Efficient Extraction of Frequency-Dependent Substrate Parasitics Using Direct Boundary Element Method

Wenjian Yu, Xiren Wang, Zuochang Ye, and Zeyi Wang

***Abstract*—An efficient method based on a direct boundary element method is proposed for extracting frequency-dependent substrate coupling parameters. A frequency-independent real-valued linear equation system is first solved. Then, the solution is transformed into frequency-dependent parameters at a specified frequency with the Sherman–Morrison–Woodbury formula. The first step is performed only once for a given structure, and the method is very efficient for extraction with multiple frequencies. The proposed method is compared with the Green's function-based method and the approach in our earlier paper for typical substrate structures. Numerical results demonstrate its accuracy, efficiency, and versatility.**

***Index Terms*—Arbitrary substrate profile, direct boundary element method (DBEM), frequency-dependent parameter extraction, Sherman–Morrison–Woodbury formula.**

I. Introduction

Due to the lossy nature of widely used Si substrate, the problem of substrate noise coupling is increasingly important for the design of mixed-signal and RF circuits [1], [2]. At frequencies lower than several gigahertz, the substrate coupling is often modeled with a resistance network among contacts, considering the transportation of ohmic current. At higher frequencies, both ohmic and displacement currents should be considered, which results in a model with frequency-dependent resistance and capacitance [2].

Among the numerical methods for substrate extraction, the method of Green's function [1]–[3] and the direct boundary element method (DBEM) [4]–[6] are very important. The Green's function-based methods discretize the contact surfaces, therefore employing fewer unknown variables. However, for the multilayer structure, the Green's function is composed of several nested infinite series, which converge very slowly. The techniques based on eigen-decomposition and discrete cosine transform (DCT) have been proposed to calculate the Green's function for acceleration [1], [2]. It should be pointed out that these Green's function-based methods are limited to the multilayer structure. For more complex structures such as those containing lateral resistivity variations, the corresponding Green's function can hardly be deduced [5].

The DBEM discretizes the boundary of a homogeneous region and converts the Laplace equation to the boundary integral equation (BIE) employing the free-space Green's function [7]. DBEM is very suitable for solving a three-dimensional (3-D) electrostatic problem within finite domain [7], [8]. Because the BIE is derived for each homogeneous region and the BIEs for different regions are coupled through the interface compatibility equations, the DBEM produces a blocked sparse linear equation system for multiregion problems. A quasi-multiple medium (QMM) technique was proposed to decompose a physical medium into fictitious medium regions, to produce a sparser linear system corresponding to the problem after medium decomposition [7]. The DBEM was recently applied to substrate resistance extraction [5]. With a matrix reduction and QMM techniques applied, the DBEM outperforms the Green's function-based methods in [1] and [3] with speedup of several tens. More importantly, the DBEM has no demand of the layered-geometry substrate [5]. It is able to handle realistic substrates with noise reduction components, where layout-dependent doping profile makes lateral resistivity variation.

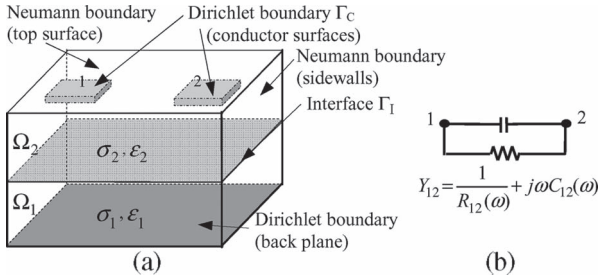

Fig. 1. (a) Typical structure for frequency-dependent substrate extraction. (b) Equivalent coupling admittance between contacts 1 and 2.

In this paper, the DBEM is applied to calculate the frequency-dependent substrate parameters, considering both capacitive and resistive effects. In order to extract the substrate parameters at multiple frequencies, a two-step approach is proposed to separate the frequency-independent computation and the computation at each frequency, which avoids building and solving the whole discretized DBEM equations repeatedly. As the first step, a real-valued linear equation system corresponding to the substrate resistance extraction [5] is solved. Then, its solution and the Sherman–Morrison–Woodbury formula are utilized to efficiently extract the frequency-dependent parameters. The proposed method is compared with the ASITIC program employing the Green's function-based method [2], [9]. Numerical results demonstrate that the proposed method is highly efficient and preserves good accuracy.

II. Background

A. Problem Formulation

Strictly speaking, both electric and magnetic effects should be considered for substrate coupling, but the latter is usually negligible because the lightly doped Si substrate (with much larger resistivity than metal) is widely used [11]. In this paper, we ignore the magnetic effect. Fig. 1(a) shows a 3-D substrate structure, including two mediums Ω_1 and Ω_2. Each medium has its own conductivity σ and permittivity ε. The coupling effects between contacts 1 and 2 in Fig. 1(a) can be represented by the equivalent model in Fig. 1(b). In general, a substrate includes different mediums. Therefore, the equivalent capacitance and resistance in Fig. 1(b) are both frequency dependent [2].

Similar to that shown in Fig. 1, the substrate coupling in a multicontact structure can be modeled with the admittance matrix $Y(\omega)$, where ω is the angular frequency. The real part of Y parameter stands for resistive effect, whereas the imaginary part stands for capacitive effect. Following the definition of admittance, we calculate the Y parameters by setting a sinusoidal voltage of 1 V on one contact (for example,

Manuscript received August 30, 2007; revised March 20, 2008. This work was supported in part by NSFC under Grants 60401010 and 60720106003. This paper was presented in part at the Proceedings of the 12th Asia and South Pacific Design Automation Conference (ASP-DAC 2007), pp. 62–67. This paper was recommended by Associate Editor R. Suaya.

