Substrate Optimization Based on Semi-Analytical Techniques

Edoardo Charbon, *Member, IEEE*, Ranjit Gharpurey, Robert G. Meyer, *Fellow, IEEE*, and Alberto Sangiovanni-Vincentelli, *Fellow, IEEE*

***Abstract*— Several methods are presented for highly efficient calculation of substrate noise transport in integrated circuits. A three-dimensional Green's function-based boundary element method, accelerated through use of the fast Fourier transform, allows the computation of sensitivities with respect to all substrate parameters at a considerably higher speed than any methods reported in the literature. Substrate sensitivities are used in a number of physical optimization tools, such as placement and trend analysis. The aim is a fast and accurate estimation of the impact of technology migration and/or layout redesign on substrate noise and, ultimately, on the circuit's overall performance. The suitability of the approach is shown through industrial-strength mixed-mode integrated circuits fabricated on a standard CMOS process.**

***Index Terms*—Boundary element methods, convergence of numerical methods, discrete Fourier transients, integrated circuit noise, noise, noise generators, noise measurement, numerical analysis, numerical stability, optimization methods, phase noise, sensitivity, semiconductor device noise, switching circuits, switching transients.**

I. Introduction

INCREASED chip size, device density, and feature miniaturization, as well as overall higher frequencies of operation, have made the problem of substrate noise critical in the design of integrated circuits.

Accurately characterizing substrate noise is problematic for various reasons. The noise results from superposition of a large number of local and remote sources, each attenuated and delayed in a unique way. Modeling signal attenuation and delay individually may be extremely time-consuming and would require accurate *ad hoc* characterization of all the sources, which is in itself a hard problem [1]. The interaction of substrate noise with sensitive devices is often difficult to evaluate due to the complexity of the effects of noise on performance, especially with erratic noise waveforms. Moreover, reduced distances between high-swing high-frequency noise sources and sensitive circuitry exacerbates the problem, thus making the design task even more challenging.

To combat the effects of substrate noise, heavily overdesigned structures are generally adopted, thus seriously limiting the advantages of innovative technologies. For this reason, recently a serious effort has been made to model substrate noise sources and transport mechanisms, thus allowing designers to detect potential problems before fabrication. Specific guidelines have also been drafted for more aggressive, substrate-aware design practices.

At a macroscopic level, substrate noise is characterized by intrinsic and switching noise. Intrinsic noise is a background spurious signal originated in active and passive devices through various physical phenomena, namely thermal, shot, and flicker noise. Switching noise originates mostly in digital blocks where frequent state transitions, occurring in gates across the chip, result in current pulses absorbed from and transmitted to supply/ground lines through direct feedthrough and load charge/discharge. Such pulsing currents are partially injected into the substrate through impact ionization and capacitive coupling.

Generally, switching noise is by far the most destructive of all substrate noise types. It can be broadcasted over great distances, acting on all transistors by modulating threshold voltage and gain, and directly coupling with signal voltages, thus increasing the average delay of digital blocks. Switching noise has an especially detrimental effect on analog and mixed-signal circuits. In these circuits the presence of sensitive structures and large noise injectors on the same chip makes it imperative for the designer to accurately and efficiently estimate the strength of substrate noise at various locations. Evidently, CAD tools able to provide accurate injection and substrate transmission models are key to successful design flows. Furthermore, efficient substrate analysis, coupled with layout optimization, provides higher guarantees to reduce the design cycle, while ensuring satisfaction of tighter performance specifications.

Anisotropic substrates were first studied by Fukahori [2], who discretized the space into a resistive/capacitive mesh. DC/steady-state analysis was carried out by direct solution of the system of simultaneous thermal and electrical equations. Transient analysis was performed by using variable time-step trapezoidal integration techniques. In the dc analysis, direct LU factorization was later replaced with the incomplete Choleski conjugate gradient iterative method (ICCG). Transient analysis on the contrary was accelerated with frequency domain solutions such as asymptotic waveform evaluation (AWE) [3], [4].

Recently, attempts to introduce the effects of substrate in the design of medium-sized IC's have been made using numerical finite-difference methods (FDM's). These tech-

Manuscript received September 16, 1997; revised April 18, 1998. This paper was recommended by Associate Editor D. Ling.

E. Charbon is with Cadence Design Systems, Inc., San Jose, CA 95134 USA (e-mail: charbon@cadence.com).

R. Gharpurey is with Texas Instruments, Inc., Dallas, TX 75080 USA.

R. G. Meyer and A. Sangiovanni-Vincentelli are with the University of California at Berkeley, Berkeley, CA 94720 USA.

Publisher Item Identifier S 0278-0070(99)01006-4.

niques are versatile and general in nature, since they can handle lateral and vertical resistivity variations and arbitrary substrate geometries. However, to obtain accurate substrate characterization, a fine mesh is required, thus making storage and computational efforts often prohibitive. To overcome the formidable computational complexity of the problem, sparse nonuniform grids are often used. The grid size is made fine in areas close to substrate contacts and coarse in distant regions [5], [6]. The use of nonuniform or coarse grids, however, involves speed-accuracy tradeoffs, which are often difficult to evaluate *a priori*. Boundary element methods (BEM's) can also be used for parasitic and substrate extraction. In [7]–[9], the use of the Green's Function was proposed for a finite uniform medium and later for a multilayer substrate, with zero normal electric field boundary conditions, exploiting the technique of the separation of variables. Image-charge-based concepts have been used, in order to avoid the series computation involved in the method.

Traditionally, the analysis of substrate noise has been performed after the completion of physical design as a verification step. Experience has shown the extreme time complexity required to accurately model substrate and to estimate performance degradations due to switching noise. In many design problems, however, a *dynamic* substrate noise analysis would be preferable, since it could drive the design toward solutions more resilient to substrate noise. Recently, this problem was addressed by a number of authors who proposed heuristics to speed up substrate analysis during physical assembly phases, e.g., [6] and [10]. The approaches have in common the use of an FDM for the evaluation of the electric field on a coarse grid spanning the workspace, combined with AWE for an efficient solution of the resulting system of simultaneous algebraic equations. A potential problem with this approach is a strict requirement of alignment between grid and layout objects. Thus, unless specific tessellation [11] or analytical approximations [12], [13] are used, iterative algorithms based on progressive and often minimal modifications may not fully take advantage of the algorithms.

In this paper we propose a set of fast semi-analytical techniques for substrate analysis, which have been further accelerated for use within optimization loops. The algorithm at the heart of the substrate analysis package SUBRES, is a Green's Function-based BEM for multilayered substrates accelerated using the discrete cosine transform (DCT), which is efficiently computed via the fast Fourier transform (FFT). SUBRES generates a network accurately modeling contact-to-contact resistances in arbitrarily-shaped doping regions. The method can be further accelerated if accuracy can be traded off for circuit complexity. We show how upperbounds to accuracy degradation can be computed exactly.

Sensitivities of all the network components with respect to a number of technology parameters are computed using analytical manipulations of the Green's Function expressions and coded directly in the FFT, thus allowing fast evaluations on demand. Computing sensitivities of substrate coupling is useful for a number of reasons. First, it allows the evaluation of the impact of slight imperfections in the fabrication process on the circuit's performance and, ultimately, its yield. Second, it can be used as a quality factor for the selection of the best cost-effective technology on the basis of a class of circuits one wants to fabricate with given specifications. Furthermore, one can characterize the *trend* of circuit performance when engineering changes are performed on substrate geometry, technology parameters, or design. Third, the technique can be used during optimization to help the decision process providing guidance to the best possible improvement. Hence, the effects of technology migration/scaling can be carried out efficiently for a given chip without the need of performing a large number of complete substrate extractions. Fourth, sensitivities can be used to build performance models accounting for discrete parasitics as well as substrate effects. We show how these models can be efficiently built and used in demanding optimization algorithms at little computational cost.

The paper is organized as follows. In Section II substrate evaluation techniques based on a DCT accelerated BEM are described. Section III outlines the techniques used for sensitivity analysis and sensitivity-based optimization. A number of design optimization problems are presented in Section IV. In Section V the suitability of the approach is illustrated with a medium-sized mixed-signal IC designed using substrate-aware optimization and fabricated on a standard CMOS process.

II. Modeling Substrate Transport

A. Problem Formulation and Solutions

In general, silicon substrates are composed by one or more lightly doped epitaxial layers and a highly doped "core." Hence, differently conductive areas are present in the vertical section of the chip, while lateral resistivity variations are due to device and well implants as well as other integrated components. The latter are junctions with the substrate, and, in many cases, they may be considered equipotential. There are, however, situations in which this assumption is not adequate; in these cases it is advisable to partition such structures into separate contacts. Calculating resistances between any contact locations on the substrate requires the computation of electric potential $\Phi(x,y,z,t)$ at any location (x,y,z) in the bulk. From Maxwell's equations one can show that

$$\frac{1}{\rho}\nabla \bullet \nabla\Phi(x,y,z,t) + \epsilon\frac{\partial}{\partial t}\big(\nabla \bullet \nabla\Phi(x,y,z,t)\big) = 0 \qquad (1)$$

holds, where ϵ and ρ are, respectively, the local dielectric permittivity and resistivity of the substrate. In the electrostatic case, (1) reduces to the Laplace equation

$$\nabla^2\Phi = 0 \qquad (2)$$

with either Dirichlet or Neumann[1] boundary conditions or a combination of the two on the surfaces. Equation (2) can be solved using Green's Function-based BEM's [14].

Let $\Phi(\mathbf{r})$ be the potential at point $\mathbf{r} = (x,y,z)$ resulting from a localized charge density $\rho(\mathbf{r}')$, and $\Psi(\mathbf{r},\mathbf{r}')$ the potential at $\mathbf{r}$ due to a point charge placed at a point $\mathbf{r}'$. Then, $\Phi(\mathbf{r})$

[1] Dirichlet conditions impose a given potential, Neumann conditions a given electric field.

