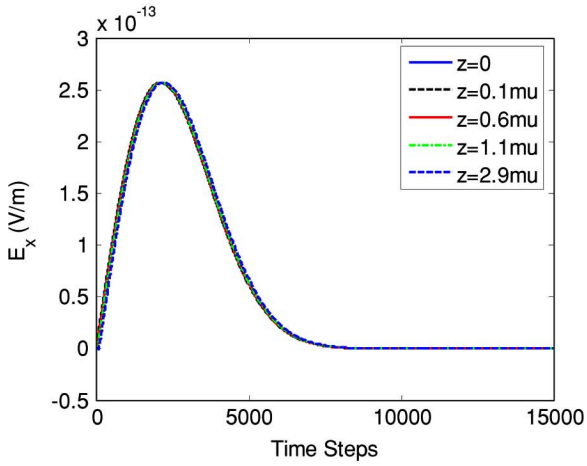
What is solved by the proposed time-domain LAFE-RR method is still (6). It is in essence a high-capacity and efficient direct solver of (6) by fully exploring the property of layered structures. Hence, its stability analysis follows that for a traditional time-domain finite element method as given by [23]. Although (6) is decomposed into (9), it is a natural and rigorous decomposition because the basis functions associated with volume unknowns and those associated with surface ones are perpendicular to each other. The time step used for (6) is used for (9). The unknowns solved from (9) at each time step are exactly the same as those solved from (6).

V. EXAMPLES

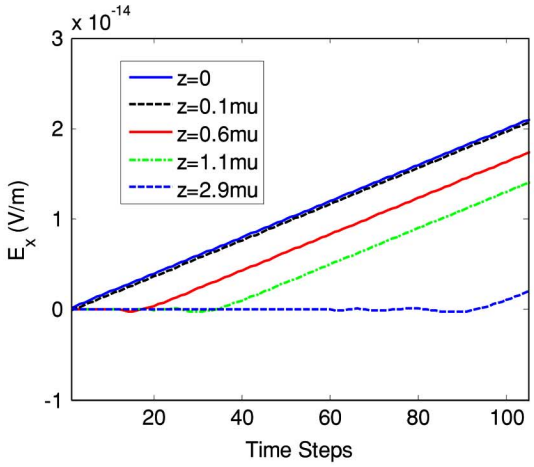
The proposed method is first validated with a structure having an analytical solution: a parallel-plate waveguide structure. According to typical geometrical dimensions of on-chip problems, the waveguide width (along y) was set as 1 μm; the waveguide height (along x) was set as 0.1 μm. The waveguide length (3.5 μm) was subdivided into 35 layers. The dominant TEM mode was launched on the incident plane



(a)



(b)



(c)

Fig. 8. Simulation of a parallel-plate waveguide in comparison with the analytical solution. (a) Electric field sampled at $z = 1.1\ \mu$m. (b) Electric fields sampled at $z = 0, 0.1\ \mu$m, $0.6\ \mu$m, $1.1\ \mu$m, and $2.9\ \mu$m. (c) A magnified plot showing time delay at $z = 0, 0.1\ \mu$m, $0.6\ \mu$m, $1.1\ \mu$m, and $2.9\ \mu$m.

at $z = 0$. The exact absorbing boundary condition for the dominant mode was placed on both the incident and exiting planes. The incident pulse was the time derivative of a Gaussian pulse, $\hat{E}^{\text{inc}}(t) = \hat{x}2t\exp(-t^2/\tau^2)$, in which τ was chosen to be 3.0e–13 s. The time step was chosen to be 1.0e–16 s. In

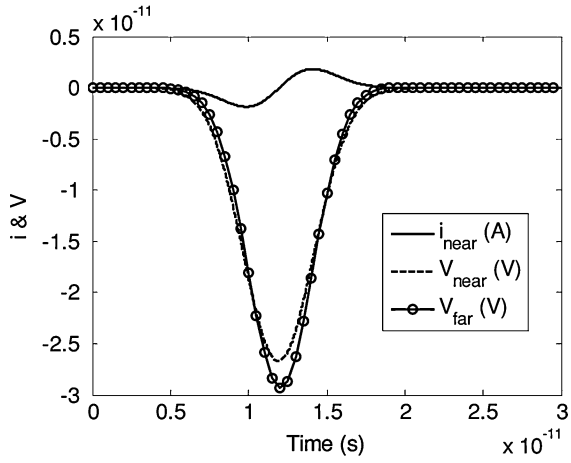


Fig. 9. Time-domain waveforms of a 100-μm long test-chip interconnect structure simulated by the proposed method. ($\mathrm{i_{near}}$ is the current source placed at the near end of an interconnect line; $\mathrm{V_{near}}$ and $\mathrm{V_{far}}$ are the voltages observed at the near end, and far end respectively.)

