

# Stable and Efficient Reduction of Substrate Model Networks Using Congruence Transforms

Kevin J. Kerns, Ivan L. Wemple, and Andrew T. Yang

Department of Electrical Engineering  
University of Washington, Seattle, WA 98195

## Abstract

*Parasitic analog-digital noise coupling has been identified as a key issue facing designers of mixed-signal integrated circuits. In particular, signal cross talk through the common chip substrate has become increasingly problematic. This paper demonstrates a new methodology for developing simulation, synthesis, and verification models to analyze the global electrical behavior of the non-ideal semiconductor substrate. RC substrate network models, which are generated via a triangular discretization method, are accurately approximated for subsequent analysis by an efficient reduction algorithm. This algorithm utilizes the well-conditioned Lanczos process to formulate Padé approximations of the network port admittance. Congruence transformations are employed to ensure stability, and to create reduced networks which are easily realizable with SPICE-compatible RC elements. For validation, the strategy has been successfully applied to several mixed-signal circuit examples.*

## 1: Introduction

Industry trends aimed at integrating higher levels of circuit functionality have triggered a proliferation of analog and digital subsystems fabricated side-by-side on the same die. The combined requirements for both high speed digital and high precision analog circuitry produce unique challenges to mixed-A/D circuit designers. Specifically, monolithic mixed-signal ICs are often characterized by parasitic analog-digital interactions which can cripple the operation of high-performance designs. Noise coupling through the common chip substrate has been identified as a significant contributor to this important problem [1].

Modeling the electrical behavior of non-ideal semiconductor substrates is of key interest to the mixed-signal design community. For state-of-the-art circuits, chip-level verification which excludes the effects of substrate coupling is of questionable validity. As a result, substrate modeling for circuit simulation has been the focus of much research in recent years. Early work in this area [2] used a box integration technique to construct 3-D rectangular RC mesh networks as equivalent circuit representations of the modeled substrates. The mesh topology could be correlated to the circuit's physical design by distributing grid points according to the layout features on relevant fabrication photomasks. Unfortunately, layout-driven *rectangular* grid generation is prone to substrate "overpartitioning", which yields unnecessar-

ily dense grid crowding in many regions of the chip. The strategy produces enormous circuit networks, even for moderately sized layouts. Since a primary objective of equivalent circuit macro-modeling is to build simulation-ready networks, the inordinate complexity of the generated models is self-defeating — subsequent simulation on conventional CAD workstations becomes virtually impossible.

To address the complexity issues, intermediate processing is required to approximate the generated linear RC networks by smaller circuits which exhibit similar electrical properties. Since a typical mesh is dense and three-dimensional, only a small percentage of network nodes, called *ports*, are physically connected to the external circuit (at the top surface of the modeled substrate). In theory, an "equivalent" network can be formulated by eliminating a substantial fraction of the *internal* nodes. The resulting network is appropriate for simulation if its port characteristics remain consistent with those of the original mesh. This technique is generally referred to as *network reduction*.

Asymptotic Waveform Evaluation (AWE) [3] has been proposed as a method to reduce RC networks for mixed-signal switching noise analysis [4]. The AWE algorithm approximates a network's multiport behavior by recursively calculating the moments of the port characteristics and then fitting these moments to pole-residue functions via the Padé approximation. A well-known problem with this technique is that calculation of the higher moments is inherently ill-conditioned— increasing the number of poles used to model a given network does not guarantee a better approximation. Heuristic methods have been developed to address this issue (e.g., see references in [5]), but only at the cost of increased computational complexity. Another problem with AWE relates to the stability of the network approximation. While *asymptotic stability* is maintained by eliminating positive poles, *absolute stability* is not easily ensured. Consequently, non-physical, artificial oscillations may appear during subsequent transient simulations.

Network models detailed enough to accurately predict the chip-level impact of substrate coupling are, by necessity, very complex. It is not surprising that tangible results have been demonstrated only for small device-scale examples, as existing methods possess inherent limitations which render them impractical for even moderately sized circuits. The mixed-signal design process can be greatly enhanced by the development of software tools which can efficiently extract accurate chip-level simulation

substrate models directly from a physical design specification. Reliable verification of circuit functionality obviously reduces the length of the overall design cycle and promotes the likelihood of first-time silicon success. Perhaps more importantly, robust new noise reduction techniques can be developed more rapidly if capabilities exist to accurately assess and analyze the impact of switching noise in proposed design methodologies.

We propose a substrate modeling strategy which addresses the mesh complexity issue at *both* the model generation *and* network reduction levels. For initial mesh generation, a well-known geometric construct can be efficiently applied to overcome the single most important drawback of *rectangular* mesh formulation methods— that is, localized mesh refinement, often required in regions of dense switching activity, is propagated to distant layout regions where a coarser mesh might otherwise be adequate. By using a non-rectangular gridding method based on *Voronoi tessellation*, we extract a mesh which automatically and locally adjusts itself to the density of substrate features as inferred from the layout specification. Mesh extraction based on this approach generates substrate circuit networks containing orders of magnitude fewer circuit nodes than those of conventional gridding techniques. A description of the substrate model formulation is given in Section 2.

In spite of the improved model generation technique, extracted full-chip substrate networks still promise to be exceedingly complex. For model reduction, we demonstrate a new multiport algorithm which fully exploits the pure-RC property of our formulated networks and directly generates reduced equivalent circuit models in a well-conditioned manner. Using *congruence transformations*, full-network conductance and susceptance matrices are transformed to reduced equivalents which can be directly realized with resistors and capacitors. The approximated networks are guaranteed to be passive, and thus well-behaved in subsequent simulations. Proper formulation of the transformation ensures that the networks possess a minimal number of internal nodes, and yield a specified accuracy from DC to a maximum frequency of interest. The requisite transforms are generated using a symmetric Lanczos method which exploits the specialized structure of the extracted substrate networks. Required matrix inversions are performed using efficient methods which also profit from the problem symmetry. Since matrix inversion often accounts for a substantial network reduction bottleneck, this strategy can be significantly faster than general AWE methods, which employ nonsymmetric techniques. Section 3 provides the theoretical details and an implementation description of the proposed substrate model network reduction methodology. The non-rectangular gridding strategy and the congruence-transformation-based network reduction algorithm combine to form a unified, efficient strategy for developing parasitic substrate models for mixed-signal circuit simulation and design verification. The overall approach has been applied to several mixed-signal design examples, which we present in Section 4.