W. Yu and Z. Wang are with the Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China (e-mail: yu-wj@tsinghua.edu.cn).

X. Wang was with the Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China. He is now with the Department of Electrical Engineering, University of Washington, Seattle, WA 98195 USA.

Z. Ye was with the Institute of Microelectronics, Tsinghua University, Beijing 100084, China. He is now with Cadence Berkeley Labs, Berkeley, CA 94704 USA.

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/TCAD.2008.925787

contact i) and 0 V on others. Then, the current flowing through contact k is equal to the value of Y_{ik}. Once a Y parameter is calculated, the corresponding resistance and capacitance parameters can be obtained with the following:

$$R(\omega) = 1/\text{real}(Y) \qquad C(\omega) = \text{imag}(Y)/j\omega. \tag{1}$$

For the two-medium structure shown in Fig. 1(a), the admittance between contacts i and k can be calculated with the following:

$$Y_{ik} = \int_{\Gamma_{C_k}} (\sigma_2 + j\omega\varepsilon_2) q \, d\Gamma \tag{2}$$

where Γ_{C_k} is the surface of contact k, and q is the normal electric field intensity on the contact boundary. Note that the value of q is usually frequency dependent. Equation (2) considers both ohmic and displacement currents, and then, the problem becomes how to calculate the value of q on the boundary.

The conductors in Fig. 1(a) are often approximated as planar contacts with zero thickness, because conductor thickness may be much smaller than its lateral dimensions. The model of 2-D contact or a 3-D conductor makes little difference for the DBEM employed in this paper because it only affects the boundary conditions of simulated region. Realistic substrate structures may be more complicated than that shown in Fig. 1(a) with nonstratified medium topology or dielectric material (whose σ is zero) included. DBEM can easily handle these complicated structures, as revealed in [5] and [8].

B. Direct Boundary Element Method

Because only the quasi-state electric and steady current fields are considered, the electric potential u fulfills the Laplace equation in each homogenous medium. With the DBEM, a linear equation system is formed for each medium region [7]

$$\boldsymbol{H}^{(i)} \cdot \boldsymbol{u}^{(i)} = \boldsymbol{G}^{(i)} \cdot \boldsymbol{q}^{(i)}, \qquad \text{for region } \Omega_i \tag{3}$$

where vectors $\boldsymbol{u}^{(i)}$ and $\boldsymbol{q}^{(i)}$ stand for the discretized u and q unknowns on the ith medium's boundary, respectively.

On the interface of two adjacent mediums a and b, the u and q unknowns fulfill the compatibility equations

$$\begin{cases} (\sigma_a + j\omega\varepsilon_a) q_a = -(\sigma_b + j\omega\varepsilon_b) q_b, & \text{on interface } \Gamma_I \\ u_a = u_b. \end{cases} \tag{4}$$

The matrix equations (3) for all regions can be combined together with (4). Using the boundary conditions [see Fig. 1(a)] and organizing the equations with the approach proposed in [7], a linear equation system with multiple right-hand side (RHS) vectors is obtained for extracting the admittance matrix [5]

$$\boldsymbol{A}X = \boldsymbol{B}. \tag{5}$$

Here, $\boldsymbol{B}$ consists of different RHS vectors for the specified voltage settings, and the discretized unknowns of u and q form $\boldsymbol{X}$.

Because (4) includes complex-valued linear equations, the overall equation (5) is a complex-valued system. Moreover, the aforementioned formulation also suits the problem of resistance or capacitance extractions. If frequency is zero, the $j\omega\varepsilon$ items disappear in (2) and (4), and it is for resistance extraction [5]; if the σ items are dropped, it is for capacitance extraction. In both cases, a real-valued linear system is solved.

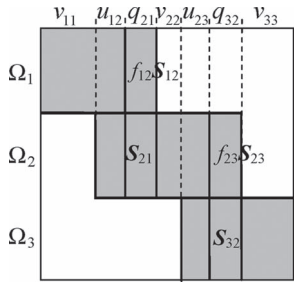

Fig. 2. Nonzero-entry distribution of matrix $\boldsymbol{A}$, where blocks $f_{12}\boldsymbol{S}_{12}$ and $f_{23}\boldsymbol{S}_{23}$ are frequency dependent.

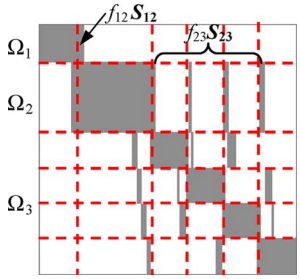

Fig. 3. With 2×2 QMM cutting applied for the top medium (Ω_3), matrix $\boldsymbol{A}$ becomes a 6×6 blocked sparse matrix.

Because the substrate parameters are frequency dependent, it is usually necessary to extract them at multiple frequencies for comprehensive knowledge of substrate coupling. A trivial approach for such multi-frequency extraction is to build matrix $\boldsymbol{A}$ for different frequencies and solve (5) repeatedly [4]. This method is not efficient if there are a lot of frequency points. In the next section, we propose a two-step approach, which is computationally preferable to the trivially repeated approach in [4].

III. Two-Step Approach

The distribution of frequency-dependent entries in matrix $\boldsymbol{A}$ is first investigated. Then, a two-step approach based on equation perturbation is proposed, which is followed by efficient techniques solving the original and perturbed equation systems.

A. Frequency-Dependent Entries in Matrix $\boldsymbol{A}$

Fig. 2 shows the distribution of nonzero entries in matrix $\boldsymbol{A}$ for a three-medium structure. If the top-layer medium containing contacts is cut into 2×2 fictitious regions with the QMM technique [5], matrix $\boldsymbol{A}$ becomes that shown in Fig. 3.