total 15 000 time steps were simulated. The electric field at $z = 1.1\ \mu$m was sampled and compared with the analytical solution, which revealed an excellent agreement, as shown in Fig. 8(a). Fig. 8 (b) and (c) depicts the electric fields sampled at $z = 0, 0.1\ \mu$m, $0.6\ \mu$m, $1.1\ \mu$m, and $2.9\ \mu$m respectively. The delay and waveform are accurately simulated by the proposed method.

With the accuracy validated, next, a test-chip interconnect structure of length 100 μm was simulated, and the performance of the proposed time-domain LAFE-RR method was compared against that of the traditional time-domain finite element method. The structure was fabricated using conventional silicon processing technology. It comprised three metal layers and thirteen inhomogeneous dielectric stacks. The structure was subdivided into 102 layers along its length. Each layer was divided into 676 triangular prism elements, rendering 1053 surface unknowns and 378 volume unknowns per layer. The total number of unknowns was 147015. A comparison of the performance between the proposed method and the standard TDFEM is given Table II.

As shown in Table II, the speedup in matrix factorization is more than four orders of magnitude. The speedup in the time-marching process is more than seven. The performance gain listed here is not as ideal as that listed in Table I because the multi-frontal solver [24] used in this example for the matrix solution performed better than $O(N^3)$ in matrix factorization, and better than $O(N^2)$ in the forward and backward substitution when handling a matrix of size 147015. For example, its performance is shown to be proportional to $N^{1.5}$ in the forward and backward substitution. The computing platform here is a PC with AMD Athlon 64 X2 Dual Core 4200+ processors running at 2.21 GHz with 2 GB RAM.

Fig. 9 depicts the time-domain waveforms of the sampled voltages at the near- and far-end of one wire in the interconnect structure with its near end excited by a current source and the far end left open. Clearly, as expected, RC effects dominate the behavior of this short interconnect structure because the sampled voltage behaves as an integration of the current