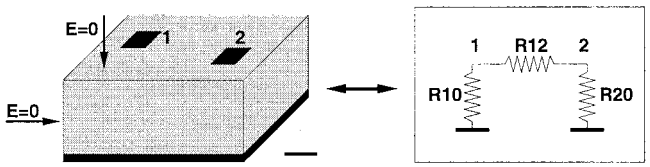


Fig. 1. Substrate boundaries and contact resistance modeling.

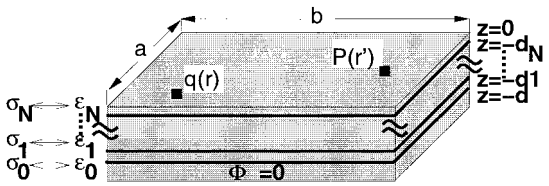


Fig. 2. Multilayer doping profiles.

can be expressed as

$$\Phi(\mathbf{r}) = \int_V \rho(\mathbf{r}')\Psi(\mathbf{r},\mathbf{r}')\,d^3\mathbf{r}' + \epsilon \oint_S \left(\Psi\frac{\partial\Phi}{\partial n} - \Phi\frac{\partial\Psi}{\partial n}\right) ds' \tag{3}$$

where $\cdot/\partial n$ symbolizes the derivative with respect to $\hat{\mathbf{n}}$, the unit outward normal vector to surface S enclosing volume V. $\Psi(\mathbf{r},\mathbf{r}')$ is called *Green's Function*. If the Green's Function is known, (3) allows one to determine the potential at any point in the volume V due to a known arbitrarily distributed charge density. Image-based techniques and the method of separation-of-variables (SOV) are two different approaches for evaluating the Green's Function. The methods are described in detail in [15, Ch. 3].

In the electrostatic case, the problem of computing the resistance between a substrate contact and all the others can be translated into that of computing the charge at the contact when set at a potential of 1 V, while the other contacts and the backplane are grounded. The reason for this is the following. Capacitance C_{ij} between contacts i and j is defined as the ratio of the charge on contact j to the potential of contact i, or $C_{ij} = Q_j/\Phi_i$. By Stokes Theorem

$$C_{ij} = \frac{1}{\epsilon}\oint_S E \bullet \hat{\mathbf{n}}\, ds \tag{4}$$

where $\hat{\mathbf{n}}$ is the unit outward normal vector to the surface S which encompasses the contact. E is the electric field in the medium. Similarly, the resistance between contacts is defined as

$$R_{ij} = Y_{ij}^{-1} = \left[-\sigma\oint_S E \bullet \hat{\mathbf{n}}\, ds\right]^{-1} = -\frac{1}{\sigma\epsilon}\frac{\Phi_i}{Q_j} \tag{5}$$

where σ is the medium conductivity. Note that in both the resistive and the capacitive cases the potential satisfies the Laplace equation, thus the problems can be interchanged freely.

At frequencies up to 4–5 GHz, substrate susceptance is typically much smaller than the conductance, hence it may be ignored and all substrate impedances may be considered real. Consider the problem of computing the resistance between contacts **1** and **2**, and toward ground in Fig. 1. This represents a mixed-boundary problem, since zero potential in the chip's backplane is assumed (Dirichlet condition) and vanishing normal electric field on the other faces (Neumann condition). Under these conditions, (3) simplifies to

$$\Phi(\mathbf{r}) = \int_V \rho(\mathbf{r}')G(\mathbf{r},\mathbf{r}')\,d^3\mathbf{r}' \tag{6}$$

where V is the chip's volume region and G the Green's Function. The potential of a contact is computed as the result of averaging all internal contact partitions. Hence, using (6), the potential of contact i can be derived as $\bar{\Phi}_i = \frac{1}{V_i}\int_{V_i}\int_{V_j}\rho_j G\,dv_j\,dv_i$, V_i and V_j being the volumes of contacts i and j and ρ_j the charge distribution on j. If a uniform charge distribution $\rho_j = Q_j/V_j$ is chosen over j, we obtain

$$\bar{\Phi}_i = \frac{Q_j}{V_j V_i}\int_{V_i}\int_{V_j} G\,dv_j\,dv_i. \tag{7}$$

The solution to (6) for each contact pair yields the *coefficient of potential* matrix $\mathbf{P}$. The relation between matrix $\mathbf{P}$ and vector $\bar{\mathbf{\Phi}}$, the average potential at each contact, and $\mathbf{Q}$, the charge associated with all contacts, is described as

$$\bar{\mathbf{\Phi}} = \mathbf{PQ} \quad \text{and} \quad \mathbf{Q} = \mathbf{c}\bar{\mathbf{\Phi}} \tag{8}$$

where $\mathbf{c} = \mathbf{P^{-1}}$ is called *coefficient of induction* matrix. For a contact i, the capacitance to ground C_i and all mutual capacitances C_{ij} are characterized as

$$C_i = \sum_{j=1}^{N} c_{ij}, \quad C_{ij} = -c_{ij} \tag{9}$$

where N is the size of matrix $\mathbf{c}$ [15]. Using (4) and (5) in combination with relations (9), all mutual and ground resistances can be easily derived.

B. Computing the Green's Function in Multilayered Substrates

The full derivation of the Green's Function for multilayered problems can be found in [15] and [16]. Here, we shall outline the basic steps to justify the sensitivity analysis and some optimization techniques proposed in this paper. Fig. 2 shows the multilayered structure for which a Green's Function must be computed. The figure shows for each layer i its conductivity σ_i and the permittivity ϵ_i associated with the equivalent electrostatic problem. Consider the case in which the point-charge at $\mathbf{r} = (x,y,0)$ and the observation point at $\mathbf{r}' = (x',y',0)$ are localized to a layer with dielectric permittivity ϵ_N. The Green's Function corresponds to an infinite series of sinusoidal functions

$$G(r,r') = G_0|_{z=z'=0} + \sum_{m=0}^{\infty}\sum_{n=0}^{\infty} f_{mn}C_{mn}\cos\left(\frac{m\pi x}{a}\right) \times \cos\left(\frac{m\pi x'}{a}\right)\cos\left(\frac{n\pi y}{b}\right)\cos\left(\frac{n\pi y'}{b}\right) \tag{10}$$

where $C_{mn} = 0$ for $m = n = 0$, $C_{mn} = 2$ for $m = 0$ or $n = 0$, but $m \neq n$, and $C_{mn} = 4$ for all $m, n > 0$. Parameters a, b, and d are the dimensions of substrate in x-, y-, and z-direction (see Fig. 2). Formulae for terms $G_0|_{z=z'=0}$ and f_{mn} can be found in [15] and [17].

From (7), adapted for surface contacts, one can derive an expression for the average potential at contact i due to the charge on contact j. Consequently, the entry p_{ij} of matrix $\mathbf{P}$, computed as the ratio of $\bar{\Phi}_i$ and Q_j, becomes

$$p_{ij} = \frac{\bar{\Phi}_i}{Q_j} = \frac{1}{S_i S_j} \int_{S_i} \int_{S_j} G(s_j, s_i)\, ds_j\, ds_i \qquad (11)$$

where S_j and S_i are the surfaces of the contacts.

Replacing (10) into (11) and integrating, one obtains an explicit formula for p_{ij}

$$p_{ij} = G_0|_{z=z'=0} + \sum_{m=0}^{\infty}\sum_{n=0}^{\infty} k_{mn} \times \frac{\left[\sin\left(m\pi\frac{a_2}{a}\right) - \sin\left(m\pi\frac{a_1}{a}\right)\right]\left[\sin\left(m\pi\frac{a_4}{a}\right) - \sin\left(m\pi\frac{a_3}{a}\right)\right]}{(a_2 - a_1)(a_4 - a_3)} \times \frac{\left[\sin\left(n\pi\frac{b_2}{b}\right) - \sin\left(m\pi\frac{b_1}{b}\right)\right]\left[\sin\left(m\pi\frac{b_4}{b}\right) - \sin\left(m\pi\frac{b_3}{b}\right)\right]}{(b_2 - b_1)(b_4 - b_3)} \qquad (12)$$

with

$$k_{mn} = \frac{a^2 b^2 f_{mn} C_{mn}}{m^2 n^2 \pi^4}.$$

Parameters (a_1, a_2) and (b_1, b_2) are the x- and y-coordinates of node i, and (a_3, a_4) and (b_3, b_4) those of node j. Appropriately rewriting the second term of (12), after proper scaling, as a cosine series we obtain

$$\sum_{m=0}^{\infty}\sum_{n=0}^{\infty} k_{mn} \cos\left(m\pi\frac{a_{1,2} \pm a_{3,4}}{a}\right) \cos\left(n\pi\frac{b_{1,2} \pm b_{3,4}}{b}\right) \qquad (13)$$

which is a compact representation of a sum of 64 terms forming all possible combinations of signs and indexes. By replacing the ratios of contact coordinates and the substrate dimensions with ratios of integers $\tilde{p}$, $\tilde{q}$ and summing over finite limits $\tilde{P}$ and $\tilde{Q}$, term (11) becomes

$$K(\tilde{p}, \tilde{q}) = \sum_{m=0}^{\tilde{P}-1}\sum_{n=0}^{\tilde{Q}-1} k_{mn} \cos\left(m\pi\frac{\tilde{p}}{\tilde{P}}\right) \cos\left(n\pi\frac{\tilde{q}}{\tilde{Q}}\right) \qquad (14)$$

a two-dimensional (2-D) DCT of k_{mn}. Hence, the computation of p_{ij} ultimately requires only a simple DCT [15], [16]. Several techniques exist for efficient computation of the DCT, e.g., FFT-based techniques only require a computation complexity $O(\tilde{P}\tilde{Q}\log_2 \tilde{P}\tilde{Q})$. Note that the value of k_{mn} is solely dependent on the properties of the substrate in z-direction. Hence, for a given substrate structure, the DCT needs be derived **only once**. Any modification in the relative position of one or more nodes is captured completely by the Fourier transform, thus only matrix $\mathbf{P}$ needs be calculated and inverted. However, due to the relatively small size of $\mathbf{P}$, typically 50–5000, this process does not require a significant CPU time. Nonabrupt doping profiles can be analyzed at low CPU cost by simply discretizing in z-direction with a gradually changing value of permittivity as shown in Fig. 3.