In Fig. 2, the matrix $\boldsymbol{A}$ is of a blocked sparse pattern, and the types of unknowns are given beside the corresponding matrix columns (for the definition of unknown type, please refer to [7]). The unknowns of types q_{21} and q_{32} are q unknowns on two medium interfaces. Suppose that the interface unknowns q_{21} are the actual q unknowns in (3) for medium Ω_2, whereas the interface q unknowns in (3) for medium Ω_1 are represented by q_{21} with (4). Thus, the matrix block $\boldsymbol{S}_{21}$ has real-valued entries calculated through panel integral, and the block directly above $\boldsymbol{S}_{21}$ can be expressed as $f_{12}\boldsymbol{S}_{12}$, where $f_{12} = -(\sigma_2 + j\omega\varepsilon_2)/(\sigma_1 + j\omega\varepsilon_1)$ and $\boldsymbol{S}_{12}$ consists of panel integrals. Similar things happen for the interface between Ω_2 and Ω_3; the coefficients of q_{32} are a real-valued matrix block $\boldsymbol{S}_{32}$ and a scalar-matrix product $f_{23}\boldsymbol{S}_{23}$.

Because only (4) introduces ω to the formulation of DBEM, all nonzero entries in matrix $\boldsymbol{A}$ except those in blocks $f_{12}\boldsymbol{S}_{12}$ and

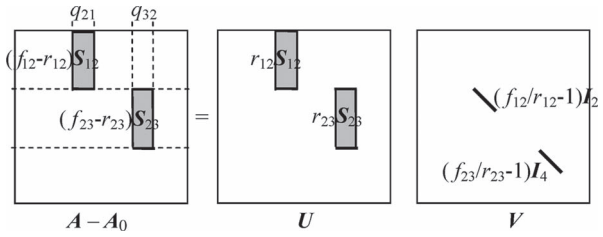

Fig. 4. Matrix $\boldsymbol{A}$ in Fig. 2 is decomposed into matrix $\boldsymbol{A}_0$ and the product of matrices $\boldsymbol{U}$ and $\boldsymbol{V}$. $\boldsymbol{I}_2$ and $\boldsymbol{I}_4$ in matrix $\boldsymbol{V}$ are two identity matrices.

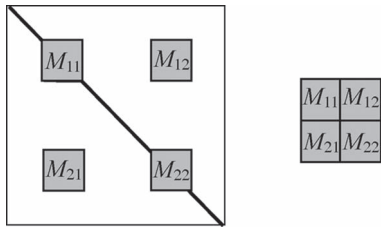

Fig. 5. Matrix $\boldsymbol{I} + \boldsymbol{V}A_0^{-1}\boldsymbol{U}$ and its compact form $\boldsymbol{M}$ for the example in Fig. 2.

$f_{23}\boldsymbol{S}_{23}$ are frequency independent and of real values. The frequency dependence is only caused by the scalars f_{12} and f_{23}.

The QMM technique hardly affects the aforementioned observation. Because the value of $\sigma + j\omega\varepsilon$ for a fictitious medium region is the same as the original medium it belongs to, the additional interface brought by QMM cutting does not introduce any frequency-dependent or complex-valued entry to matrix $\boldsymbol{A}$. Comparing Figs. 2 and 3, we find out that the QMM cutting only makes the frequency-dependent block $f_{23}\boldsymbol{S}_{23}$ dispersed horizontally to be several strips. The matrix reduction technique proposed in [5] does not affect the aforementioned observation either, because it discards the u unknowns on top surface and condenses the equations for the top medium Ω_3. Therefore, we can introduce the two-step approach according to the simple matrix structure shown in Fig. 2 without loss of generality.

B. Perturbed Equation System and Its Efficient Solution

Equation (5) can be regarded as a perturbed equation system with coefficient matrix made from a frequency-independent real-valued matrix $\boldsymbol{A}_0$. For the matrix $\boldsymbol{A}$ in Fig. 2, $\boldsymbol{A}_0$ is defined as

$$\boldsymbol{A}_0 \equiv \boldsymbol{A}, \qquad \text{if } f_{12} = r_{12} \quad \text{and} \quad f_{23} = r_{23} \tag{6}$$

where r_{12} and r_{23} are of real values (for example, -1). Then

$$\boldsymbol{A} = \boldsymbol{A}_0 + \boldsymbol{U}V \tag{7}$$

where $\boldsymbol{U}$ is a sparse matrix with nonzero entries, being a subset of those of $\boldsymbol{A}_0$, and $\boldsymbol{V}$ is a sparse diagonal matrix (see Fig. 4). Note that matrix $\boldsymbol{U}$ is frequency independent. Moreover, we call the following as the original equation system:

$$\boldsymbol{A}_0\boldsymbol{X}_0 = \boldsymbol{B}. \tag{8}$$

In the following, we consider the relationship between the solutions of (5) and (8).