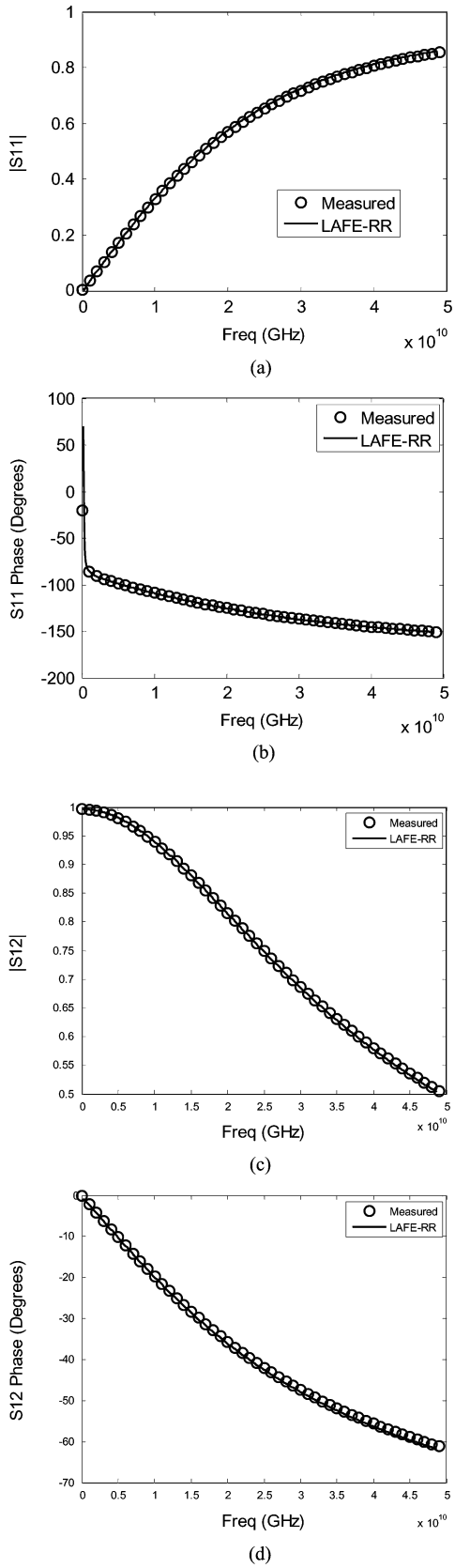


Fig. 10. Simulated S-parameters of a test-chip interconnect of 100 μm length. (a) $|S11|$. (b) S11 Phase. (c) $|S12|$. (d) S12 Phase.

over the time. Fig. 10 shows the frequency-domain S-parameters extracted from the time-domain results in comparison with the measured data. Excellent agreement was observed. In addition, the method has shown to be robust from high frequencies all the way down to DC.

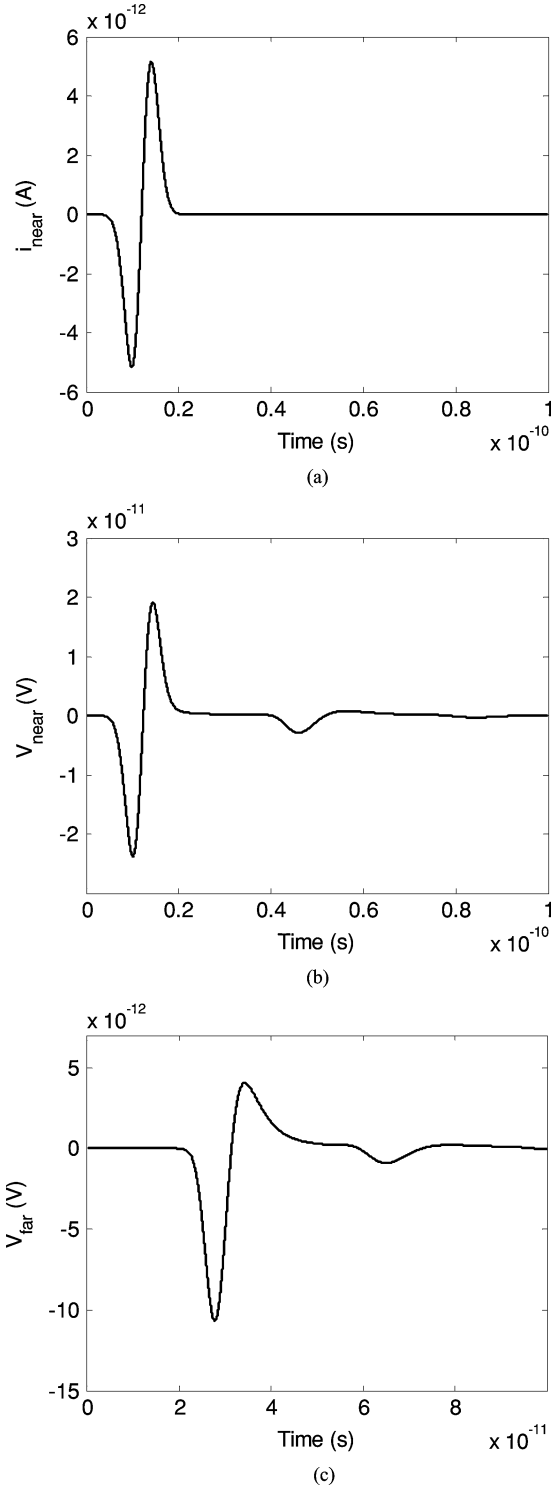


Fig. 11. Time-domain waveforms of a large-scale test-chip interconnect simulated by the proposed method. (a) Current source launched at the near end. (b) Voltage sampled at the near end. (c) Voltage sampled at the far end.