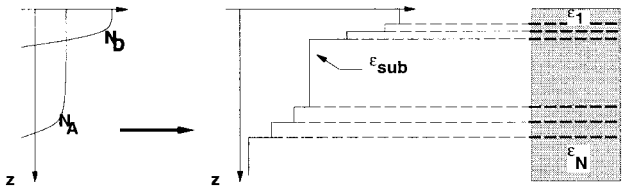


Fig. 3. Discretization of nonabrupt doping profiles.

C. Substrate Extraction Algorithm

The DCT of k_{mn} is computed for each location in a Manhattan grid covering the whole substrate surface. To generate matrix $\mathbf{P}$, it is necessary to compute the parameter p_{ij} for all the pairs of partition elements composing each contact. If no scheme is used for its reduction (see Section II-D), the size of $\mathbf{P}$ and $\mathbf{c}$ is

$$|\mathbf{P}| = N = \sum_{i=1}^{N_c} \mathcal{P}_i$$

where N_c is the total number of contacts and $\mathcal{P}_i$ the number of partitions in contact i.

Due to the dense nature of $\mathbf{P}$, the inversion is the most time-consuming operation of the whole algorithm. Several inversion techniques, both direct and iterative, have been implemented by us and in [15]. Among the direct methods, an LU decomposition-based algorithm of complexity $O(N^3)$ has been used for relatively small configurations of less than approximately 1000 partitions. Larger circuits required the use of various accuracy-driven simplification schemes.

After the computation of $\mathbf{c}$, the actual resistive or conductive networks $\mathbf{R}$ and $\mathbf{Y}$ are calculated. $\mathbf{R}$ and $\mathbf{Y}$ are $N_c \times N_c$ matrices. Assuming appropriate scaling of $\mathbf{c}$, one can easily show that a direct relation exists between $\mathbf{c}$ and $\mathbf{Y}$ via mapping $\mathbf{X}$

$$\mathbf{Y} = \mathbf{X^T c X}, \quad \text{with} \quad \mathbf{X} = \left[\mathbf{e_1}, \mathbf{e_2}, \cdots, \mathbf{e_{N_c}}\right] \qquad (15)$$

where $\mathbf{e_i} = [0, \cdots, 0, 1, \cdots, 1, 0, \cdots, 0]^T$. The indexes of the nonzero entries of $\mathbf{e_i}$ are associated with the contact. Due to the structure of $\mathbf{X}$, (15) only involves $(N-1)^2$ summations. The elements of matrix $\mathbf{R}$ are computed simply using the relation $R_{ij} = 1/Y_{ij}$. For simplicity but without loss of generality, in what follows we will assume that $N_c = N$, i.e., every contact is modeled in terms of one partition.

D. Schemes for Efficient Solution of Large Substrate Problems

Many authors who have dealt with the substrate problem have also proposed methods for the reduction of its size to improve the overall computation efficiency and to reduce the mesh of extracted parasitics. A classical approach consists of creating *active extraction windows* around each contact encompassing all the structures which are **not** ignored in the computation of the resistive network associated with the contact. However, it is not clear how inaccuracies can be bounded by a particular selection of window size and shape.

In [16] an alternative method was proposed. The aim of the method is to make matrix $\mathbf{P}$ sparse, with bounded accuracy reduction. Consider the scenario depicted in Fig. 4. n contacts

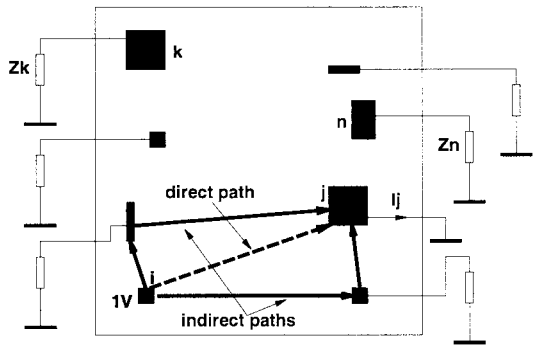


Fig. 4. Direct and indirect current-flow paths.

are laid out on the substrate surface. Each contact k is loaded with impedance Z_k. Suppose that contact j is grounded while i is at 1 V and that the ratio $Y_{ij} = I_j/V_i$ is sought. If $Z_k = 0$, $\forall k \neq i$, then the direct path dominates in the computation of Y_{ij}, thus it cannot be ignored. On the contrary, when $Z_k \neq 0$, for some $k \neq i$, then a bound on the conductances associated with k can be derived for which the direct path can be ignored, without violating predetermined accuracy constraints.

Let matrix $\mathbf{P}$ be known. By (9) and (4) $\mathbf{Y}$, i.e., the matrix which relates the voltages of all contacts to the currents flowing out of them, can be easily derived. Let $\mathbf{Y}_L$ be the vector of the load admittances at each contact, for simplicity assume that Y_L is purely resistive, i.e., $\mathrm{Imag}\{Y_L\} = 0$. Equation (16) represents the effects of loading on the circuit

$$[\mathbf{Y} + \mathbf{diag}(\mathbf{Y}_L)]V = I. \tag{16}$$

V and I are the vectors of contact potentials and currents, respectively. One can show that if condition

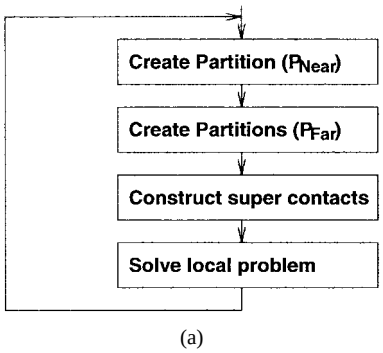
$$\left| \sum_{k=1}^{N} Y_{jk} V_k \right|_{k \neq i, k \neq j} > |Y_{ji}| V_0 \tag{17}$$

is met, component Y_{ji} can be set to zero. Note that V_0 is normalized to 1 V. For each component of $\mathbf{Y}$ set to zero, a precise value can be computed for the lost accuracy of all currents I_k [15]. Hence the process can be applied until the cumulative inaccuracies reach a predetermined value.

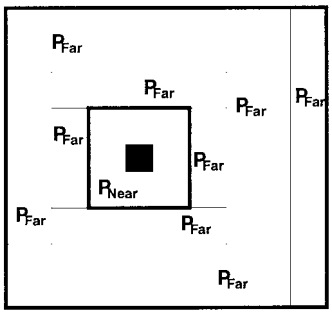
The procedure is most suited for an iterative solution scheme. Fig. 5(a) shows the complete extraction scheme. Consider each contact separately, call it *subject*. First, a partition P_{Near} is defined around the subject containing all such contacts which **do not** satisfy (17). Second, larger partitions P_{Far}, containing the remaining contacts, are created with geometrically increasing size.[2] See Fig. 5(b).

Third, all contacts contained in each partition P_{Far} are replaced by a single contact, or *super-contact*, placed in the center of the partition and with an area equal to the sum of the areas of all the original contacts. The location and area of

[2]The formula for the computation of the size is the following: $d_{k+1} = \alpha(d_k)$, where $\alpha = 5$ in our prototype. Since in these partitions all the contacts satisfy (17), the growth criterion does not affect the accuracy in any way and it was chosen so as to facilitate partition computations.



(a)



(b)

Fig. 5. (a) Simplified substrate extraction scheme. (b) Partitioning of substrate.

the super-contact are

$$x_{\mathrm{sc}} = \frac{x_{\mathrm{left}} + x_{\mathrm{right}}}{2}, \quad y_{\mathrm{sc}} = \frac{y_{\mathrm{left}} + y_{\mathrm{right}}}{2},$$
$$A_{\mathrm{sc}} = \sum_{k \in P_{\mathrm{Far}}} A_k \tag{18}$$

where $x_{\mathrm{left/right}}$ and $y_{\mathrm{left/right}}$ are the left/right x- and y-coordinates of the partition boundary, respectively. Fourth, a simplified equation is derived for the subject j. The original equation is replaced as follows:

$$Y_{j\cdot} \bullet V = I_j \rightarrow \tilde{Y}_{j\cdot} \bullet \tilde{V} = I_j \tag{19}$$

where $Y_{j\cdot}$ denotes the jth row of matrix $\mathbf{Y}$, $\tilde{Y}_{j\cdot}$ is a vector of size $s < |Y|$ resulting from replacing the required number of contacts by super-contacts. $\tilde{V}$ is the vector of the potentials on the remaining contacts and super-contacts. Potential vector $\tilde{V}$ is evaluated and the iteration proceeds to the next subject. The algorithm terminates when every contact has been considered.

To obtain some bounds on the maximum attainable simplification rate, consider the following extreme cases: 1) every contact satisfies (17); 2) no contact satisfies (17). In case 1), $P_{\mathrm{Near}} = \emptyset$, hence only super-contacts exist and the size of matrix $\mathbf{Y}$ is reduced by one or the size of $\mathbf{P}$ is decreased by the number of partitions internal to the subject. In 2), matrix $\mathbf{Y}$ is not simplified, and neither is $\mathbf{P}$, hence the complexity of the problem remains that of inverting $\mathbf{P}$. However, this is generally not the case in real substrate problems, where complexity reduction in schematics is typically a factor of ten [15].

III. Substrate-Aware Optimization

In this paper we often refer to low- and high-resistivity substrate. The distinction is necessary for two reasons. First, these

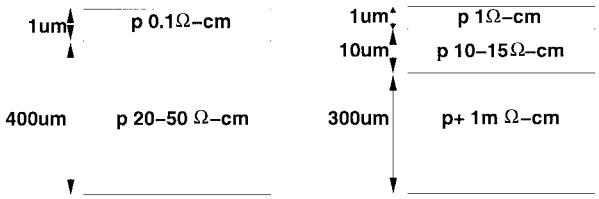


Fig. 6. Typical IC substrates: (a) high-resistivity and (b) low-resistivity.

substrate types are used mainly in BiCMOS and CMOS applications, respectively. Second, switching noise transport mechanisms are substantially different in the two substrate types, thus resulting in different design guidelines to obtain isolation. Typical substrate implementations generally used in IC fabrication are shown in Fig. 6. Injection of switching currents into substrate follow similar mechanisms. For a full description of injection and reception mechanism, see [15] and [17].