The Sherman–Morrison–Woodbury formula claims [12] the following:

$$(\boldsymbol{A}_0 + \boldsymbol{U}V^{\mathrm{T}})^{-1} = \boldsymbol{A}_0^{-1} - \boldsymbol{A}_0^{-1}\boldsymbol{U}(\boldsymbol{I} + \boldsymbol{V}^{\mathrm{T}}\boldsymbol{A}_0^{-1}\boldsymbol{U})^{-1}\boldsymbol{V}^{\mathrm{T}}\boldsymbol{A}_0^{-1} \tag{9}$$

provided that $\boldsymbol{A}_0$ and $(\boldsymbol{I} + \boldsymbol{V}^T\boldsymbol{A}_0^{-1}\boldsymbol{U})$ are invertible. Because $\boldsymbol{A} = \boldsymbol{A}_0 + \boldsymbol{U}V$ and $\boldsymbol{V}$ is a diagonal matrix, we have the following:

$$\begin{aligned}\boldsymbol{X} &= \boldsymbol{A}^{-1}\boldsymbol{B} \\ &= \boldsymbol{A}_0^{-1}\boldsymbol{B} - \left(\boldsymbol{A}_0^{-1}\boldsymbol{U}\right)\left(\boldsymbol{I} + \boldsymbol{V}A_0^{-1}\boldsymbol{U}\right)^{-1}\boldsymbol{V}\left(\boldsymbol{A}_0^{-1}\boldsymbol{B}\right).\end{aligned} \tag{10}$$

Therefore, the solution of the perturbed equation (5) can be derived by basing on the solution of the original system (8). In other words, the frequency-dependent parameters can be obtained by basing on the frequency-independent solution $\boldsymbol{X}_0$. Because both $\boldsymbol{X}_0$ and $\boldsymbol{A}_0^{-1}\boldsymbol{U}$ are frequency independent, they can be calculated in advance and reused for different frequencies.

With (10), the main task for solving (5) becomes computing $(\boldsymbol{I} + \boldsymbol{V}A_0^{-1}\boldsymbol{U})^{-1}\boldsymbol{V}X_0$. The complex-valued matrix $\boldsymbol{I} + \boldsymbol{V}A_0^{-1}\boldsymbol{U}$ has the same dimension as $\boldsymbol{A}$, but it has a special sparse pattern. In $\boldsymbol{A} - \boldsymbol{A}_0$, the possible nonzero entries are at the positions corresponding to the q unknowns of medium interface (see Fig. 4). For a simulated structure including N_I "physical" interfaces, there are exactly N_I nonzero blocks in $\boldsymbol{A} - \boldsymbol{A}_0$ if all r parameters in (6) are set to be -1. Here, "physical" interface denotes the one whose related two mediums have different values of σ or ε. For a "nonphysical" interface whose related two mediums have the same σ and ε (for example, the fictitious interface induced by QMM), the corresponding f parameter in Fig. 2 will be constant -1, which does not contribute any nonzero entry to matrix $\boldsymbol{A} - \boldsymbol{A}_0$. Suppose that the nonzero blocks in $\boldsymbol{A} - \boldsymbol{A}_0$ occupy m columns, where m is the number of elements on the N_I "physical" interfaces. Because matrix $\boldsymbol{U}$ has the same sparse pattern as $\boldsymbol{A} - \boldsymbol{A}_0$, $\boldsymbol{A}_0^{-1}\boldsymbol{U}$ has exactly m nonzero columns. As shown in Fig. 4, the row numbers of nonzero diagonal entries in $\boldsymbol{V}$ is the same as the column numbers of nonzero entries in $\boldsymbol{A}_0^{-1}\boldsymbol{U}$. Therefore, $\boldsymbol{V}A_0^{-1}\boldsymbol{U}$ is a sparse matrix with $m \times m$ nonzero entries in $N_I \times N_I$ blocks. Fig. 5 shows the matrix $\boldsymbol{I} + \boldsymbol{V}A_0^{-1}U$ for the three-medium example. $\boldsymbol{M}_{ik}(1 \le i, k \le N_I)$ denotes the nonzero blocks in $\boldsymbol{I} + \boldsymbol{V}A_0^{-1}\boldsymbol{U}$ and forms a compact matrix $\boldsymbol{M} = [\boldsymbol{M}_{ik}]_{N_I \times N_I}$.

Theorem 1: Computing the $(\boldsymbol{I} + \boldsymbol{V}A_0^{-1}\boldsymbol{U})^{-1}$ is equal to inversing the $m \times m$ matrix $\boldsymbol{M}$.

Proof: Let $\boldsymbol{W} = \boldsymbol{M}^{-1}$. $\boldsymbol{W}$ can be expressed as the blocked form like $\boldsymbol{M}$: $\boldsymbol{W} = [\boldsymbol{W}_{ik}]_{N_I \times N_I}$. Then, filling the $\boldsymbol{W}_{ik}(1 \le i, k \le N_I)$ into the identity matrix $\boldsymbol{I}$ according to the pattern of $\boldsymbol{I} + \boldsymbol{V}A_0^{-1}\boldsymbol{U}$ produces $(\boldsymbol{I} + \boldsymbol{V}A_0^{-1}\boldsymbol{U})^{-1}$. This can be easily verified with the example in Fig. 5 ($\boldsymbol{I}_1$, $\boldsymbol{I}_3$, and $\boldsymbol{I}_5$ denote the three identity matrices)

$$\begin{bmatrix} \boldsymbol{I}_1 & & & \\ & \boldsymbol{M}_{11} & & \boldsymbol{M}_{12} \\ & & \boldsymbol{I}_3 & \\ & \boldsymbol{M}_{21} & & \boldsymbol{M}_{22} \\ & & & & \boldsymbol{I}_5 \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{I}_1 & & & \\ & \boldsymbol{W}_{11} & & \boldsymbol{W}_{12} \\ & & \boldsymbol{I}_3 & \\ & \boldsymbol{W}_{21} & & \boldsymbol{W}_{22} \\ & & & & \boldsymbol{I}_5 \end{bmatrix} = \boldsymbol{I}. \tag{11}$$

For a general case, $(\boldsymbol{I} + \boldsymbol{V}A_0^{-1}\boldsymbol{U})^{-1}$ can be constructed similarly, which involves inversing the matrix $\boldsymbol{M}$. ■