## 2: Substrate model extraction via Voronoi tessellation

A popular and physically-based approach to parasitic substrate modeling employs an equivalent mesh representation of

the modeled substrate [2], [4], [6]. A common drawback of previously reported modeling strategies, however, is that the derived networks ultimately contain circuit nodes in substrate regions where they are not required to obtain accurate simulation results. As emphasized in our introduction, subsequent mesh processing typically involves network reduction, and, ultimately, simulation. Since the computational efficiency of these procedures is directly impacted by the complexity of the generated network, it is of enormous advantage to constrain the size of the original mesh by adopting techniques to formulate more compact models. For the mesh *generation* phase of substrate modeling, we employ a *non-rectangular* substrate discretization based on geometric methods called *Voronoi tessellation* and *Delaunay triangulation* [7]. The derived mesh efficiently conforms to the substrate feature topology as dictated by the physical layout of the circuit. This section summarizes the modeling approach.

Our strategy is motivated by the observation that most real chips contain areas of intricate complexity surrounded by comparatively large regions with little structural detail. In gate array, standard-cell, and most custom designs, large chip areas contain no active devices but are dedicated to routing channels. Since transistors and contacts are the primary sources and collectors of noise current, it makes sense to partition the chip according to the “localized” densities of relevant substrate features.

Voronoi tessellation is a geometric method which subdivides the Euclidean plane according to a distribution of point sites. Applied to substrate modeling, the sites are determined during layout extraction by the locations of the transistors, well edges, and substrate tie-downs. For each site, the tessellation assigns a convex polygon which spans the region of the plane that is nearer to the enclosed site than to any other site. The polygons partition the substrate surface into a convex net called the *Voronoi diagram*. Fig. 1 (a) illustrates the Voronoi diagram for a random collection of tessellation sites. Circuit nodes corresponding to the sites are used to define a nodal mesh topology in the  $(x,y)$  plane, which we term a *nodeplane*. Multiple nodeplanes are then stacked on top of each other and interconnected site-to-site to extend the mesh in the  $z$ -direction. The extent of physical separation between adjacent planes largely depends on the substrate doping profiles, and the total mesh thickness corresponds to the thickness of the modeled substrate.

Connecting each pair of sites that share a common Voronoi edge is called Delaunay triangulation. Fig. 1 (b) shows the triangulation for the sites in (a). The edges of the triangulation physically connect adjacent sites in each nodeplane and can be used to represent the branches of a corresponding network. The model-

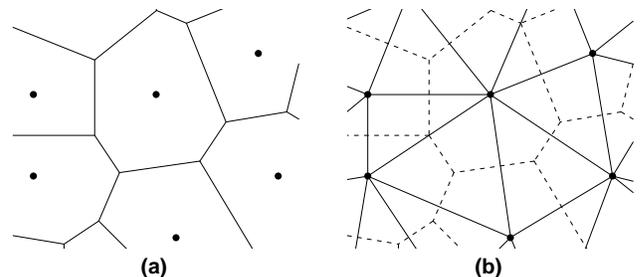


Figure 1. (a) A Voronoi diagram and (b) the corresponding Delaunay triangulation.

ing strategy combines information about the substrate properties and the geometries of the individual Voronoi polygons and associated Delaunay triangulation edges to derive the values of the resistors and capacitors which comprise the network. To accommodate the depletion capacitance associated with the well junctions, special sites are introduced in pairs which straddle the well boundaries. As a result, the corresponding triangulation is always perpendicular to the well edges, and modeling the junction capacitance is simplified. Fig. 2 shows a CMOS D flip-flop layout and a pictorial representation of the triangulated network topology used to model the top  $(x,y)$  substrate nodeplane.

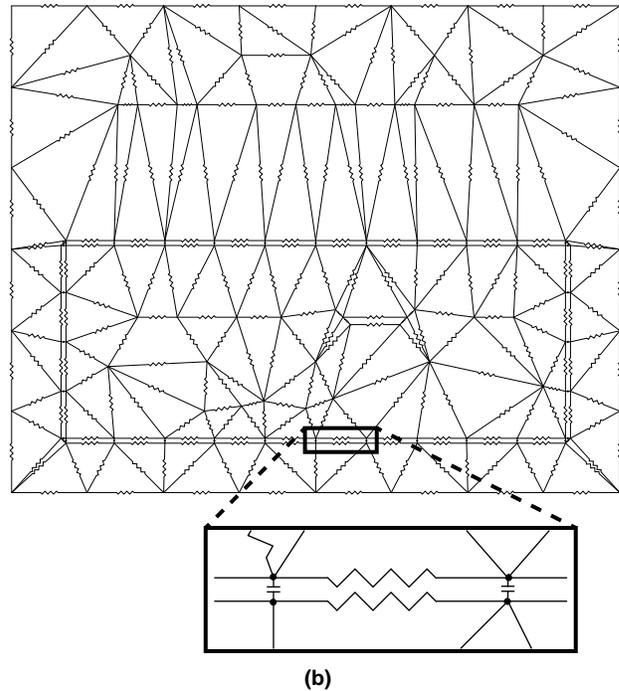
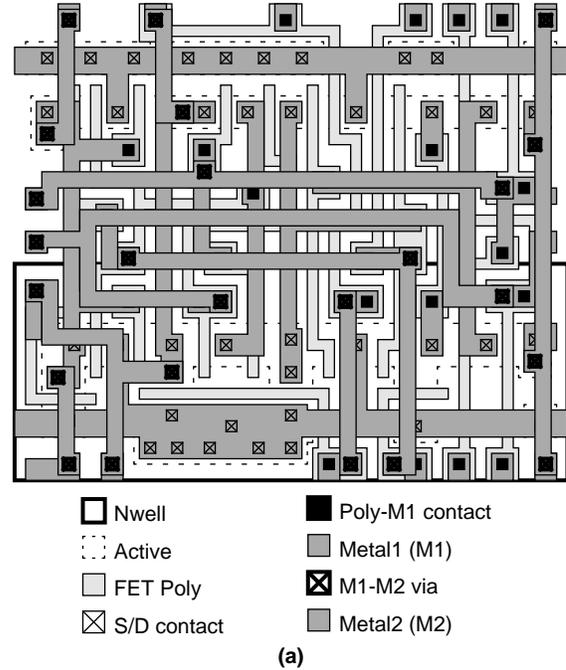
The tessellation procedure is computationally efficient, with an  $O(N \log N)$  time complexity and  $O(N)$  memory requirement. Since the  $(x,y)$  mesh topology is duplicated plane-to-plane,  $N$  is the number of sites (i.e., circuit nodes) in a *single* nodeplane. Strictly speaking, it is the Delaunay triangulation which forms the basis for the network topology. But since the tessellation and triangulation form mathematical duals, the triangulation is derived at no additional expense. In [8] it was shown that the quantity of circuit nodes in models generated by Voronoi tessellation are orders of magnitude fewer than in networks derived by rectangular discretization techniques. The accuracy of the approach has been verified by both detailed 2-D device simulation, and, for small circuits, by SPICE circuit-level simulation. Extensive comparisons were made to networks obtained by conventional gridding methods. The simulation results and details of the Voronoi-tessellated network formulation appear in [8].