A. Sensitivity Analysis

The relation between circuit performance K and technology, via substrate-related parasitics, is obtained using the following expression:

$$\Delta K = \sum_{\ell} \frac{\partial K}{\partial T_\ell} \Delta T_\ell, \quad \text{with} \quad \frac{\partial K}{\partial T_\ell} = \sum_{i,j} \frac{\partial K}{\partial Y_{ij}} \frac{\partial Y_{ij}}{\partial T_\ell} \tag{20}$$

where (i, j) represents a contact pair, Y_{ij} the substrate conductive coupling between i and j, and T_ℓ a technology parameter. Hence, assuming $\frac{\partial K}{\partial Y_{ij}}$ exists,[3] ΔK can be easily evaluated as a linear function of technology parameters T_ℓ, provided that term $\frac{\partial Y_{ij}}{\partial T_\ell}$ has been computed. Assume that the capacitive problem has been solved and that the equivalent resistive network has been computed from the coefficient of induction matrix $\mathbf{c}$. Furthermore, let $\mathbf{c}$ be scaled in such a way that the node-to-node conductance Y_{ij} and the ground conductance Y_{ii} can be computed directly using

$$Y_{ii} = \sum_{j=1}^{N} c_{ij}, \quad Y_{ij} = -c_{ij}. \tag{21}$$

Let us define $\mathbf{Y}$ as an $N \times N$ matrix consisting of Y_{ii} on the diagonal and Y_{ij} everywhere else. Let us call $\partial \mathbf{Y}/\partial T_\ell$ the sensitivity of matrix $\mathbf{Y}$ with respect to technology parameter T_ℓ. The components of the sensitivity matrix are terms $\partial Y_{ii}/\partial T_\ell$ on the diagonal and $\partial Y_{ij}/\partial T_\ell$ everywhere else. The terms are computed using

$$\frac{\partial Y_{ii}}{\partial T_\ell} = \sum_{j=1}^{N} \frac{\partial c_{ij}}{\partial T_\ell} \quad \text{and} \quad \frac{\partial Y_{ij}}{\partial T_\ell} = -\frac{\partial c_{ij}}{\partial T_\ell}. \tag{22}$$

Recall that N is the size of matrix $\mathbf{c}$. In order to derive $\partial c_{ij}/\partial T_\ell$, (8) is differentiated on both hand-sides and solved with respect to $\partial \mathbf{Q}/\partial T_\ell$. Using the fact that $\partial \bar{\mathbf{\Phi}}/\partial T_\ell$ vanishes, we obtain

$$\frac{\partial \mathbf{Q}}{\partial T_\ell} = -\mathbf{P}^{-1}\left(\frac{\partial \mathbf{P}}{\partial T_\ell}\mathbf{Q}\right). \tag{23}$$

[3]This term can be computed numerically in an efficient manner, during circuit simulation.

Using the definition of c_{ij}

$$\frac{\partial c_{ij}}{\partial T_\ell} = \frac{1}{\Phi_j}\frac{\partial Q_i}{\partial T_\ell} \tag{24}$$

where $\partial Q_i/\partial T_\ell$ is computed using (23). Now, only the derivative $\partial \mathbf{P}/\partial T_\ell$, i.e., $[\partial p_{ij}/\partial T_\ell]_{i,j=1,\cdots,N}$, remains to be computed. From (12), assuming zero-depth contacts and $T_\ell \neq \epsilon_N, d, a$ or b

$$\begin{aligned}\frac{\partial p_{ij}}{\partial T_\ell} &= \frac{\dot{\Gamma}_N \beta_N - \Gamma_N \dot{\beta}_N}{ab\epsilon_N {\beta_N}^2} + \sum_{m=0}^{\infty}\sum_{n=0}^{\infty} \dot{k}_{mn} \\ &\times \frac{\left[\sin\left(m\pi\frac{a_2}{a}\right) - \sin\left(m\pi\frac{a_1}{a}\right)\right]\left[\sin\left(m\pi\frac{a_4}{a}\right) - \sin\left(m\pi\frac{a_3}{a}\right)\right]}{(a_2 - a_1)(a_4 - a_3)} \\ &\times \frac{\left[\sin\left(n\pi\frac{b_2}{b}\right) - \sin\left(m\pi\frac{b_1}{b}\right)\right]\left[\sin\left(m\pi\frac{b_4}{b}\right) - \sin\left(m\pi\frac{b_3}{b}\right)\right]}{(b_2 - b_1)(b_4 - b_3)}\end{aligned} \tag{25}$$

where $\dot{\Gamma}_N = \partial \Gamma_N/\partial T_\ell$, $\dot{\beta}_N = \partial \beta_N/\partial T_\ell$, and $\dot{k}_{mn} = \partial k_{mn}/\partial T_\ell$. Expressions for Γ_N, β_N, k_{mn} for all-depth contacts have been derived in [15]. The calculation of the derivatives can be found in the Appendix.

The first term of (25) can be easily calculated from the formulae in the Appendix, while the second term can be efficiently computed using the DCT by replacing k_{mn} with $\dot{k}_{mn}$ in (14). The DCT can be computed for each location in the grid and repeated for all parameters $T_\ell, \ell = 1, \cdots, N_T$, where N_T is the number of technology parameters considered. Notice that this calculation need be performed **only once** for a given substrate structure.

To generate matrices $\partial \mathbf{c}/\partial T_\ell$ and $\partial \mathbf{P}/\partial T_\ell$, it is necessary to compute sensitivities $\partial p_{ij}/\partial T_\ell$ and $\partial c_{ij}/\partial T_\ell$ for all pairs of partition elements composing each contact. Every sensitivity measure requires additional $N \times N$ storage. As an example, assume $N_T = 10$, i.e., ten technology parameters T_ℓ are considered; moreover, assume that a grid of 1024 $\times$ 1024 points is used. Then the total storage needed by our approach is 41.9 MByte, which is relatively low considering that a 1-μm resolution would be achieved on a 1 $\times$ 1 mm chip size.

B. Constraint Generation for Substrate Parasitic Effects

Constraint generation in a strict sense requires that parasitics be entities associated with one or more physical structures of the layout. In the case of switching noise, the physical location and transmissions paths through the substrate may not be known before floorplanning. For this reason, the constraint generation process cannot take place before the layout is, at least in part, generated, i.e., when constraints are mostly needed. To address this issue we introduce the concept of *local noise generators*. A local noise generator is defined as a voltage or current source producing the equivalent of the cumulative noise contributed to by the real noise generators located in the substrate. The generator should simulate as closely as possible the waveform felt at an arbitrary location, including distortions, attenuations, and group delays which transformed the original noise signal.

Consider *sensing node* n (see Fig. 7). Let us call $g_n(t, \mathbf{\Pi})$ a waveform felt at n, where t is the time and $\mathbf{\Pi}$ is a vector of all

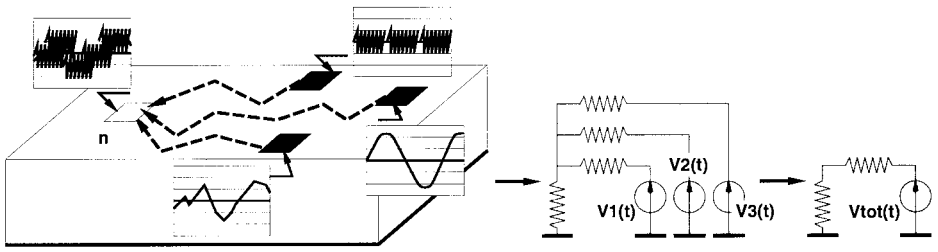


Fig. 7. The principle and modeling of local generators.

the parameters relevant to it. Let us define G_n as a local noise generator producing waveform g_n. Due to the diverse nature of its parameters, $\mathbf{\Pi}$ can be split into its basic components $\mathbf{\Pi} = [\mathbf{W}^T \ \mathbf{G}^T \ T \ V_0]^T$. $\mathbf{W}$ represents process-dependent and $\mathbf{G}$ layout-related parameters, T is the temperature, and V_0 the local substrate potential. One can also define vector $\Delta\mathbf{\Pi} = [\Delta\mathbf{W}^T \ \Delta\mathbf{G}^T \ \Delta T \ \Delta V_0]^T$ as the variation of $\mathbf{\Pi}$ from nominal. Consider performance measure K_i, its degradation from nominal is given by the product of the ith row of sensitivity matrix $\mathbf{S}_{\mathbf{\Pi}}$ and vector $\mathbf{\Delta\Pi}$

$$\Delta K_i = (\mathbf{S}_{i,\mathbf{\Pi}})^T \mathbf{\Delta\Pi} \tag{26}$$

where vector $\mathbf{S}_{i,\mathbf{\Pi}} = [\mathbf{S}_{i,\mathbf{W}} \ \mathbf{S}_{i,\mathbf{G}} \ \mathbf{S}_{i,T} \ \mathbf{S}_{i,V_0}]^T$ represents the sensitivity of K_i with respect to all the parameters of interest. Suppose now that the exact waveform felt at n is not available, and only an estimate can be derived. Moreover, suppose that a range can be set for $\mathbf{\Pi}$ such as $\mathbf{\Pi}^{(\min)} \leq \mathbf{\Pi} \leq \mathbf{\Pi}^{(\max)}$. Assuming that $\mathbf{S}_{i,\mathbf{\Pi}}$ exists and has been computed, bounds on all parameter variations $\Delta\mathbf{\Pi}^{(\text{bound})}$ can be calculated using, for example, constrained optimization as shown in [18]. Hence, the amount of noise at the sensing nodes can be constrained *a priori*, without a precise knowledge of the structure of the layout being built.