Notice that what we really want is $(\boldsymbol{I} + \boldsymbol{V}A_0^{-1}\boldsymbol{U})^{-1}\boldsymbol{V}X_0$. If the number of conductors is N_c, the dimension of $\boldsymbol{X}_0$ in (5) is $N \times N_c$. We shall first calculate $\boldsymbol{V}X_0$, which can be easily accomplished because $\boldsymbol{V}$ is a diagonal matrix. Then, these N_c RHS vectors are used while solving a linear equation with coefficient matrix $\boldsymbol{I} + \boldsymbol{V}A_0^{-1}\boldsymbol{U}$. Similar to the proof of Theorem 1, we actually solve the equation with coefficient matrix $\boldsymbol{M}$ for N_c RHS vectors to get $(\boldsymbol{I} + \boldsymbol{V}A_0^{-1}\boldsymbol{U})^{-1}\boldsymbol{V}X_0$. This result is then multiplied by $\boldsymbol{A}_0^{-1}\boldsymbol{U}$ (an $N \times N$ matrix with m nonzero columns) to get $\boldsymbol{X}$.

C. Efficient Technique for Solving the Original System

Because the RHS vectors $\boldsymbol{B}$ in (5) do not include complex numbers, (8) is a real-valued linear equation system. It can be regarded as an imaginary problem of resistance extraction, which keeps geometry information of the original problem, but with all mediums replaced by normal conductors. Both $\boldsymbol{X}_0$ and $\boldsymbol{A}_0^{-1}\boldsymbol{U}$ are needed in (10),

which form a problem with a lot of RHS vectors $\boldsymbol{P} = [\boldsymbol{B}\ \boldsymbol{U}]$. We propose to use the blocked Gauss method instead of the preconditioned generalized minimal residual (GMRES) algorithm in [5] for solving the original linear equation system.

For brief introduction, we consider a 2×2 blocked equation

$$\begin{bmatrix} \boldsymbol{D}_{11} & \boldsymbol{D}_{12} \\ \boldsymbol{D}_{21} & \boldsymbol{D}_{22} \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_2 \end{bmatrix} = \begin{bmatrix} \boldsymbol{b}_1 \\ \boldsymbol{b}_2 \end{bmatrix}. \tag{12}$$

After making a block substitution in (12), we get the following:

$$\begin{cases} \left(\boldsymbol{D}_{22} - \boldsymbol{D}_{21}\boldsymbol{D}_{11}^{-1}\boldsymbol{D}_{12}\right)\boldsymbol{x}_2 = \boldsymbol{b}_2 - \boldsymbol{D}_{21}\boldsymbol{D}_{11}^{-1}\boldsymbol{b}_1 \\ \boldsymbol{D}_{11}\boldsymbol{x}_1 = \boldsymbol{b}_1 - \boldsymbol{D}_{12}\boldsymbol{x}_2. \end{cases} \tag{13}$$

With (12), $\boldsymbol{x}_2$ and $\boldsymbol{x}_1$ can be solved in sequence. In this procedure, smaller linear system with coefficient matrices $\boldsymbol{D}_{11}$ and $\boldsymbol{D}_{22} - \boldsymbol{D}_{21}\boldsymbol{D}_{11}^{-1}\boldsymbol{D}_{12}$ (the update of $\boldsymbol{D}_{22}$) is solved. For a general $N_b \times N_b$ blocked equation, a similar deduction can be applied. Note that $\boldsymbol{D}_{11}^{-1}$ in (13) is purely symbolic. The matrix inverse need never be computed, and $\boldsymbol{D}_{11}^{-1}c$ is calculated by solving equation $\boldsymbol{D}_{11}\boldsymbol{x} = \boldsymbol{c}$, where $\boldsymbol{c}$ is either a column vector or a matrix including multiple columns.

Compared with the original Gauss elimination, the blocked Gauss method is advantageous in that LU factorization that is not done for the whole matrix but for each smaller diagonal block. If many matrix blocks in the coefficient matrix are zero, or include a few of nonzero columns, the blocked Gauss method will greatly improve the computational efficiency. At the same time, for a linear system with multiple RHS vectors, the blocked Gauss method would perform better than the iterative equation solver, because the LU factorization of diagonal blocks is done only once. On the contrary, the procedure of searching solution in an iterative equation solver repeats for each RHS. Thus, the more RHS vectors the problem contains, the more advantages the blocked Gauss method will show.

With the QMM technique and nonuniform element partition presented in [5], matrix $\boldsymbol{A}_0$ becomes a blocked sparse matrix with small diagonal blocks. Thus, the blocked Gauss method has high efficiency for solving the original linear system.

D. Algorithm Flow and Discussion

We summarize the algorithm flow of the two-step approach for frequency-dependent substrate extraction.

The first step:

1. Make QMM cutting, and form the discretized DBEM equations for the imaginary structure (all r parameters in (6) are set to -1).
2. Condense the linear equation system with the matrix reduction technique [5]. Now, the equation becomes (8).
3. Generate matrix $\boldsymbol{U}$ from matrix $\boldsymbol{A}_0$, and solve for $\boldsymbol{X}_0 = \boldsymbol{A}_0^{-1}\boldsymbol{B}$ as well as $\boldsymbol{X}_U = \boldsymbol{A}_0^{-1}\boldsymbol{U}$ with the blocked Gauss method.