### 3: Efficient RC network reduction using congruence transformations

In spite of the reduced complexity achieved by improved model generation, the extracted substrate networks are still too large for conventional circuit simulation. Some method of mesh reduction is required to approximate the networks using simpler equivalent circuit models. To maintain accuracy, efficiency, and flexibility, an acceptable network reduction algorithm should meet the following requirements:

- The simulation macromodels must accurately approximate the multiport admittance of the original RC network from DC to a specified maximum frequency.
- To obtain accurate, physically-based simulation results, the reduced substrate networks must be passive so that absolute stability is preserved.
- For efficient simulation, the reduced network models must contain a near-minimum quantity of internal nodes and branches.
- To maintain compatibility with an assortment of circuit simulation and timing analysis software, the reduced models must be realizable with standard SPICE-compatible circuit elements.

As mentioned in Section 1, Asymptotic Waveform Evaluation has been proposed as a technique for reducing the computational expense incurred by the simulation of substrate model circuit networks. Some fundamental drawbacks of AWE (ill-conditioning, no guarantee of absolute stability, etc.) were discussed ear-



**Figure 2. (a) Layout representation of a type D flip-flop based on the MOSIS SCMOS double metal, double poly process. (b) The top nodeplane substrate model network topology derived from layout. The inset shows the well capacitance.**

lier. Feldman and Freund recently introduced a new multiport linear circuit approximation technique called Matrix Padé Via a Lanczos-type process (MPVL) [9]. In contrast to AWE, this method avoids using ill-conditioned recursive moment calculations for pole-residue formulation, but the issue of reduced network stability and passivity has not yet been addressed. And

because MPVL is a general technique, the algorithm does not capitalize on the unique properties of pure RC networks — as a result, relatively inefficient matrix techniques must be used for the required matrix inversions.

In this work, we propose a multiport algorithm which generates *passive* circuit approximations of the extracted substrate networks. The network poles are retained from DC to a specified upper frequency limit, within a specified error bound. Using congruence transformations, the full-network admittance matrices are transformed to simpler equivalents, which are used to generate SPICE-compatible RC netlist representations of the reduced circuits. Proper formulation of the transformation ensures that the networks possess a minimal number of internal nodes. The approach capitalizes fully on the inherent properties of the substrate models. This section summarizes the reduction strategy.

### 3.1: Multiport admittance matrix formulation

The admittance of an RC network with  $m$  ports and  $n$  internal nodes can be represented by the conductance and susceptance matrices,  $\mathbf{G}$  and  $\mathbf{C}$ . These matrices have  $l = m + n$  rows and columns and relate voltage and current in the frequency domain by

$$(\mathbf{G} + s\mathbf{C})\mathbf{x} = \mathbf{b}. \quad (1)$$

Here,  $\mathbf{x}$  and  $\mathbf{b}$  are column vectors with  $l$  rows representing nodal voltages and injected currents, respectively, and  $s$  is complex frequency. Since the network contains only resistors and capacitors, both  $\mathbf{G}$  and  $\mathbf{C}$  are symmetric. If the resistors and capacitors are real and positive, then  $\mathbf{G}$  and  $\mathbf{C}$  are non-negative definite, which means that none of the eigenvalues of either matrix is negative.

A logical partitioning of the voltage vector,  $\mathbf{x}$ , in (1), orders the entries which correspond to the port nodes first, followed by the internal node entries, i.e.,

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}' \\ \mathbf{x}'' \end{bmatrix}. \quad (2)$$

Accordingly,  $\mathbf{G}$  and  $\mathbf{C}$  can be partitioned as follows:

$$\mathbf{G} = \begin{bmatrix} \mathbf{A} & \mathbf{Q}^T \\ \mathbf{Q} & \mathbf{D} \end{bmatrix} \text{ and } \mathbf{C} = \begin{bmatrix} \mathbf{B} & \mathbf{R}^T \\ \mathbf{R} & \mathbf{E} \end{bmatrix}, \quad (3)$$

where  $\mathbf{A}$  and  $\mathbf{B}$  are  $m \times m$ ,  $\mathbf{Q}$  and  $\mathbf{R}$  are  $n \times m$ , and  $\mathbf{D}$  and  $\mathbf{E}$  are  $n \times n$  and non-negative definite. Also,  $\mathbf{D}$  is positive definite provided each internal node has a dc path to either a port node or the common node of the network (i.e. the existence of a dc solution for  $\mathbf{x}''$  implies that  $\mathbf{D}$  is non-singular).

To formulate  $\mathbf{Y}(s)$ , the multiport admittance of the network, we first re-write (1) as

$$\begin{bmatrix} (\mathbf{A} + s\mathbf{B}) & (\mathbf{Q}^T + s\mathbf{R}^T) \\ (\mathbf{Q} + s\mathbf{R}) & (\mathbf{D} + s\mathbf{E}) \end{bmatrix} \begin{bmatrix} \mathbf{x}' \\ \mathbf{x}'' \end{bmatrix} = \begin{bmatrix} \mathbf{b}' \\ \mathbf{0} \end{bmatrix}. \quad (4)$$

The last  $n$  elements of the column vector  $\mathbf{b}$  are identically zero because, by definition, no external current can be directly injected into the internal nodes. Equation (4) represents two equations with two unknowns, namely  $\mathbf{x}'$  and  $\mathbf{x}''$ , which, according to the partitioning, are the port and internal node voltage vectors, respectively. Using the definition  $\mathbf{Y}(s)\mathbf{x}' = \mathbf{b}'$ , and eliminating  $\mathbf{x}''$  in (4) gives

$$\mathbf{Y}(s) = (\mathbf{A} + s\mathbf{B}) - (\mathbf{Q}^T + s\mathbf{R}^T)(\mathbf{D} + s\mathbf{E})^{-1}(\mathbf{Q} + s\mathbf{R}). \quad (5)$$

The *poles* of  $\mathbf{Y}(s)$  occur where  $(\mathbf{D} + s\mathbf{E})$  is singular, and are equal to  $-\lambda^{-1}$  where  $\lambda$  is the solution to the generalized symmetric eigenvalue problem

$$\det[\mathbf{E} - \lambda\mathbf{D}] = 0. \quad (6)$$

Since  $\mathbf{E}$  is symmetric non-negative definite and  $\mathbf{D}$  is symmetric positive definite, the poles of  $\mathbf{Y}(s)$  are real and negative.