Let us now generalize the problem by considering a large number of sensing nodes. From a theoretical standpoint, at each receptor a different waveform could be felt. However, since the size of the analog section of a mixed-signal circuit is small compared to the distance to the noise sources, it is assumed that all the substrate nodes are reached by an identical waveform at different times. Suppose M sensing nodes exist, each of them connected to a local generator $G_m(t - \tau_m, \mathbf{\Pi}_m)$, with $m = 1, \cdots, M$, where τ_m is the propagation delay between nodes. Due to the highly nonlinear dependence of performance on phase, an additive linearization around a nominal value could inaccurately model the parasitic effects of substrate.

The problem can be effectively addressed by deriving a set of worst-case sensitivities as described in [19]. Call $\mathbf{\Pi}'$ the array of all design parameters for which K_i is not strongly nonlinear and $\bar{\mathbf{S}}$ the corresponding sensitivity matrix. Hence, the total linearized worst-case variation of K_i, due to node m, is derived as

$$\Delta K_i|_m = \left(\bar{\mathbf{S}}_{i,\mathbf{\Pi}'_m}\right)^T \mathbf{\Delta\Pi}'_m. \tag{27}$$

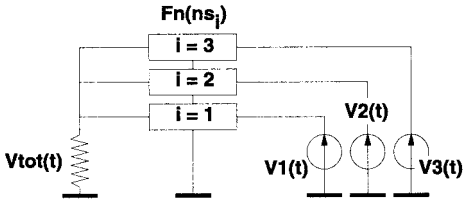


Fig. 8. Evaluation of constraint violations.

Using the same formalism of (26) and considering all the sensing nodes m in the circuit, we can define the matrices

$$\bar{\mathbf{S}}_{i,\mathbf{\Pi}'} = \begin{pmatrix} \left(\bar{\mathbf{S}}_{i,\mathbf{\Pi}'_0}\right)^T \\ \vdots \\ \left(\bar{\mathbf{S}}_{i,\mathbf{\Pi}'_{M-1}}\right)^T \end{pmatrix} \quad \text{and} \quad \mathbf{\Delta\Pi}' = \begin{pmatrix} (\mathbf{\Delta\Pi}'_0)^T \\ \vdots \\ (\mathbf{\Delta\Pi}'_{M-1})^T \end{pmatrix}. \tag{28}$$

Thus, the degradation of performance K_i is expressed as

$$\Delta K_i = \mathbf{trace}(\bar{\mathbf{S}}_{i,\mathbf{\Pi}'} \Delta\mathbf{\Pi}'). \tag{29}$$

Equation (29) models the contributions of all sensing nodes onto performance K_i. Bounds on the parameters associated with each sensing node $\mathbf{\Delta\Pi}_n^{(\text{bound})}$ can be computed using constrained optimization provided that conservative upper and lower bounds on the realization of $\mathbf{\Pi}$ are also available for each sensing node n.

The use of worst-case sensitivity matrix $\bar{\mathbf{S}}_{i,\mathbf{\Pi}'}$ has the advantage of reducing the parameter space of $\mathbf{\Pi}$. Moreover, nonlinear behavior in a certain range of performance can be accurately modeled.

Due to the mechanism of noise modeling obtained using local generators, constraints on noise parameters can be derived independently of a particular IC process. Hence, constraint generation is required **only once** for a given circuit. During physical assembly, process-dependent substrate extraction, in combination with estimates of the sources of switching noise, is used to enforce the bounds. Furthermore, the effect of substrate noise can be evaluated locally, taking into consideration neither the exact floorplan nor the actual position of the noise sources. Once the substrate has been extracted, a transfer function $F_n(\text{ns}_i)$ can be computed relating each noise source ns_i to receptor n. Assuming that approximations or exact waveforms are known for each noise source, waveform $g_n(t, \mathbf{\Pi}_n)$ and the corresponding parameter $\mathbf{\Pi}_n$ can be easily evaluated for each node n. Thus a simple check can be performed to verify that constraints $\mathbf{\Delta\Pi}_n^{(\text{bound})}$ and consequently the original specifications have been met (see Fig. 8).

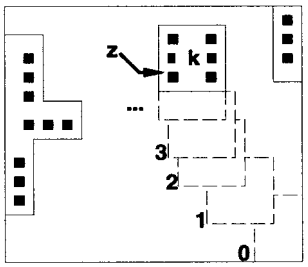


Fig. 9. Sensitivity of resistive macro-model from transformation of a componenet and its contacts.

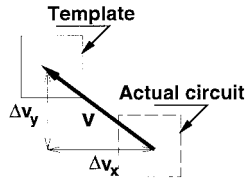


Fig. 10. Computation of update matrix δc based on contact displacement relative to template.

C. Substrate Transport Evaluation in Iterative Algorithms

Due to its "global" effects felt everywhere in the chip, substrate noise cannot be translated into a compact analytical model accounting for the entire substrate area. Hence, even if a small incremental modification is performed on the chip, the whole substrate analysis needs to be reevaluated. Unlike traditional approaches based on FDM's, SUBRES exploits the locality of incremental changes avoiding the resolution of the entire substrate at each optimization iteration. The techniques proposed hereafter are designed for very fast estimation of *variations* and *trends* within computationally expensive algorithms.

The first technique exploits the fact that small adjustments in the configuration of layout elements results in a small change in the coefficient of potential matrix $\mathbf{P}$. Let $\mathbf{P'}$ be the potential matrix associated with the new configuration. Note that in $\mathbf{P'}$, only row r and column c will differ from $\mathbf{P}$. Let $\delta\mathbf{P}_{r.}$ be the rth row and $\delta\mathbf{P}_{.c}$ the cth column of $\mathbf{P'}$, then $\mathbf{P'} = \mathbf{P} + \delta\mathbf{P}_{.c} + \delta\mathbf{P}_{r.}$. For simplicity, consider only the modification due to $\delta\mathbf{P}_{r.}$. Using the Sherman–Morrison formula, $\mathbf{P'^{-1}}$ can be computed directly as

$$\mathbf{c'} = \mathbf{P'^{-1}} = \mathbf{c} + \delta\mathbf{c}, \quad \text{with} \quad \delta\mathbf{c} = -\frac{\mathbf{c}_{.r}(\mathbf{c}\delta\mathbf{P}_{r.})}{1 + \delta\mathbf{P}_{r.}\mathbf{c}_{.r}} \tag{30}$$

where $\mathbf{c}_{.r}$ is the rth column of $\mathbf{c}$. The computation of the entire resistive network is dominated by the Sherman–Morrison update, completed in $O(N^2)$ time for each contact partition being moved.

The second technique, known as *Gradient Based Method*, is based on the concept of *sensitivity to relocation*. Suppose that a contact or a collection of contacts z is to be relocated on the substrate surface from location $\mathbf{x}_0$ to $\mathbf{x}_k$ going through intermediate locations $\mathbf{x}_1, \cdots, \mathbf{x}_{k-1}$ (see Fig. 9). One can easily show that

$$[\mathbf{c}]_k = [\mathbf{c}]_0 + \sum_{n=1}^{k} [\delta\mathbf{c}]_n$$

where $[\mathbf{c}]_0$ is the coefficient of induction matrix associated with location $\mathbf{x}_0$, and $[\delta\mathbf{c}]_{n+1} = [\mathbf{c}]_{n+1} - [\mathbf{c}]_n$ is the $(n+1)$th update of $\mathbf{c}$. The updates $[\delta\mathbf{c}]_{n+1}$ can be computed using the Sherman–Morrison formula in $O(N^2)$ time.

To further speed up the computation, one can exploit the "gradient" information of resistive and conductive networks $\mathbf{R}$ and $\mathbf{Y}$, contained in $[\delta\mathbf{c}]_1$. Assume that a single contact z is relocated in direction $\mathbf{v}$ by an amount $|\mathbf{v}| \to 0$. Let us define the vector $\nabla_{\mathbf{v}}\mathbf{Y}$ to be

$$\nabla_{\mathbf{v}}\mathbf{Y} = [\mathbf{A} \quad \mathbf{B}]^T$$

where $\mathbf{A} = \partial\mathbf{Y}/\partial v_x$, $\mathbf{B} = \partial\mathbf{Y}/\partial v_y$, and $\mathbf{v} = [v_x v_y]^T$. The components of matrix $\mathbf{A}$ are defined as $A_{ij} = \partial\mathbf{Y}_{ij}/\partial\mathbf{v}_x$, those of $\mathbf{B}$ as $B_{ij} = \partial\mathbf{Y}_{ij}/\partial\mathbf{v}_y$. Recall that, since $N_c = N$, Y_{ij} is defined as the mutual conductance between contact partitions i and j for a given substrate configuration and that Y_{ii} is the ground conductance of i. The minimum step size in x- and y-direction corresponds to a unit of the grid of the DCT. Hence, matrix $\partial\mathbf{Y}/\partial\mathbf{v_x}$ can be approximated by first computing differences $\delta p_{i,j\pm1}$ and $\delta p_{i\pm1,j}$ using

$$\delta p_{i,j\pm1} = p_{i,j\pm1} - p_{i,j}, \quad \text{and} \quad \delta p_{i\pm1,j} = p_{i\pm1,j} - p_{i,j}. \tag{31}$$

Then, each component $\partial Y_{ij}/\partial v_x$ is calculated by replacing term c_{ij} with $\delta c_{i,j+1}$ in (4), (5), (8), and (9). Notice that term $\delta c_{i,j+1}$ is derived directly from matrix $\mathbf{c}$ and $\delta p_{i,j+1}$ using the Sherman–Morrison formula. Moreover, the direct replacement of c_{ij} in the equations is legitimated by the fact that all manipulations are linear. The same method is used to derive $\partial\mathbf{Y}/\partial v_y$. The time complexity of the operation is $O(N^2)$ since the Sherman–Morrison formula needs to be repeated for all the contacts or partitions involved in the move.