Finally a large-scale test-chip interconnect structure of length 2000 μm was simulated. Along the length of the interconnect, the structure was discretized into 2000 layers. Due to the large scale of the resultant matrix, the traditional TDFEM fails to

TABLE II
PERFORMANCE COMPARISON

	Standard TDFEM	This Method
Matrix factorization time	492.39 s (inverse of **P** in (6))	0.021 s
Cost at each time step	0.86 s	0.125 s

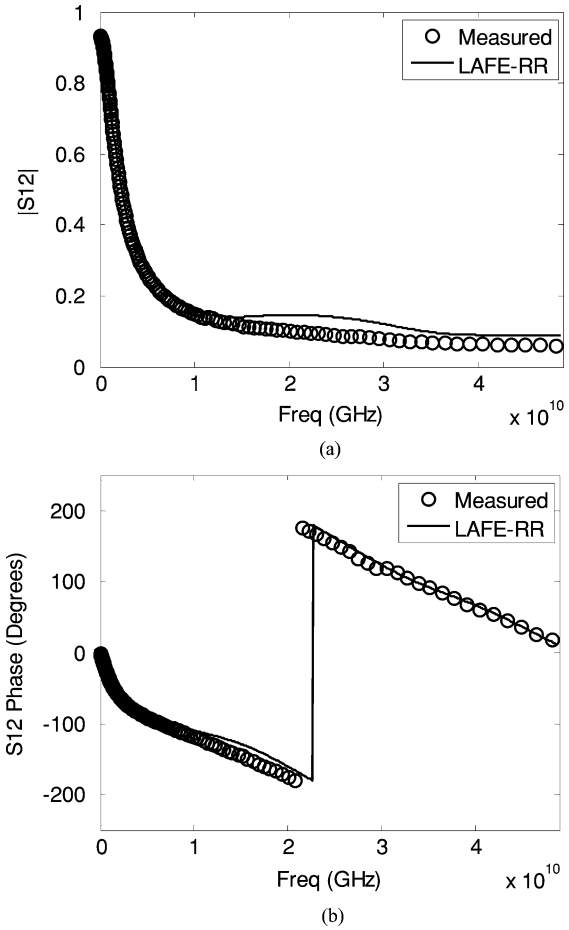


Fig. 12. Simulated Crosstalk of a large-scale test-chip interconnect. (a) S12 magnitude. (b) S12 Phase.

factorize the system matrix, whereas the proposed method successfully simulated the S-parameters of the test-chip structure. Fig. 11 depicts the time-domain waveforms sampled at the near and far end of an interconnect line, which demonstrate an inductance effect. Fig. 12 plots the simulated crosstalk in comparison with the measured data. Once again, the agreement is very good.

VI. CONCLUSION

The analysis and design of next generation high-speed ICs using the most accurate electromagnetics-based models results in numerical problems of ultra large scale, requiring billions of parameters to describe them accurately. In order to address the large problem size, electromagnetic solutions have to scale favorably. The proposed time-domain LAFE-RR method is capable of producing a smaller problem (by several orders of magnitude) to rigorously obtain the solution of the original large problem. In essence, it solves $\mathbf{A}x = b$ rigorously by solving $\mathbf{A}'x = b'$ with the dimension of $\mathbf{A}'$ orders of magnitude smaller than $\mathbf{A}$. Hence, large-scale problems can be made tractable using the proposed method with existing computational resources. Furthermore, due to the analytical reduction procedure, the CPU time is dramatically reduced. Numerical and experimental results have demonstrated the accuracy, efficiency, and capacity of the proposed method.

ACKNOWLEDGMENT

The authors would like to thank M. J. Kobrinsky and S. Chakravarty at Intel Corporation for providing measured data.

REFERENCES

[1] M. J. Kobrinsky, S. Chakravarty, D. Jiao, M. C. Harmes, S. List, and M. Mazumder, "Experimental validation of crosstalk simulations for On-Chip interconnects using S-parameters," *IEEE Trans. Adv. Packag.*, vol. 28, no. 1, pp. 57–62, Feb. 2005.

[2] W. C. Chew, J. M. Jin, E. Michielssen, and J. M. Song, Eds., *Fast and Efficient Algorithms in Computational Electromagnetics*. Norwood, MA: Artech House, 2001.