The moments,  $\mathbf{Y}_k$ , defined such that  $\mathbf{Y}(s) = \mathbf{Y}_0 + s\mathbf{Y}_1 + s^2\mathbf{Y}_2 + \dots$ , are found by first expanding  $(\mathbf{D} + s\mathbf{E})^{-1}$  of (5) as a Taylor series about  $s = 0$ , i.e.,

$$(\mathbf{D} + s\mathbf{E})^{-1} = [\mathbf{I} + s(-\mathbf{D}^{-1}\mathbf{E}) + s^2(-\mathbf{D}^{-1}\mathbf{E})^2 + \dots]\mathbf{D}^{-1}. \quad (7)$$

Substituting (7) into (5) and equating coefficients of like powers of  $s$  yields

$$\mathbf{Y}_0 = \mathbf{A} - \mathbf{Q}^T\mathbf{D}^{-1}\mathbf{Q} \quad (8)$$

$$\mathbf{Y}_1 = \mathbf{B} + \mathbf{Q}^T\mathbf{D}^{-1}\mathbf{P} - \mathbf{R}^T\mathbf{D}^{-1}\mathbf{Q} \quad (9)$$

$$\mathbf{Y}_k = -\mathbf{P}^T(-\mathbf{D}^{-1}\mathbf{E})^{k-2}\mathbf{D}^{-1}\mathbf{P} \quad (k \geq 2), \quad (10)$$

where

$$\mathbf{P} = \mathbf{E}\mathbf{D}^{-1}\mathbf{Q} - \mathbf{R}. \quad (11)$$

Note that (8)-(11) can be used to recursively generate the moments of  $\mathbf{Y}(s)$  in a manner similar to AWE. However, due to symmetry, *two* moments can be matched for each set of  $m$  forward and backward substitutions. In contrast, AWE matches only one moment for the same amount of computational effort. Also, the matrix to be inverted,  $\mathbf{D}$ , is symmetric positive definite with negative off-diagonal elements. This property is significant because techniques more efficient than sparse LU factorization (e.g., Cholesky factorization [10]) can be used for inversion. Finally, we do not apply (8)-(11) directly, as this would yield the same ill-conditioning problem from which AWE suffers. Instead, we utilize the *congruence transformation*, which can be used to efficiently reduce  $\mathbf{G}$  and  $\mathbf{C}$  in a well-conditioned manner and, at the same time, preserve the moments,  $\mathbf{Y}_k$ , of (8)-(10).

### 3.2: Accurate moment matching by congruence transformations

A congruence transformation applied to  $\mathbf{A}$  is defined as the transformation  $\mathbf{B} = \mathbf{V}^T\mathbf{A}\mathbf{V}$ , where  $\mathbf{V}$  is square and non-singular.  $\mathbf{V}$  is referred to as the *congruence transform*, and the matrices  $\mathbf{A}$  and  $\mathbf{B}$  are said to be *congruent*. A fundamental property of congruence transformations is that they preserve the eigenvalues of the generalized eigenvalue problem demonstrated in (6) above (see [11], chapter 8). The eigenvalues of

$$\det[\mathbf{V}^T\mathbf{E}\mathbf{V} - \lambda\mathbf{V}^T\mathbf{D}\mathbf{V}] = 0 \quad (12)$$

(where  $\mathbf{V}$  is square and non-singular) are identical to those of (6). This property is useful since, if eigenvalues are preserved, the poles of  $\mathbf{Y}(s)$  are also retained. In this work, we make use of network transformations using the transform,  $\mathbf{V}$ , which has  $n$  rows and  $k$  linearly independent columns, with  $k \leq n$ . If  $k < n$  the transform is an *incomplete congruence transform*, but we henceforth refer to all cases as congruence transforms and identify the special cases where  $\mathbf{V}$  is square.

In (5), inserting  $\mathbf{V}\mathbf{V}^{-1}$  on the left side of  $(\mathbf{D} + s\mathbf{E})^{-1}$  and  $(\mathbf{V}\mathbf{V}^{-1})^T$  on the right side gives

$$\mathbf{Y}'(s) = (\mathbf{A} + s\mathbf{B}) - (\mathbf{Q}^T + s\mathbf{R}^T)\mathbf{V}(\mathbf{V}^T\mathbf{D}\mathbf{V} + s\mathbf{V}^T\mathbf{E}\mathbf{V})^{-1}\mathbf{V}^T(\mathbf{Q} + s\mathbf{R}). \quad (13)$$

We define the following simplifying relationships:

$$\mathbf{Q}' = \mathbf{V}^T \mathbf{Q} \quad (14)$$

$$\mathbf{R}' = \mathbf{V}^T \mathbf{R} \quad (15)$$

$$\mathbf{D}' = \mathbf{V}^T \mathbf{D} \mathbf{V} \quad (16)$$

$$\mathbf{E}' = \mathbf{V}^T \mathbf{E} \mathbf{V} \quad (17)$$

Substituting (14)-(17) in (13) yields

$$\mathbf{Y}'(s) = (\mathbf{A} + s\mathbf{B}) - (\mathbf{Q}' + s\mathbf{R}'^T)(\mathbf{D}' + s\mathbf{E}')^{-1}(\mathbf{Q}' + s\mathbf{R}'). \quad (18)$$

The transformations using  $\mathbf{V}$ , shown in (14)-(17), can be represented as congruence transformations on  $\mathbf{G}$  and  $\mathbf{C}$ , i.e.,

$$\mathbf{G}' = \mathbf{U}^T \mathbf{G} \mathbf{U} \quad \text{and} \quad \mathbf{C}' = \mathbf{U}^T \mathbf{C} \mathbf{U}, \quad (19)$$

and thus

$$(\mathbf{G}' + s\mathbf{C}') = \mathbf{U}^T (\mathbf{G} + s\mathbf{C}) \mathbf{U}, \quad (20)$$

where

$$\mathbf{U} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{V} \end{bmatrix}. \quad (21)$$

We later show that (20) has important implications related to the stability of the reduced network.