Let us assume that $\partial\mathbf{Y}/\partial v_x$ and $\partial\mathbf{Y}/\partial v_y$ have been computed at the zeroth step of our incremental algorithm. Call $[\partial\mathbf{Y}/\partial v_x]_0$ and $[\partial\mathbf{Y}/\partial v_y]_0$ these matrices. Assuming that the moving partition, contact, or collection of contacts remains *close enough* to its position at step 0, then the conductance matrix at steps $1 \le n \le k$ can be approximated as

$$\begin{aligned}[\mathbf{Y}]_n &\approx [\mathbf{Y}]_0 + \left[\frac{\partial\mathbf{Y}}{\partial v_x}\right]_0 \Delta v_x + \left[\frac{\partial\mathbf{Y}}{\partial v_y}\right]_0 \Delta v_y \\ &= [\mathbf{Y}]_0 + \left[\nabla_{\mathbf{v}}\mathbf{Y^T}\right]_0 \mathbf{\Delta v}\end{aligned} \tag{32}$$

where $\mathbf{\Delta v} = [\Delta v_x, \Delta v_y]^T$ is the vector representing the move of contact or partition z from step 0 to n.

The Green's Function and its DCT are well behaved functions everywhere in the workspace [15]. Hence, necessarily terms $\delta p_{i,j\pm1} < \infty$ and $\delta p_{i\pm1,j} < \infty$. No "high-frequency" components are present in the function, making it an ideal candidate for a highly accurate use of a gradient-based method. In fact, in our experiments the method has shown a 1% accuracy when the move occurred in the vicinity (less than five steps away) of the position at step 0, while a 10% accuracy was reached when the move was up to one-tenth of the chip size.

D. Template-Based Substrate Extraction

In Section II a technique was presented to speed-up the extraction process and to simplify the schematic based on the

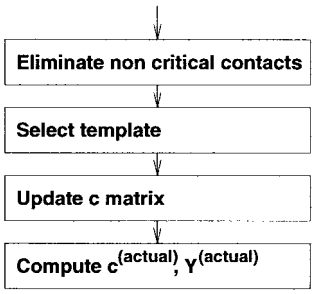


Fig. 11. Block diagram of the template-based substrate extraction algorithm.

knowledge of contact loading. In this section we discuss a method for further reduction of the extraction time of large circuits that share a set of recurring contact patterns.

Fig. 11 illustrates the technique through a block diagram. First, a set of templates with N_c or more contacts, for which an extracted schematic exists, is compared to the sample layout. Among the available ones, a template is selected and its precomputed coefficient of induction matrix $\mathbf{c}^{(\mathbf{template})}$ is used to compute $\mathbf{c}^{(\mathbf{actual})}$, the matrix associated with the actual circuit. Each progressive update matrix $\delta\mathbf{c}$ is computed based on the displacement $\mathbf{v} = [\Delta v_x \ \Delta v_y]^T$ of each contact nonoverlapping exactly with a corresponding contact in the template, as shown in Fig. 10. Finally, the partial conductance matrix $\mathbf{Y}^{(\mathbf{actual})}$ is computed directly from $\mathbf{c}^{(\mathbf{actual})}$ using (15)

$$\mathbf{Y}^{(\mathbf{actual})} = \mathbf{X}^{\mathbf{T}}\mathbf{c}^{(\mathbf{actual})}\mathbf{X}. \tag{33}$$

Fig. 12(A) shows an example of physical layout being extracted. The template selected for this circuit is shown in Fig. 12(B). The procedure of eliminating and aligning some of the contacts of the template onto the actual circuit is shown in Fig. 12(C). In order to derive bounds on the time complexity of the procedure, consider the following cases. First, assume the worst-case scenario, i.e., no contact exists which overlaps exactly with a contact in the template. In this case, N updates are needed for complete substrate evaluation, the resulting complexity is therefore $O(N^3)$. This case is equivalent to a full inversion of matrix $\mathbf{P}$, hence no improvement is achieved over the nonsimplified substrate extraction. Second, consider the case in which the sample and the template are identical. In this case no computation is needed, hence the extraction complexity is zero. The second scenario, or one as near as possible to it, is most desirable.

Since the complexity of computing an update of matrix $\mathbf{c}$ is independent of the transformation involved, an effective criterion for selecting the template is one aimed at maximizing N_o, the number of contacts exactly overlapping a contact in the actual circuit layout. Consequently, assuming that $N_o < N$ contacts differ in location from corresponding contacts of a template, the complexity of the procedure could be a fraction of that needed to invert $\mathbf{P}$.

In real circuits, however, a large number of contacts rarely overlaps to those on the template. To cope with this problem, we propose a criterion based on performance sensitivities for the template selection and the minimization of updates needed for full extraction given predefined accuracy constraints. The

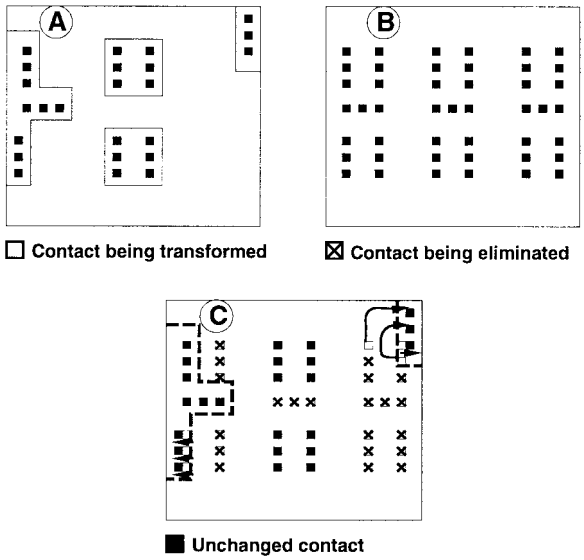


Fig. 12. Speed-up mechanism for the extraction of large substrates.

modified template-based substrate extraction algorithm is described in Fig. 14. For simplicity but without loss of generality, let us consider only one performance function K. Assume that the matrix $S_{Y_{ij}}$ of the sensitivities of K with respect to all partial conductances Y_{ij} has been computed or estimated. Moreover, assume that estimates exist for the maximum values of all substrate conductances.[4] Using a fraction of the specified maximum degradation of K as threshold, all conductances, whose cumulative effect on performance is lower than the threshold, are eliminated from the schematic. All nodes connected to one or zero conductances are also eliminated as illustrated in Fig. 13. The resulting substrate configuration must be then compared with a set of templates and the best template must be selected. This problem is solved using optimization. A by-product of the selection procedure is the set D of all contacts that need be extracted in all details. The displacements of the contacts in D, relative to the selected template, are identified, and updates needed for the computation of $\mathbf{c}^{(\mathbf{actual})}$ are computed using the Sherman–Morrison formula. Partial conductance matrix $\mathbf{Y}^{(\mathbf{actual})}$ is finally derived directly from $\mathbf{c}^{(\mathbf{actual})}$ using (33).

Hereafter, the template selection procedure is illustrated. Let us consider matrix update $\delta\mathbf{c}|_i$ representing the move of contact i from its location in the template to that of the actual circuit. The coefficient of induction matrix $\mathbf{c}^{(\mathbf{actual})}$ associated with the actual circuit is computed as

$$\mathbf{c}^{(\mathbf{actual})} = \mathbf{c}^{(\mathbf{template})} + \sum_{i \in D} \delta\mathbf{c}\Big|_i \tag{34}$$

where D is the set of all the contacts whose locations in the template and in the actual circuit are nonidentical and hence need to be extracted in full detail. Combining (34) and (33),

[4]Rough estimates of the maximum/minimum value of substrate conductances can be easily computed from a simple set-up of two contacts located at chip edges or in close proximity.

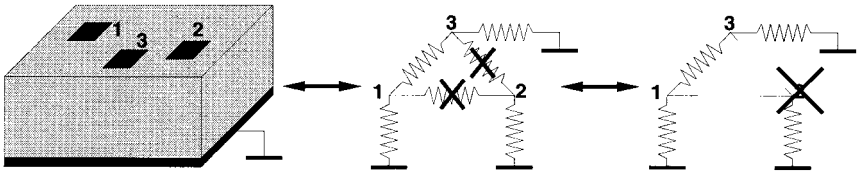


Fig. 13. Elimination of all noncritical conductances and contacts.

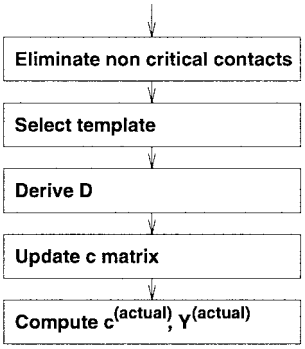


Fig. 14. Block diagram of the modified template-based substrate extraction algorithm.

one obtains

$$\mathbf{Y^{(actual)}} = \mathbf{Y^{(template)}} + \sum_{i \in D} \mathbf{X^T} \delta \mathbf{c}\Big|_i \mathbf{X} \tag{35}$$

where $\mathbf{Y^{(template)}} = \mathbf{X^T c^{(template)} X}$ is the precomputed partial conductance matrix of the template. Let us define the error matrix, i.e., the update needed to translate $\mathbf{Y^{(template)}}$ into $\mathbf{Y^{(actual)}}$, as

$$\epsilon = \mathbf{Y^{(actual)}} - \mathbf{Y^{(template)}} = \sum_{i \in D} \epsilon\Big|_i \tag{36}$$

where $\epsilon|_i = \mathbf{X^T} \delta \mathbf{c}|_i \mathbf{X}$ is the error matrix due to the displacement of contact i in the actual circuit relative to the template.[5] Assume one could calculate $\epsilon|_i$, $\forall i \in D$ *a priori*. Using the sensitivity[6] of performance K with respect to matrix $\mathbf{Y}$, performance degradation ΔK due to the displacement of contacts in the actual circuit relative to the template can be calculated as

$$\Delta K = \mathbf{e}^T \left(\frac{\partial K}{\partial Y_{ij}} \odot \epsilon \right) \mathbf{e} \tag{37}$$

where $\mathbf{e}$ is an $N \times 1$ unity vector such that $\mathbf{e} = [1, \cdots, 1]^T$. The $\odot$ operator is defined as follows: $\mathbf{A} = \mathbf{B} \odot \mathbf{C} \Leftrightarrow a_{ij} = b_{ij} c_{ij}$. Combining (36) and (37), one obtains

$$\Delta K = \sum_{i \in D} \mathbf{e}^T \left(\frac{\partial K}{\partial Y_{ij}} \odot \epsilon\Big|_i \right) \mathbf{e}.$$