[3] D. Jiao, C. Dai, S.-W. Lee, T. R. Arabi, and G. Taylor, "Computational electromagnetics for high-frequency IC design," in *Proc. IEEE Int. Symp. on Antennas and Propagation*, 2004, pp. 3317–3320, invited paper.

[4] P. J. Restle, A. E. Ruehli, S. G. Walker, and G. Papadopoulos, "Full-wave PEEC time-domain method for the modeling of on-chip interconnects," *IEEE Trans. CAD*, vol. 20, no. 7, pp. 877–887, Jul. 2001.

[5] A. Rong, A. C. Cangellaris, and L. Dong, "Comprehensive broadband electromagnetic modeling of on-chip interconnects with a surface discretization-based generalized PEEC model," in *Proc. IEEE 12th Topical Meeting on Electrical Performance of Electronic Packaging (EPEP)*, 2003, pp. 367–370.

[6] Z. H. Zhu, B. Song, and J. K. White, "Algorithms in FastImp: A fast and wideband impedance extraction program for complicated 3-D geometries," *IEEE Trans. CAD*, vol. 24, no. 7, pp. 981–998, Jul. 2005.

[7] D. Jiao, M. Mazumder, S. Chakravarty, C. Dai, M. Kobrinsky, M. Harmes, and S. List, "A novel technique for full-wave modeling of large-scale three-dimensional high-speed on/off-chip interconnect structures," in *Proc. Int. Conf. on Simulation of Semiconductor Processes and Devices*, 2003, pp. 39–42.

[8] S. Kapur and D. E. Long, "Large-scale full-wave simulation," *DAC*, 2004.

[9] D. Gope, A. E. Ruehli, C. Yang, and V. Jandhyala, "(S)PEEC: time- and frequency-domain surface formulation for modeling conductors and dielectrics in combined circuit electromagnetic simulations," *IEEE Trans. on MTT*, vol. 54, no. 6, pp. 2453–2464, Jun. 2006.

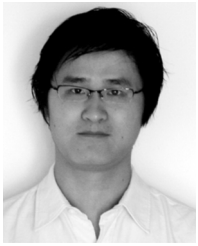
[10] Z. G. Qian, J. Xiong, L. Sun, I. T. Chiang, W. C. Chew, L. J. Jiang, and Y. H. Chu, "Crosstalk analysis by fast computational algorithms," in *Proc. IEEE 14th Topical Meeting on Electrical Performance of Electronic Packaging*, 2005, pp. 367–370.

[11] F. Ling, V. I. Okhamtovski, W. Harris, S. McCracken, and A. Dengi, "Large-scale broadband parasitic extraction for fast layout verification of 3-D RF and mixed-signal on-chip structures," *IEEE Trans. Microw. Theory Tech.*, vol. 53, no. 1, pp. 264–273, Jan. 2005.

[12] D. Jiao, S. Chakravarty, and C. Dai, "A layered finite-element method for high-capacity electromagnetic analysis of high-frequency ICs," *IEEE Trans. Antennas Propag.*, vol. 55, no. 2, pp. 422–432, Feb. 2007.

[13] Z. Y. Yuan, Z. F. Li, and M. L. Zou, "Computer-aided analysis of on-chip interconnects near semiconductor substrate for high-speed VLSI," *IEEE Trans. CAD*, vol. 19, no. 9, pp. 990–998, Sept. 2000.