If  $k = n$  then  $\mathbf{V}\mathbf{V}^{-1} \equiv \mathbf{I}$  and  $\mathbf{Y}'(s) = \mathbf{Y}(s)$ . If  $k < n$ ,  $\mathbf{G}'$  and  $\mathbf{C}'$  are reduced in size (relative to  $\mathbf{G}$  and  $\mathbf{C}$ , respectively), which is our primary objective. In this case, however,  $\mathbf{V}\mathbf{V}^{-1} \neq \mathbf{I}$  (although  $\mathbf{V}^{-1}\mathbf{V} = \mathbf{I}$ ) so that  $\mathbf{Y}'(s) \neq \mathbf{Y}(s)$  and the transformed network has port characteristics which differ from those of the original network — non-square congruence transforms do not retain all the eigenvalues of (6), so the poles are not preserved.

To circumvent this difficulty, we note that, even if  $k < n$ , a proper choice of the transform  $\mathbf{V}$  will result in a congruence transformation which preserves the lower order moments of  $\mathbf{Y}(s)$  given by (8)-(10). For example, if  $\text{span}\{\mathbf{D}^{-1}\mathbf{Q}\} \subseteq \text{span}\{\mathbf{V}\}$ , it can be shown that the transform  $\mathbf{U}$  in (21) preserves  $\mathbf{Y}_0$  and  $\mathbf{Y}_1$  of (8) and (9). The *span* requirement means that each column of the product matrix  $\mathbf{D}^{-1}\mathbf{Q}$  is representable as some linear combination of the columns of  $\mathbf{V}$ . Additionally, if

$$\text{span}\{\mathbf{D}^{-1}\mathbf{Q}, \mathbf{D}^{-1}\mathbf{P}, \mathbf{D}^{-1}\mathbf{E}\mathbf{D}^{-1}\mathbf{P}, \dots, (\mathbf{D}^{-1}\mathbf{E})^{q-2}\mathbf{D}^{-1}\mathbf{P}\} \subseteq \text{span}\{\mathbf{V}\}, \quad (22)$$

then the transformation matches the first  $2q$  moments of  $\mathbf{Y}(s)$ . In general,  $\mathbf{V}$  must contain  $m$  columns for each pair of moments to be preserved, and the size of  $\mathbf{G}'$  and  $\mathbf{C}'$  is  $(q+1)m \times (q+1)m$  when  $2q$  moments are matched. If the columns of  $\mathbf{V}$  are linearly independent, then as  $q$  is increased, the poles of  $\mathbf{Y}'(s)$  converge to those of  $\mathbf{Y}(s)$ ; it is easily seen that  $\mathbf{Y}'(s) = \mathbf{Y}(s)$  when  $q$  is large enough to make  $\mathbf{V}$  a non-singular square matrix. By choosing the columns of  $\mathbf{V}$  to match the moments expanded about  $s = 0$ , as in (8)-(10), the low frequency poles converge first.

The *symmetric Lanczos process* (chapter 9 in [11]) for the generalized symmetric eigenvalue problem is used to generate the columns of  $\mathbf{V}$  in a well-conditioned manner. The process requires a single factorization of  $\mathbf{D}$ , and each block of  $m$  columns of  $\mathbf{V}$  requires  $m$  forward and backward substitutions and matches two moments. Both AWE and MPVL require  $2m$  substitutions to match the same number of moments because the general algorithms do not exploit symmetry. The properties of  $\mathbf{V}$  assert that  $\mathbf{D}'$  is diagonal and  $\mathbf{E}'$  is banded with a band width of  $m+2$ . Blocks are iteratively generated until the poles of  $\mathbf{Y}'(s)$  have converged to those of  $\mathbf{Y}(s)$  in a specified frequency range. Each iteration increases the size of the reduced network by  $m$  nodes and

matches two additional moments. The reduced networks generally contain non-converged high frequency poles in addition to those which have converged. These increase the size of the reduced network without impacting its electrical behavior in the specified frequency range. The extraneous poles can be removed by a post-processing step, which we describe in Section 3.3.

Congruence transformations preserve the symmetry and non-negative definiteness of  $\mathbf{G}$  and  $\mathbf{C}$ , even if  $k < n$ . The consequence of this is of great significance with respect to the stability of the reduced network. A *passive* substrate network is incapable of providing energy to the overall circuit. For the network represented by the immittance matrix  $\mathbf{W}(s) = \mathbf{G}' + s\mathbf{C}'$  to be passive, it is necessary that [12]:

- All elements of  $\mathbf{W}(s)$  are analytic for  $\sigma = \text{Re}(s) > 0$ ;
- $\mathbf{W}(s^*) = \mathbf{W}^*(s)$  where  $*$  denotes the complex conjugate; and
- $\mathbf{W}^{*T}(s) + \mathbf{W}(s)$  is non-negative definite for  $\sigma > 0$ .

We first note that the original network is passive, and  $\mathbf{G}$  and  $\mathbf{C}$  are real, non-negative definite, and symmetric. The first requirement is met because each element of  $\mathbf{G}'$  and  $\mathbf{C}'$  is a scalar. The second requirement is met because the transformation does not introduce complex numbers. The third is satisfied because

$$\mathbf{W}^{*T}(s) + \mathbf{W}(s) = 2(\mathbf{G}' + \sigma\mathbf{C}'). \quad (23)$$

Since  $\sigma\mathbf{C}'$  is non-negative definite (for  $\sigma \geq 0$ ), and since the sum of two real, symmetric, non-negative definite matrices is non-negative definite, then  $\mathbf{W}^{*T}(s) + \mathbf{W}(s)$  is non-negative definite for  $\sigma > 0$ . Consequently, *when applied for network reduction, any square or non-square congruence transformation on  $(\mathbf{G} + s\mathbf{C})$  preserves passivity, and thus ensures absolute stability.*

### 3.3: Implementation of the reduction algorithm

A flow diagram for the network reduction process is shown in Fig. 3. Inputs are the raw extracted RC substrate network, a specified upper frequency of interest, and an error tolerance. The network data is used to generate the conductance and susceptance