The second step:

4. For each frequency, extract the Y parameters:
   (a) Form the sparse diagonal matrix $\boldsymbol{V}$, and calculate $\boldsymbol{V}X_0$;
   (b) Generate the smaller matrix $\boldsymbol{M}$ from $\boldsymbol{I} + \boldsymbol{V}X_U$;
   (c) Solve the equation with coefficient matrix $\boldsymbol{M}$ and the right-hand sides consisting of some rows in $\boldsymbol{V}X_0$;
   (d) Organize the solution to get $(\boldsymbol{I} + \boldsymbol{V}X_U)^{-1}\boldsymbol{V}X_0$, according to the proof of Theorem 1;
   (e) Calculate $\boldsymbol{X} = \boldsymbol{X}_0 - \boldsymbol{X}_U \cdot (\boldsymbol{I} + \boldsymbol{V}X_U)^{-1}\boldsymbol{V}X_0$, and the $\boldsymbol{Y}$ matrix.

The first step runs only once for a given structure, whose major work is to solve a real-valued linear system with an $N \times N$ coefficient matrix and $N_c + m$ RHS vectors. Because the nonzero entries of $\boldsymbol{X}_U$ are in m columns, the extra memory usage is of $O(N(N_c + m))$. The item 2 in the first step is optional, and it only performed for the substrate model with planar contacts. In the second step, an $m \times m$ complex-valued linear system with N_c RHS vectors is solved for each frequency. Because m (the number of elements on "physical" interfaces) is usually small, the frequency-dependent equation can be solved with LU factorization. Based on the exact Sherman–Morrison–Woodbury formula, the two-step approach does not degrade the accuracy.

The two-step approach has more applications. Because only the f parameters in the DBEM matrix involve the physical properties of mediums (see Fig. 2), the stored $\boldsymbol{X}_0$ and $\boldsymbol{X}_U$ can also be utilized for the extraction problem, where only σ or ε parameters vary. This makes quick extraction for structures with only frequency and/or medium parameters changed.

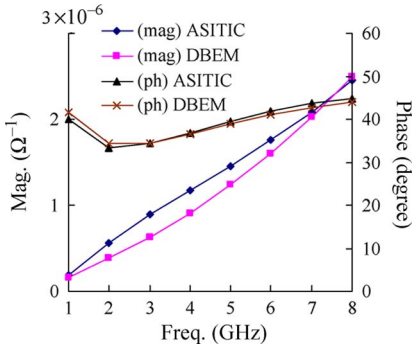

Fig. 6. Magnitude and phase of coupling admittance between contacts 26 and 52.

IV. Numerical Results

The proposed method is implemented as subRCbem, which is a program written in C++. We first use a typical structure with 52 contacts to demonstrate the efficiency of subRCbem. Then, other experimental results are briefly introduced for further validation. All experiments are run on a Sun Fire V880 server with a CPU of 750 MHz, except for that indicated explicitly.

A. Fifty-Two-Contact Structure

The 52-contact structure is described in [1] and [5], where the layout size is 128 μm $\times$ 128 μm (see [5, Fig. 7]). The layout is combined with low-resistivity (LR) and high-resistivity (HR) substrate profiles (see [2, Figs. 3–4]). The extraction results of subRCbem are compared with those of ASITIC [10]. In ASITIC, considering the Neumann boundary condition at the magnetic walls, the Green's function is expanded in cosine series. Thus, the fast Fourier transform can be applied to accelerate the summation of cosine series [2]. The method implemented in ASITIC is described in [2] and [9], and it is referred to as the DCT-accelerated Green's function method.

For the 52-contact substrate with HR profile, the whole admittance (Y) matrix is extracted at frequencies from 1 to 8 GHz. To show an example of frequency dependence, the coupling admittances between (two close contacts at the top right corner) contacts 26 and 52 are shown in Fig. 6. Both results obtained with ASITIC and subRCbem show the right trends of magnitude and phase. From the figure, we can see that the discrepancy between both results is within 4% for phase, whereas the discrepancy for magnitude is mostly within 15%. For other Y parameters, both results also have little discrepancy and show the same trend of frequency dependence.

TABLE I
RELEVANT PARAMETERS FOR CALCULATING THE EXAMPLE WITH HR PROFILE

Program	subRCbem	ASITIC		
# Contacts	52	52		
# Panels	7252	483	860	4352*
Memory (MB)	60	24	32	310
PreCal Time (s)	421	556	571	656
Time (s)	9.0	1930	9000	--

* Parameters corresponding to the default mesh setting in ASITIC.

The relevant computational parameters of both programs for the HR case are shown in Table I. "PreCal Time" means the CPU time spent on the first step of subRCbem or the preprocess for generating the Green's function in ASITIC. The "Time" in the tables means the CPU time of the frequency-dependent computation for a single frequency point. For the 52-contact structure, subRCbem takes only 9.0 s to extract the admittance matrix at each frequency once the first step was performed. On the contrary, the computational time of ASITIC with its default mesh is too long to be obtained. We then set coarser meshes in ASITIC and run it again. Even with only 483 panels, ASITIC takes 1930 s to accomplish a single-frequency extraction, which is 214 times more than that used by subRCbem. The "PreCal Time" and the memory usage of subRCbem are less than or comparable with ASITIC, as shown in Table I. For the structure with LR profile, the comparison of ASITIC and subRCbem is very similar.

The high efficiency of subRCbem is because only a small linear equation system is solved for each frequency (whose order is several hundreds). With the techniques of QMM, which are the matrix reduction and blocked Gauss solver, the preprocess stage in subRCbem also runs very fast, even though the number of unknowns is in several thousands. On the contrary, ASITIC needs to build and solve (for each frequency) a complex-valued linear system with coefficients generated by calculating the computationally expensive Green's function.