[14] J. Lee, R. Lee, and A. Cangellaris, "Time-domain finite-element methods," *IEEE Trans. Antennas Propag.*, vol. 45, no. 3, pp. 430–442, Mar. 1997.
[15] S. Gedney and U. Navsariwala, "An unconditionally stable finite element time-domain solution of the vector wave equation," *IEEE MicrowGuided Wave Lett.*, vol. 5, pp. 332–334, Oct. 1995.
[16] D. Jiao and J. M. Jin, "Finite element analysis in time domain," in *The Finite Element Method in Electromagnetics*. New York: Wiley, 2002, pp. 529–584.
[17] H. Tsai, Y. Wang, and T. Itoh, "An unconditionally stable extended finite-element time-domain solution of active nonlinear microwave circuits using perfectly matched layers," *IEEE Trans. Microw. Theory Tech.*, vol. 50, no. 10, pp. 2226–2232, Oct. 2002.
[18] F. Edelvik, G. Ledfelt, P. Loststedt, and D. Riley, "An unconditionally stable subcell model for arbitrarily oriented thin wires in the FETD method," *IEEE Trans. Antennas Propag.*, vol. 51, no. 8, pp. 1797–1805, Aug. 2003.
[19] D. White and M. Stowell, "Full-wave simulation of electromagnetic coupling effects in RF and mixed-signal ICs using a time-domain finite-element method," *IEEE Trans. Microw. Theory Tech.*, vol. 52, no. 5, pp. 1404–1413, May 2004.
[20] D. Riley and J. Jin, "Modeling of magnetic loss in the finite-element time-domain method," *Microw. Opt. Tech. Lett.*, vol. 46, no. 2, pp. 165–168, Jul. 2005.
[21] Z. Lou and J. M. Jin, "A novel dual field time-domain finite-element domain-decomposition method for computational electromagnetics," *IEEE Trans. Antennas Propag.*, vol. 54, no. 6, pp. 1850–1862, Jun. 2006.
[22] R. D. Graglia, D. R. Wilton, A. F. Peterson, and I. Gheorma, "Higher order interpolatory vector bases on prism elements," *IEEE Trans. Antennas Propag.*, vol. 46, no. 3, pp. 442–450, Mar. 1998.
[23] D. Jiao and J. M. Jin, "A general approach for the stability analysis of time-domain finite element method," *IEEE Trans. Antennas Propag.*, vol. 50, no. 11, pp. 1624–1632, Nov. 2002.
[24] *UMFPACK*, [Online]. Available: http://www.cise.ufl.edu/research/sparse/umfpack/



Houle Gan received the B.S. and M.S. degree in information science and electronic engineering from Zhejiang University, Hangzhou, China, in 2003 and 2006, respectively. He is now working toward the Ph.D. degree at Purdue University, West Lafayette, IN.

In 2006, he was a System Engineer in Realsil Microelectronics Inc., Suzhou, China. In September 2006, he joined the On-Chip Electromagnetics Group, Purdue University, as a Research Assistant. His current research interest is computational electromagnetics for large-scale high-frequency integrated circuit design.

Dan Jiao (S'00-M'02-SM'06) received the Ph.D. degree in electrical engineering from the University of Illinois at Urbana-Champaign, in October 2001.

From October 2001 until September 2005, she worked in the Technology CAD Division at Intel Corporation as a Senior CAD Engineer, Staff Engineer, and Senior Staff Engineer. In September 2005, she joined Purdue University as an Assistant Professor in the School of Electrical and Computer Engineering. She has authored two book chapters and over 65 papers in refereed journals and international conferences. Her current research interests include high frequency digital, analogue, mixed-signal, and RF IC design and analysis, high-performance VLSI CAD, modeling of micro- and nano-scale circuits, computational electromagnetics, applied electromagnetics, fast and high-capacity numerical methods, fast time domain analysis, scattering and antenna analysis, RF, microwave, and millimeter wave circuits, wireless communication, and bio-electromagnetics.

Dr. Jiao received the 2006 Jack and Cathie Kozik Faculty Start-up Award, which recognizes an outstanding new faculty member in Purdue ECE. She received an ONR award through the Young Investigator Program in 2006. In 2004, she received the Best Paper Award from Intel's annual corporate-wide technology conference (Design and Test Technology Conference) for her work on generic broadband model of high-speed circuits. In 2003, she won the Intel Logic Technology Development (LTD) Divisional Achievement Award in recognition of her work on the industry-leading BroadSpice modeling/simulation capability for designing high-speed microprocessors, packages, and circuit boards. She was also awarded the Intel Technology CAD Divisional Achievement Award for the development of innovative full-wave solvers for high frequency IC design. In 2002, she was awarded by Intel Components Research the Intel Hero Award (Intel-wide she was the tenth recipient) for the timely and accurate two- and three- dimensional full-wave simulations. She also won the Intel LTD Team Quality Award for her outstanding contribution to the development of the measurement capability and simulation tools for high frequency on-chip cross-talk. She was the winner of the 2000 Raj Mittra Outstanding Research Award given her by the University of Illinois at Urbana-Champaign. She has served as the reviewer for many IEEE journals and conferences.