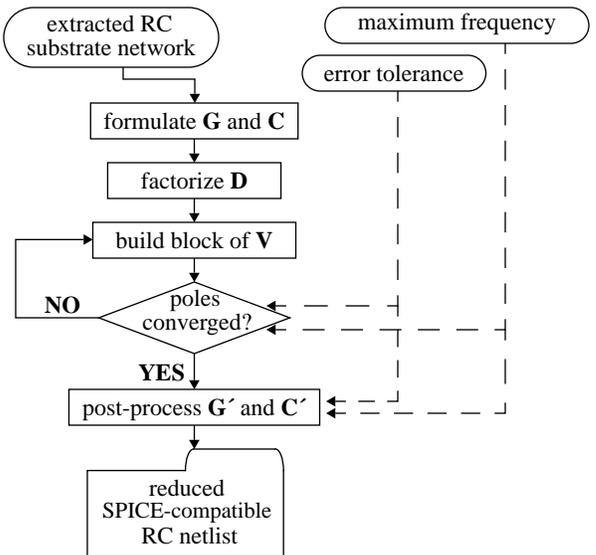


Figure 3. Flow diagram for substrate mesh reduction process.

matrices,  $\mathbf{G}$  and  $\mathbf{C}$ . The last  $n$  rows and columns of  $\mathbf{G}$  form the  $\mathbf{D}$  matrix, which is factorized. The congruence transform,  $\mathbf{V}$ , is then built iteratively using a symmetric Lanczos procedure. The process terminates when the “dominant” poles of the reduced network multiport admittance converge within the specified error tolerance. All pole frequencies less than the specified maximum frequency are automatically included. Program output is a SPICE-compatible RC netlist containing the reduced network.

In general, the derived  $\mathbf{G}'$  and  $\mathbf{C}'$  matrices contain unwanted (and non-converged) high frequency poles. To ensure that the final network is as simulation-efficient as possible, the unnecessary outlying poles are removed in a post-processing step. A square congruence transform is formulated and applied on  $\mathbf{E}'$  and  $\mathbf{D}'$ . The transformation is chosen to diagonalize the resulting congruent matrices without affecting the behavioral properties of  $\mathbf{Y}'(s)$ . Since the two matrices are diagonal, each internal node is associated with a single pole of  $\mathbf{Y}'(s)$ . The undesired poles and their associated network internal nodes can then be removed via transformation with a non-square congruence transform. The poles within the range of interest are unaffected, and the first two moments of  $\mathbf{Y}(s)$  are preserved. Since every post-processing transform is representable as a congruence transformation on  $(\mathbf{G}' + s\mathbf{C}')$ , network passivity is retained.

The algorithm employs a positive definite symmetric sparse matrix solver, which uses nested dissection [10] to pre-order the nodes for efficient factorization. With minimal computational expense, nested dissection provides near-optimal ordering for mesh-like networks, and the resultant matrix factorization time complexity is  $O(n^{1.5})$ . Here,  $n$  is the number of internal nodes in the original substrate network. The single most time-consuming operation is the factorization of  $\mathbf{D}$ . By applying positive definite symmetric sparse matrix techniques, we determine the factorization an order of magnitude faster than we would using general non-symmetric methods. Cumulatively, the forward-backward substitutions account for the next most expensive computational operations. Our formulation requires  $m$  of these for each pair of moments that are matched, where  $m$  is the number of port nodes.

A few remarks are in order regarding the similarities and differences between our approach and other linear network approximation techniques. On many accounts, the congruence transformation method offers substantial advantages over both AWE and MPVL. The first relates to performance. Our algorithm is specifically tailored for purely RC networks, whereas the AWE and MPVL methods are general (i.e., they are able to accommodate any linear network). With our strategy, problem formulation ensures that the  $\mathbf{D}$  matrix is positive definite and, consequently, more efficient specialized sparse matrix techniques are applied to determine  $\mathbf{D}^{-1}$ . It is important to note that we do *not* forsake accuracy for speed. The ill-conditioned higher order moment matching problems associated with AWE are well-known. Like MPVL, we overcome this pitfall by employing the Lanczos method. However, our formulation exploits symmetry in a way that published MPVL methods do not. Since the matrices  $\mathbf{D}$  and  $\mathbf{E}$  are symmetric, the span constraint of (23) is related to the generalized symmetric eigenvalue problem. Consequently, we employ a *symmetric* Lanczos algorithm. MPVL employs a *non-symmetric* Lanczos process, which requires twice as many forward-backward substitutions. As mentioned above, our exe-

cution profiles revealed that these seemingly benign computational operations are responsible for a significant portion of the total CPU time. Finally, we highlight the more fundamental differences. Our method directly generates admittance matrices which contain minimal quantities of poles and branches. The  $\mathbf{G}'$  and  $\mathbf{C}'$  matrices are easily “un-stamped” to directly realize the reduced RC network in a SPICE-compatible netlist format. More importantly, the reduced network is guaranteed to be passive, so subsequent circuit simulations are well-behaved.

#### 4: Applications to mixed-signal circuits

In this section we report on the results of the proposed modeling strategy applied to four mixed-signal circuit layouts:

- 1) A three-stage CMOS ring oscillator adjacent to a sensitive “analog” NMOS transistor (ro3+FET). The ring oscillator circuitry injects noise into the substrate, and we monitor the simulated voltage signal at the body terminal of the analog transistor.
- 2) A digital frequency divider and nearby analog current source (FreqDiv+Isrc) associated with a CMOS PLL. The subcircuits are separated by approximately 150 $\mu\text{m}$ . The divider circuitry switching transients are coupled through the substrate to degrade the operation of the current source. We monitor the simulated current source signal to study the impact of the switching noise on its designed value.
- 3) A CMOS operational amplifier completely surrounded by a ring oscillator (ro+OpAmp). We examine the impact of substrate coupling by monitoring the simulated op amp output signal resulting from a small-signal sinusoidal input.
- 4) The same analog op amp shielded by an ohmic guard ring (ro+OpAmpGua). We monitor the simulated noise on the amplifier output signal relative to the noise generated with no shielding (i.e., the result using ro+OpAmp, circuit (3)).