Let us define *weighted extraction inaccuracy* A_K of an extracted schematic with respect to performance K as the relative amount by which K varies if some or all parasitics are inexactly estimated. The weighted extraction inaccuracy is expressed as

$$A_K = \frac{|\Delta K| + \epsilon_p + \epsilon_r}{K_v} \tag{38}$$

where ϵ_p and ϵ_r are the errors due to inaccurate parasitic and performance models, respectively, and K_v is the nominal performance value. Moreover, (38) reduces to $A_K \approx \frac{|\Delta K|}{K_v}$ if $\epsilon_p + \epsilon_r + \ll |\Delta K|$. Suppose now that a constraint on the weighted accuracy $\bar{A}_K$ has been set

$$A_K \leq \bar{A}_K. \tag{39}$$

Then, (38) and (39) can be used as a criterion for selecting the appropriate template Problem (40) is *guaranteed* to have

$$\begin{array}{l} \qquad\qquad \textit{minimize}\text{: } D \\ \qquad\qquad \text{all templates} \\ \text{subject to} \\ \qquad\qquad A_K \leq \bar{A}_K. \end{array} \tag{40}$$

a solution, since a template with at least N_c contacts, all of them not overlapping with the actual circuit's contacts, exists by construction. Hence, arbitrarily small values of A_K can be achieved by simply extending D to include all the contacts i, $1 \leq i \leq N_c$. Problem (40) is solved by exhaustively calculating the minimum set D needed for each template for a given inaccuracy A_K. The procedure of calculating A_K and D has a time complexity of $O(N^2)$, while the overhead of computing $S_{Y_{ij}}$ is generally not accounted for since the evaluation is performed beforehand during circuit synthesis. Hence, a circuit with N_c contacts and a specification on A_K (39) can be extracted in $O([N_T + |D|]N^2)$ time, where N_T is the number of template circuits and $|D|$ the size of set D.

The final issue to be addressed is the efficient calculation of estimate $\epsilon|_i$, which can be computed exactly from update $\delta \mathbf{c}|_i$ using mapping $\mathbf{X}$ of (35). However, a more efficient computation of $\epsilon|_i$ can be obtained using the approximation of (32). Consider all the contacts $i \in D$, assume that the locations of i in the template and in the actual circuit are *close enough*. Then, a 2-D Taylor expansion for $\delta \mathbf{c}|_i$ can be constructed as

$$\delta \mathbf{c}|_i \approx \frac{\partial \mathbf{c}}{\partial v_x}\Big|_i \Delta v_x + \frac{\partial \mathbf{c}}{\partial v_y}\Big|_i \Delta v_y = \nabla_{\mathbf{v}} \mathbf{c}|_i \boldsymbol{\Delta} \mathbf{v}|_i \tag{41}$$

where vector $\boldsymbol{\Delta} \mathbf{v}|_i = [\Delta v_x \ \ \Delta v_y]^T$ represents the displacement needed to bring i from the template location to the location in the actual circuit. Term $\nabla_{\mathbf{v}} \mathbf{c}|_i = [\frac{\partial \mathbf{c}}{\partial v_x}|_i, \frac{\partial \mathbf{c}}{\partial v_y}|_i]^T$ is calculated using the Sherman–Morrison formula as in (32) and is valid for *small* displacements of contact i. Assume

[5] Assume all the other contacts are **not** displaced.

[6] The sensitivity of K with respect to matrix $\mathbf{Y}$ is an $N_c \times N_c$ matrix, whose terms in the ith row and jth column are given by the expression $\partial K / \partial Y_{ij}$.

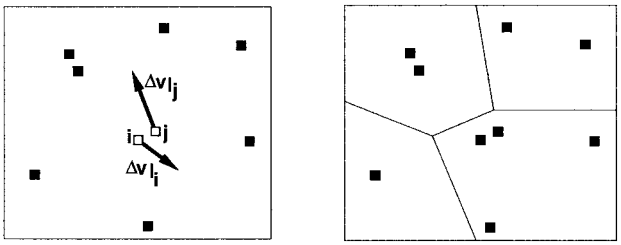


Fig. 15. (a) Displacement of contacts i and j in a single landscape. (b) Partitioning of substrate to minimize the number of different contaacts for which $\nabla_{\mathbf{v}}\mathbf{c}$ need be computed explicitly.

now that there exists a contact j in the vicinity of i which is displaced by $\mathbf{\Delta v}|_j$, where $|\mathbf{\Delta v}|_j|$ is also *small*. Assuming that the surrounding objects' relative distances from i and j are similar, one can estimate the cumulative effects of the displacement of the contacts as

$$\delta\mathbf{c}|_{i,j} \approx \nabla_{\mathbf{v}}\mathbf{c}|_i\mathbf{\Delta v}|_i + \nabla_{\mathbf{v}}\mathbf{c}|_i\mathbf{\Delta v}|_j \tag{42}$$

where vectors $\mathbf{\Delta v}|_i$ and $\mathbf{\Delta v}|_j$ relate to the displacements of i and j as shown in Fig. 15(a). Ideally, one would like to be able to compute $\mathbf{c}|_i$ using (42) for each contact $i = 1, \cdots, N_c$. However, far contacts "see" a completely different landscape, which causes term $\delta\mathbf{c}|_i$ to change by moving within the workspace. To improve the accuracy of (42), one could partition the workspace in order to minimize the number of contacts for which a new $\nabla_{\mathbf{v}}\mathbf{c}$ needs be computed. Fig. 15(b) shows such a partitioning. Notice that only one contact per partition, the *pole*, is used for the computation of $\nabla_{\mathbf{v}}\mathbf{c}$.

The problem of minimizing the number of partitions of Fig. 15(b) can be time-consuming, since it requires the estimation of each contact displacement to select the best candidates for the partitions and its poles. The complexity of this partitioning would nullify the efforts for an efficient substrate extraction. In addition, the needed parasitic estimate accuracy ϵ_p in (38) is not high. Hence, in our experiments a single contact was used to estimate $\delta\mathbf{c}|_i, \forall i$ with an error of 50% or less. Moreover, this error could be modeled as term ϵ_p in (38) and hence accounted for while determining D.

IV. Applications

A. Scaling and Technology Migration

Let us consider the scaling or technology migration for a given design (see Fig. 16). Redesign generally involves scaling in x- and y-directions, while technology migration involves a three-dimensional (3-D) scaling. Hereafter, we propose a generalized technique that can be used for both 2-D and 3-D scaling.

Consider first scaling in z-coordinate. Using (22)–(25),and the expressions in the Appendix, one can efficiently compute matrix $\partial\mathbf{Y}/\partial T_\ell$. Let us define a number of technology parameters for some design $\mathcal{D}$, $T_\ell^{(\mathcal{D})}$, $\ell = 1, \cdots, N_T$, which include layer thicknesses or profile discretizations d_k, $k = 1, \cdots, N_d$ in Fig. 3 and permittivity ϵ_k, $k = 1, \cdots, N_\epsilon$. Call $\mathbf{T}^{(\mathcal{D})}$ the $N_T \times 1$ vector whose elements are the $T_\ell^{(\mathcal{D})}$ terms.

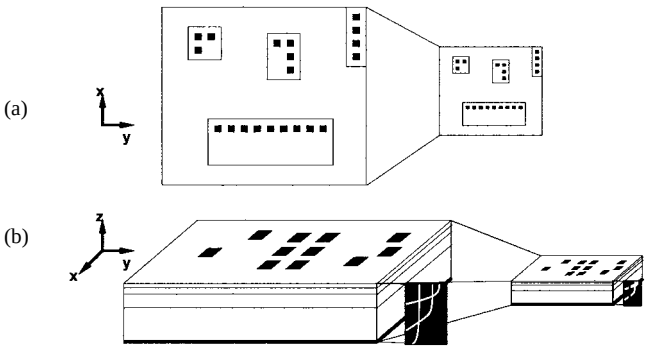


Fig. 16. (a) Two-dimensional scaling in redisign. (b) Three-dimensional scaling in technology migration.

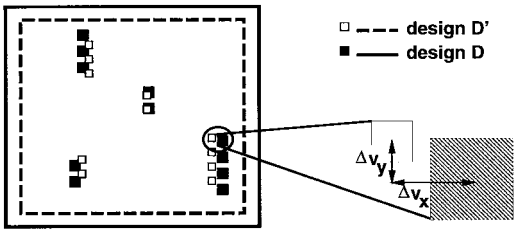


Fig. 17. Scaling in x- and y-directions. Relocation of contacts and area scaling.