To demonstrate the advantage of the two-step approach for multi-frequency extraction, we compare subRCbem with the trivially repeated BEM method [4]. The trivially repeated BEM is also based on DBEM, and it employs the technique of QMM and matrix reduction. For each frequency, it reuses the frequency-independent entries of matrix $\boldsymbol{A}$ but solves the whole linear system (5) with a preconditioned GMRES solver [4]. The 52-contact structure is extracted for 20 frequency points, and the accumulated CPU time is shown in Fig. 7. This experiment is carried out on a Linux sever with 3-GHz CPU.

Fig. 7 shows that the CPU time of the trivially repeated BEM increases dramatically along with the increase of frequency points, because the reuse in generating (5) contributes little to computation acceleration. It is interesting that the time of subRCbem for 20 frequencies is even not larger than that of the trivially repeated BEM for the first frequency. One reason is that the real-valued linear system with multiple RHS in subRCbem is solved efficiently with the blocked Gauss method, whereas the trivially repeated BEM employs an iterative equation solver for a same-sized complex-valued linear system.

B. Other Experimental Results

Another test case is a single-contact structure presented in [2]. The DCT-accelerated Green's function method was employed to extract the frequency-dependent substrate parasitics. We calculate this structure with subRCbem and compare the results with those obtained from [2]. For two substrate profiles, the results of our method coincide with those in [2, Fig. 5] very well. The maximum discrepancy is only 2.0%.

The last case is a substrate with noise reduction components, where two contacts are surrounded by two components with buried materials, respectively. The effect of noise reduction is evaluated with the resistivity of buried region changing. Because the first step in subRCbem is only related with geometry parameters, and need not be performed again while the resistivity changed, subRCbem consumes less computational time for this simulation task. With the resistivity of buried region decreasing from 0.12 to $5 \times 10^{-6}\ \Omega \cdot \mathrm{m}$, computational results show that the contact coupling impedance increases for orders of magnitude. This means that, with the conductive ability increasing, the noise reduction effect becomes prominent.

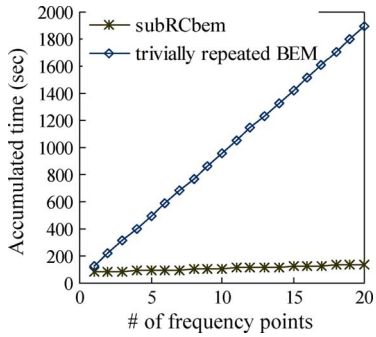

Fig. 7. Accumulated CPU time of subRCbem and trivially repeated BEM for multi-frequency extraction.

V. CONCLUSION

A two-step approach based on DBEM is proposed to extract the frequency-dependent substrate parasitics. The first step is equal to extracting the frequency-independent substrate resistance. Then, its solution is reused for computing the substrate parameters at different frequencies. With the help of the Sherman–Morrison–Woodbury formula, only a small linear system is solved for each frequency. Therefore, the propose method is much more efficient than the approach solving the whole DBEM equations for each frequency repeatedly [4]. Numerical results show that the proposed method is hundreds of times faster than the ASITIC based on the DCT-accelerated Green's function method [2] while preserving good accuracy.

REFERENCES

[1] J. P. Costa, M. Chou, and L. M. Silveira, "Efficient techniques for accurate modeling and simulation of substrate coupling in mixed-signal IC's," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 18, no. 5, pp. 597–607, May 1999.

[2] A. M. Niknejad, R. Gharpurey, and R. G. Meyer, "Numerically stable Green function for modeling and analysis of substrate coupling in integrated circuits," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 17, no. 4, pp. 305–315, Apr. 1998.

[3] R. Ghapurey and R. G. Meyer, "Analysis and simulation of substrate coupling in integrated circuits," *Int. J. Circuit Theory Appl.*, vol. 23, no. 4, pp. 381–394, Jul./Aug. 1995.

[4] X. Wang, W. Yu, and Z. Wang, "A new boundary element method for accurate modeling of lossy substrates with arbitrary doping profiles," in *Proc. ASP-DAC*, Jan. 2006, pp. 683–688.

[5] X. Wang, W. Yu, and Z. Wang, "Efficient direct boundary element method for resistance extraction of substrate with arbitrary doping profiles," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 25, no. 12, pp. 3035–3042, Dec. 2006.

[6] A. Vithayathil, X. Hu, and J. White, "Substrate resistance extraction using a multi-domain surface integral formulation," in *Proc. Int. Conf. Simul. Semicond. Processes Devices*, Sep. 2003, pp. 323–326.

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

[8] W. Yu and Z. Wang, "Enhanced QMM-BEM solver for 3-D multiple-dielectric capacitance extraction within finite domain," *IEEE Trans. Microw. Theory Tech.*, vol. 52, no. 2, pp. 560–566, Feb. 2004.

[9] A. M. Niknejad and R. G. Meyer, "Analysis, design, and optimization of spiral inductors and transformers for Si RF IC's," *IEEE J. Solid State Circuits*, vol. 33, no. 10, pp. 1470–1481, Oct. 1998.
[10] A. M. Nijnejad, *ASITIC*. [Online]. Available: http://rfic.eecs.berkeley.edu/~niknejad/
[11] W. Gao and Z. Yu, "Scalable compact circuit model and synthesis for RF CMOS spiral inductors," *IEEE Trans. Microw. Theory Tech.*, vol. 54, no. 3, pp. 1055–1064, Mar. 2006.
[12] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 3rd ed. Baltimore, MD: Johns Hopkins Univ. Press, 1996, p. 50.