All substrate models were extracted directly from layout data using the process technology and mesh properties listed in Table 1. Level 3 transistor parameters derived from MOSIS parametric tests were used for SPICE simulations. The extracted and reduced network parameters for the four circuits are tabulated in Table 2, which also specifies the computational cost and network pole retention data associated with the mesh reduction runs. Mesh generation times are also listed. The frequency below which all poles are retained is determined by the error bound at the specified maximum frequency. We use a tolerance of 5% at 1

**Table 1. Substrate parameters for model generation.**

bulk doping (p-type)	$N_{sub} = 9.0 \times 10^{14} \text{ cm}^{-3}$
substrate thickness	100 $\mu\text{m}$
well doping (n-type)	$Z_j = 3.5 \mu\text{m}, N_0 = 5 \times 10^{16} \text{ cm}^{-3}$
channel stop doping	$Z_j = 0.2 \mu\text{m}, N_0 = 8.0 \times 10^{17} \text{ cm}^{-3}$
shallow nodeplanes ( $z < \text{well } Z_j$ )	5
deep nodeplanes ( $z < \text{well } Z_j$ )	5

**Table 2. Substrate modeling results for 4 mixed-A/D circuit examples.**

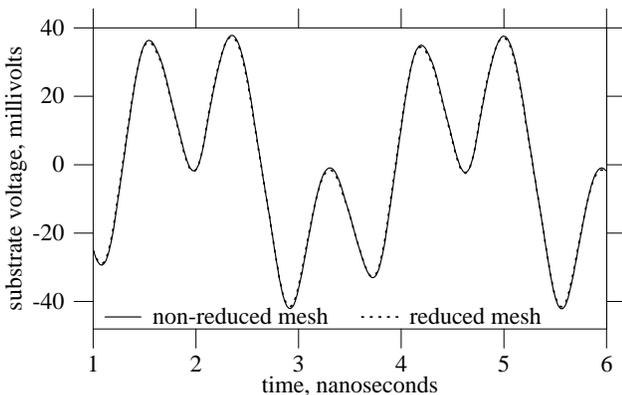
circuit description			mesh generation			mesh reduction				reduced network	
name	FETs	port nodes	CPU (sec)	total nodes	branches	CPU (sec)	poles retained			total nodes	branches
							number	min freq (GHz)	max freq (GHz)		
ro3+FET	7	10	0.6	767	2540	1.3	1	2.30	2.30	11	100
FreqDiv+Isrc	56	58	2.2	3438	11916	25.6	5	0.99	3.02	63	3596
ro+OpAmp	100	102	5.2	9149	31273	245.1	21	0.48	3.01	123	12445
ro+OpAmpGua	100	102	5.2	9461	32606	225.2	21	0.49	3.01	123	12445

GHz. All benchmarking was performed on a SUN Sparc20 workstation with 96 MBytes RAM. HSPICE 95.1 [13] was utilized for the circuit simulations.

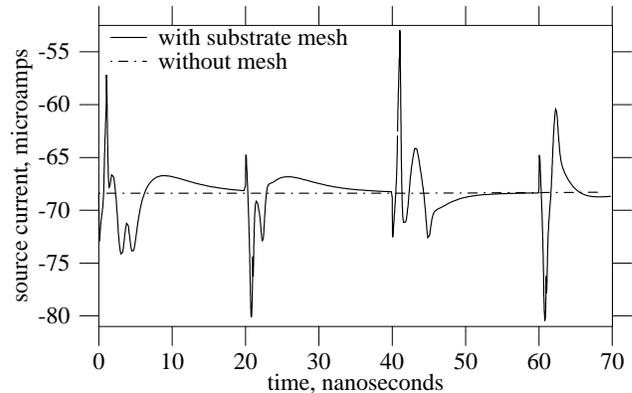
Circuit ro3+FET is simple, but useful, since its limited complexity permits SPICE simulation of the substrate mesh with *and* without network reduction. For each switching transient, the ring-oscillator injects substrate current across one of its output source/drain junction capacitors. The n-type “analog” MOSFET lies between the oscillator and the nearest substrate tie-down, which forces noise current to flow underneath the device. The signal at the transistor body terminal, Fig. 4, reflects the localized fluctuations in the substrate potential modeled by the underlying mesh network. The results obtained using the extracted model and its transformed equivalent demonstrate excellent agreement.

Transient substrate voltage fluctuations impair the performance of precision analog circuitry through manifestations of the well-known *body effect*. To assess the circuit impact of noise coupling, it is insightful to monitor the effects of the voltage fluctuations on the *performance* of the sensitive circuitry. Circuit FreqDiv+Isrc contains a CMOS (digital) frequency divider and (analog) source-coupled current source which represent nearby cells on a monolithic PLL chip. The divider accepts a 25 MHz input clock and generates signals at 12.5- and 6.25 MHz. The digital circuitry is characterized by full-swing signal transitions every 20 nanoseconds. The transients are coupled through the substrate to the current source, and, as a result, the source signal deviates from its DC value by more than 20%. By modeling the chip substrate, we are able to simulate the transient-induced noise which impairs the analog signal. The simulated current source waveform is shown in Fig. 5.

The magnitude of the noise waveform in Fig. 5 is artificially high because no effort was made to protect the sensitive circuitry



**Figure 4. HSPICE results for the ro3+FET circuit.**



**Figure 5. HSPICE results for the FreqDiv+Isrc circuit.**

from the effects of the digital subcircuit. In our final examples, ro+OpAmp and ro+OpAmpGua employ a linear amplifier circuit completely surrounded by a ring oscillator, and in the latter case, the amplifier is shielded by an ohmic guard ring. Figures 6 and 7 show four simulation waveforms, all representing the output signal of the amplifier with a 5 MHz small-signal sinusoidal input. For the first simulation (Fig. 6, curve A), we eliminated the source of noise by de-activating the ring oscillator. In the next simulation, we induced substrate-coupled switching noise by allowing the oscillator to run freely. The impact on the amplifier output is readily apparent in Fig. 6, curve B. Fig. 7 shows the results for the same simulations, except the substrate-coupling is inhibited by the presence of the guard ring.

Simulation CPU measurements for our test circuits *with* coupled substrate models are shown in Table 3. Excepting ro3+FET, the non-reduced substrate networks are far too complex for post-layout circuit analysis. For that single circuit, however, the speedup obtained with the reduced network is substantial. The simulations for the amplifier circuits are inordinately lengthy since we enforced very small transient timesteps to obtain the level of waveform detail shown in Figures 6 and 7.

Finally, we have emphasized throughout this paper that our

**Table 3. HSPICE simulation data.**

circuit	HSPICE CPU time (sec)		speedup
	non-reduced mesh	reduced mesh	
ro3+FET	1071.7	4.5	238x
FreqDiv+Isrc	†	543.3	–
ro+OpAmp	‡	3026.2	–
ro+OpAmpGua	–	2015.7	–

† memory requirement exceeded 96 MBytes

‡ terminated during setup matrix re-ordering after 20 hours.