Bitwidth Reduction via Symbolic Interval Analysis for Software Model Checking

Aleksandr Zaks, Zijiang Yang, Ilya Shlyakhter, Franjo Ivančić, Srihari Cadambi, Malay K. Ganai, Aarti Gupta, and Pranav Ashar

***Abstract*—This paper presents a lightweight interval analysis technique for determining the lower and upper bounds for program variables and its application in improving software model checking techniques. The experiments demonstrate that it is an effective approach to alleviate the state explosion problem in software model checking.**

***Index Terms*—Abstract interpretation, interval analysis, model checking, program analysis, software engineering.**

I. Introduction

Model checking [1] suffers from the *state explosion* problem, i.e., the number of states to explore exponentially grows with the number of state elements. In hardware verification, the problem is addressed using symbolic model checking [2] based on binary decision diagrams (BDDs) [3] or Boolean satisfiability (SAT) solvers [4], [5], where the states are implicitly encoded. However, symbolic model checking is less successful in the context of software verification based on bit-precise modeling of program variables. In this paper, we statically determine possible intervals for values of program variables in order to use this information to extract smaller verification models. The use of this information greatly improves the performance of back-end model checking techniques based on bit-precise reasoning.

Interval analysis is a broad field in which rigorous mathematics is associated with scientific computing. Modern development of interval arithmetic began with Moore's dissertation [6]. Since then, numerous research articles and books have appeared on the subject. In terms of the actual interval analysis techniques used, several methods have been proposed for purposes different from software model checking [7]–[10]. In the context of hardware verification, interval analysis has been used for data path abstraction [11]. Their approach is based on a fixed-point characterization of intervals [12].

Our main method is based on the framework suggested in [13]. However, we perform the analysis for constraint systems that can contain both disjunctions and conjunctions of linear inequalities, instead of just conjunctions. Furthermore, our particular modeling framework allows us to make certain simplifying assumptions that translate to a more efficient computation of tight bounds on variables. The concept of bounded interval analysis has not been addressed before and is a contribution of this paper. We have implemented the interval analysis techniques presented in this paper in our software model checker for C programs called F-SOFT [14]. We demonstrate their usefulness in practice on many benchmark examples.

Manuscript received July 18, 2007; revised December 19, 2007. This paper was recommended by Associate Editor R. F. Damiano.

A. Zaks is with Google, Mountain View, CA 94043 USA.
Z. Yang is with Western Michigan University, Kalamazoo, MI 49008-5201 USA (e-mail: zijiang.yang@wmich.edu).
I. Shlyakhter is with the Broad Institute of MIT and Harvard, Cambridge, MA 02142 USA.
F. Ivančić, S. Cadambi, M. K. Ganai, A. Gupta, and P. Ashar are with NEC Laboratories America, Inc., Princeton, NJ 08540 USA.

Digital Object Identifier 10.1109/TCAD.2008.925777

II. Constraint System Generation

Given a C program, F-SOFT applies a series of source-to-source transformations to translate a full-fledged C into smaller subsets of C, until the program state is represented as a collection of simple scalar variables and each program step is represented as a set of parallel assignments to these variables. The basic unit in the software model is a *basic block* that consists of C code that has one entry point, one exit point, and no branch instructions contained within it. In addition, a variable is assigned at most once in a basic block. For more information about the software model generation, please refer to [15].

A. Symbolic Bounds

We follow the general framework presented in [13] to generate symbolic constraints. For each basic block B_i of a procedure f, we define two locations: 1) pre_i, which represents the start of basic block B_i, and 2) post_i, which represents the end of B_i. Let V_f be the set of local integer and pointer variables of procedure f, and let $P_f \subseteq V_f$ be the set of formal parameters. We use v_{loc} to denote the value of the variable v at program location loc and p_0 to symbolically represent the value of the actual parameter that corresponds to $p \in P_f$.

For each variable $v \in V_f$ and location loc, let L^v_{loc} and U^v_{loc} represent the lower and upper bounds, respectively, of the value of v at loc. We set L^v_{loc} and U^v_{loc} to be linear combinations of the parameters of f with unknown rational coefficients, as defined in the following:

$$L^v_{\text{loc}} = C^L + \sum_{p \in P_f} C^L_p \cdot p_0 \quad U^v_{\text{loc}} = C^U + \sum_{p \in P_f} C^U_p \cdot p_0.$$

In order to obtain the lower and upper bounds of each variable, we consider the following four types of constraints.

Initialization Constraints: We generate initialization constraints for location pre_0, which represents the beginning of the initial block B_0. For each $p \in P_f$, we require that $L^p_{\text{pre}_0} = U^p_{\text{pre}_0} = p_0$. For each $v \in V_f \setminus P_f$, we set $L^v_{\text{pre}_0} = -\infty$ and $U^v_{\text{pre}_0} = +\infty$.

Assignment Constraints: The lower bound of an expression e at location loc, i.e., $l(e, \text{loc})$, can be computed as follows:

$$\begin{aligned} l(c, \text{loc}) &= c \\ l(v, \text{loc}) &= L^v_{\text{loc}} \\ l(e_1 + e_2, \text{loc}) &= l(e_1, \text{loc}) + l(e_2, \text{loc}) \\ l(e_1 - e_2, \text{loc}) &= l(e_1, \text{loc}) - u(e_2, \text{loc}) \\ l(c \cdot e, \text{loc}) &= \begin{cases} c \cdot l(e, \text{loc}), & c \geq 0 \\ c \cdot u(e, \text{loc}), & c \leq 0. \end{cases} \end{aligned}$$

Whenever we cannot compute a bound, we let $l(e, \text{loc}) = -\infty$. Similarly, we can define $u(e, \text{loc})$ for the upper symbolic bounds of expressions.