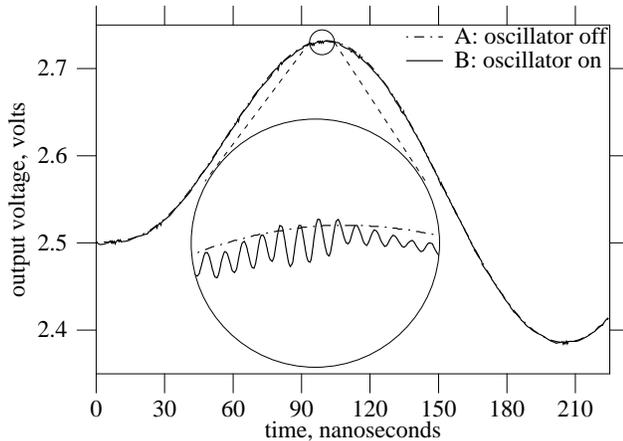


Figure 6. HSPICE results for the ro+OpAmp circuit.

algorithm exploits the matrix symmetries which characterize our formulation. Table 4 shows empirically-obtained data for the CPU times required to factorize the  $\mathbf{D}$  matrix for each substrate mesh network discussed in this section. We first employed sparse Cholesky factorization, which is standard for our system. Additionally, we implemented a version of the mesh reduction program which called the non-symmetric sparse LU factorization routines in *Sparse* [14]. The symmetric methods are faster by an order of magnitude. By comparing Table 4 data to the mesh reduction CPU times in Table 2, it is also evident that the  $\mathbf{D}$  factorization accounts for a substantial fraction of the overall run time. Tailoring the reduction algorithm to exploit the pure-RC properties of the extracted substrate networks improves the computational performance significantly.

## 5: Conclusions

We report on two techniques that greatly reduce the computational limitations associated with equivalent-circuit-based substrate coupling analysis in mixed-signal integrated circuits. RC substrate mesh networks are generated by applying Voronoi tessellation to layout-derived substrate feature data. The networks are efficiently approximated to any level of accuracy using congruent admittance matrices in conjunction with a well-conditioned Lanczos moment-matching process. Due to their enormous complexity requirements, previous approaches have been rigorously demonstrated only on small device-scale examples. With the aim of applying global substrate modeling to realistically large circuits, the proposed techniques make significant strides towards the development of a software methodology capable of analyzing mixed-signal noise behavior in a cognizant, quantitative fashion.

Table 4. Time required for the  $\mathbf{D}$  matrix sparse factorization — symmetric vs. non-symmetric methods.

circuit	internal nodes	Cholesky (CPU sec)	LU (CPU sec)
ro3+FET	757	0.5	3.8
FreqDiv+Isrc	3380	11.8	134.2
ro+OpAmp	9047	122.6	929.0
ro+OpAmpGua	9359	133.9	1163.1

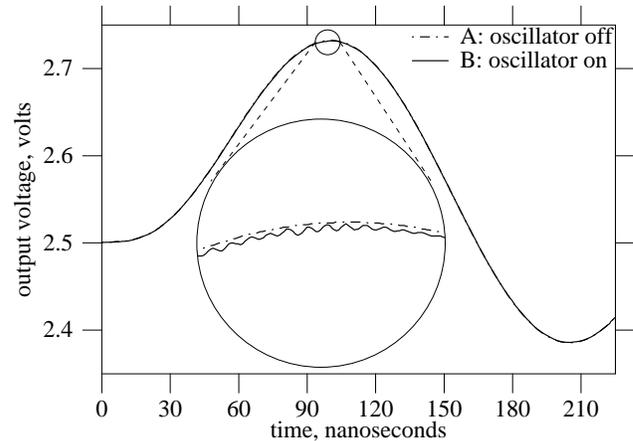


Figure 7. HSPICE results for the ro+OpAmpGua circuit.

## References

- [1] J.A. Olmstead and S. Vulih, "Noise problems in mixed analog-digital integrated circuits", *Proceedings of the IEEE Custom Integrated Circuits Conference*, pp. 659-662, 1987.
- [2] S. Kumashiro, R.A. Rohrer, and A.J. Strojwas, "A new efficient method for the transient simulation of three-dimensional interconnect structures", *Proceedings of the IEEE International Electron Devices Meeting*, pp. 193-196, 1990.
- [3] L.T. Pillage and R.A. Rohrer, "Asymptotic waveform evaluation for timing analysis", *IEEE Transactions on Computer-Aided Design*, vol. 9, no. 4, pp. 352-366, April 1990.
- [4] N.K. Verghese, D.J. Allstot, and S. Masui, "Rapid simulation of substrate coupling effects in mixed-mode ICs", *Proceedings of the IEEE Custom Integrated Circuits Conference*, pp. 18.3.1-18.3.4, 1993.
- [5] V. Raghavan, R.A. Rohrer, L.T. Pillage, J.Y. Lee, J.E. Bracken, and M.M. Alaybeyi, "AWE-inspired," *Proceedings of the IEEE Custom Integrated Circuits Conference*, pp. 18.1.3-18.1.8, 1993.
- [6] T.A. Johnson, R.W. Knepper, V. Marcello, and W. Wang, "Chip substrate resistance modeling technique for integrated circuit design", *IEEE Transactions on Computer-Aided Design*, vol. CAD-3, no. 2, pp. 126-134, April 1984.
- [7] S. Fortune, "A sweepline algorithm for Voronoi diagrams", *Algorithmica*, 2, pp. 153-174, 1987.
- [8] I.L. Wemple and A.T. Yang, "Mixed-signal switching noise analysis using Voronoi-tessellated substrate macromodels", *Proceedings of the 32nd ACM/IEEE Design Automation Conference*, pp. 439-444, 1995.
- [9] P. Feldmann and R.W. Freund, "Reduced-order modeling of large linear subcircuits via a block Lanczos algorithm", *Proceedings of the 32nd ACM/IEEE Design Automation Conference*, pp. 474-479, 1995.
- [10] A. George and J.W.-H. Liu, *Computer Solution of Large Sparse Positive Definite Systems*, Prentice-Hall, Englewood Cliffs, NJ, 1981.
- [11] G.H. Golub and C.F. Van Loan, *Matrix Computations*, second edition, Johns Hopkins University Press, Baltimore, MD, 1993.
- [12] M.R. Wohlers, *Lumped and Distributed Passive Networks: A Generalized and Advanced Viewpoint*, Academic Press, New York, NY, 1969.
- [13] *HSPICE*, Meta-Software, Inc., Version H95.1, Campbell, CA, 1995.
- [14] K.S. Kundert and A. Sangiovanni-Vincentelli, *User's Guide for Sparse: A Sparse Linear Equation Solver*, Version 1.3a, Department of EECS, University of California, Berkeley, CA, 1988.