Suppose now that conductance matrix $\mathbf{Y}^{(\mathcal{D})}$ has been calculated for a set of parameters $\mathbf{T}^{(\mathcal{D})}$. In addition, assume that an array of parameters $\mathbf{T}^{(\mathcal{D}')}$, associated with a new design $\mathcal{D}'$, is also available. Define the $N_T \times 1$ vector $\mathbf{\Delta T}^{(\mathcal{D}\mathcal{D}')} = \mathbf{T}^{(\mathcal{D}')} - \mathbf{T}^{(\mathcal{D})}$ as the variation of technology parameters across designs $\mathcal{D}$ and $\mathcal{D}'$. Conductance matrix $\mathbf{Y}^{(\mathcal{D}')}$, associated with the new design, can be computed using a first-order Taylor expansion as

$$\mathbf{Y}^{(\mathcal{D}')} \approx \mathbf{Y}^{(\mathcal{D})} + \sum_{\ell=1}^{N_T} \left[\frac{\partial\mathbf{Y}}{\partial\mathbf{T}_\ell}\right]_{\mathbf{T}_\ell^{(\mathcal{D})}} \Delta\mathbf{T}_\ell^{(\mathcal{D}\mathcal{D}')} \tag{43}$$

provided that designs $\mathcal{D}$ and $\mathcal{D}'$ are *close enough*, i.e., $\max_\ell\{|\Delta T_\ell^{(\mathcal{D}\mathcal{D}')}|/T_\ell^{(\mathcal{D})}\}$ is small. Consider next scaling in (x, y)-direction. Assume that a contact i in design $\mathcal{D}$ is located at a point $\mathbf{v_i}^{(\mathcal{D})} = [v_x, v_y]^T$, while a contact's position in design $\mathcal{D}'$ is $\mathbf{v_i}^{(\mathcal{D}')}$. Furthermore, assume the contact's area is not significantly changed across designs. Suppose that conductance matrix $\mathbf{Y}^{(\mathcal{D})}$ has been calculated for design $\mathcal{D}$ and that vectors $\mathbf{v_i}^{(\mathcal{D})}$ and $\mathbf{v_i}^{(\mathcal{D}')}$ are given $\forall i = 1, \cdots, N_c$. Let $\mathbf{\Delta v_i}^{(\mathcal{D}\mathcal{D}')}$ be the change in location for contact i as illustrated in Fig. 17. Using (32), one can approximate matrix $\mathbf{Y}$ as follows:

$$\mathbf{Y}^{(\mathcal{D}')} \approx \mathbf{Y}^{(\mathcal{D})} + \sum_{i=1}^{N_c} [\nabla_{\mathbf{v}}\mathbf{Y}]^{\mathbf{T}} \Delta v_i^{(\mathcal{D}\mathcal{D}')}. \tag{44}$$

Equations (43) and (44) can be combined so as to account for 3-D scaling realistically.

B. Technology Selection Based on Nondeterministic Data

In the above discussion, we have assumed that the values of technology variations ΔT_ℓ and geometric displacements Δv_i

are of a deterministic nature. Suppose on the contrary that we are given the statistical behavior of all or some technology parameters $\Delta T_\ell, \forall \ell = 1, \cdots, N_T$. Assume that the terms ΔT_ℓ are random variables with mean μ_ℓ and variance σ_ℓ^2, moreover suppose that all ΔT_ℓ are *statistically independent*.

Then, the mean $E(Y_{ij})$ and variance $\sigma^2(Y_{ij}) = E(Y_{ij}^2) - E^2(Y_{ij})$ of each entry of conductance matrix $\mathbf{Y}$ can be computed as

$$\begin{aligned} E(Y_{ij}) &\approx Y_{ij}^{(\mathcal{D})} + \sum_{\ell=1}^{N_T} \left[\frac{\partial Y_{ij}}{\partial T_\ell}\right]_{T_\ell^{(\mathcal{D})}} \mu_\ell \\ \sigma^2(Y_{ij}) &\approx \sum_{\ell=1}^{N_T} \left|\left[\frac{\partial Y_{ij}}{\partial T_\ell}\right]_{T_\ell^{(\mathcal{D})}}\right|^2 \sigma_\ell^2 \end{aligned} \tag{45}$$

where $[\partial Y_{ij}/\partial T_\ell]_{T_\ell^{(\mathcal{D})}}$ is the sensitivity of entry Y_{ij} with respect to T_ℓ related to the original design $\mathcal{D}$.

Our sensitivity-based method for the computation of mean and variance of $\mathbf{Y}$ can also be used for the selection of a technology which is most suitable for a certain circuit and its associated performance specifications. Suppose, for instance, that N_p constraints on all critical substrate coupling $R_{ij}^{(\text{bound})}$ have been computed using the techniques presented in [18]. Furthermore, assume that a number of technologies $\mathcal{T}$ is available and that all relevant parameters $T_\ell^{(\mathcal{T})}$ are identified. Suppose, however, that for some or all technologies, a number of parameters are not known precisely and only rough estimates with uncertainty exist. Assume that estimate and uncertainty can be modeled into each parameter in terms of its mean and variance. Then, by computing the mean and the variance of $\mathbf{R}$ for a set of parameters $T_\ell^{(\mathcal{T})}$, one can derive the probability with which constraints $R_{ij}^{(\text{bound})}$ will be met

$$P_{\mathcal{T}}\left[\mu(R_{ij}), \sigma^2(R_{ij}), R_{ij}^{(\text{bound})}\right] = \text{erf}\left(\frac{R_{ij}^{(\text{bound})} - \mu(R_{ij})}{\sigma(R_{ij})}\right) \tag{46}$$

provided that $T_\ell^{(\mathcal{T})}$ is Gaussian. Notice that $\text{erf}(x)$ is defined here as the integral of a normal distribution $N(0,1)$ from minus infinity to x. The problem of selecting a technology most likely to satisfy all constraints is equivalent to maximizing $P_{\mathcal{T}}$ over all critical constraints. Due to the efficiency of our techniques for the calculation of means and variances, the problem can be solved by exhaustively computing $P_{\mathcal{T}}[\mu(R_{ij}), \sigma^2(R_{ij}), R_{ij}^{(\text{bound})}]$ for each technology $\mathcal{T}$.

C. Placement Problem

A substrate-aware placement methodology has been implemented in a simulated annealing (SA)-based framework with analog constraints, called PUPPY-A [20], [21]. The annealing, fully characterized by search space, cost function, move-set, and cooling schedule, is described in detail in [17, ch. 4]. Improvements on the performance degradation due to substrate-induced switching noise can be achieved by placing noise injecting and noise sensitive modules at a certain distance or by creating special structures, such as low-resistivity guard-rings, around noise injectors [15].

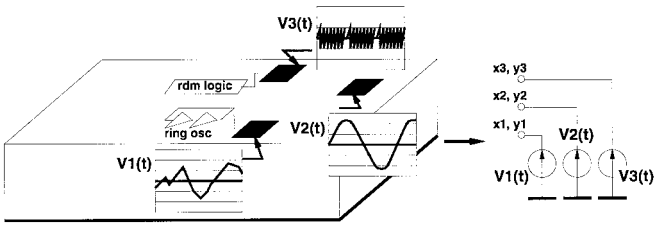


Fig. 18. Modeling noise injectors in the placer.

The first provision is implemented in the placer using the conventional SA move-set. The second issue is generally solved by extending the search space, allowing the annealing to choose from a number of alternative implementations for a module, including one with a guard-ring implemented around it. In this paper we restrict our attention to the first option, where our Green's Function-based substrate analysis method is used for the evaluation of the substrate at each annealing step.

In order for a placer to be effective in preventing violations to performance specifications, the following features must be implemented in the tool. First, a model for each noise injecting module must exist. The model should characterize the *waveform* and the *spatial location* where the noise is injected as precisely as possible (see Fig. 18). Second, a compact model of substrate transport should be available and efficient substrate current evaluation should be possible, independent of the circuit configuration. Third, a model for substrate noise absorption and its effect on performance should be defined.

For the purpose of physical assembly or schematic design, switching noise is often modeled as a simple signal, generally synchronized with the clock, if one is present. A number of examples of this modeling style can be found in the literature [6], [10], [22]. Alternatively, one can extract the actual noise waveform associated with a given logic circuit using event-driven simulation combined with a lookup table for the precise representation of every injection current. The method is explained in detail in [1]. In this paper we will use the results of this work applied to our examples.

For each noise injecting module j, a model is created which accurately reproduces substrate-injected noise, taking into account both impact ionization and capacitive coupling through devices and interconnect lines. The model $V_S(\mathbf{\Pi}_j)$ is based on a bank of independent current noise generators with a unified set of parameters, represented by vector $\mathbf{\Pi}_j$. The problem of evaluating the effects of substrate on performance is approached in the following way.

1) Compute constraints for node of noise-sensitive modules.
2) Generate resistive network associated with substrate.
3) Quantify violations to constraints.

The sensitivity of a given performance K_i is computed with respect to the parameters $\mathbf{\Pi}_j$ related to each noise source j acting on every node in the analog modules being placed. In step 1), a set of bounds $\mathbf{\Pi}_j^{(\text{bound})}$ is generated for a subset of *critical nodes* n_c using constrained optimization techniques [23] and the specification on the maximum positive and negative performance degradation $\overline{\Delta K_i}^{\pm}$. Subset n_c is

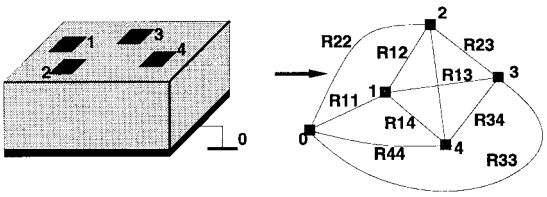
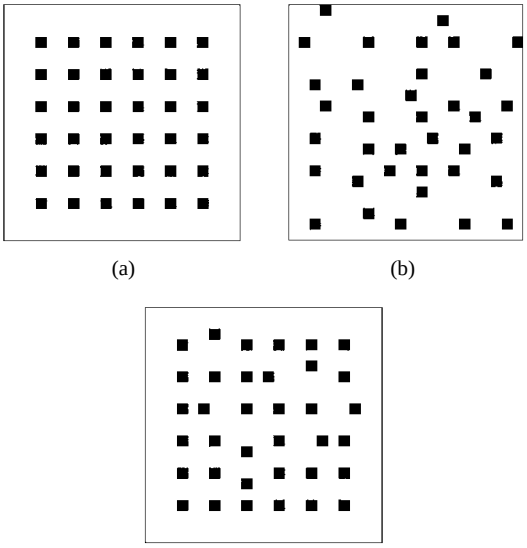


Fig. 19. Mapping substrate onto fully connected graph $G_S(V, E)$.



(a) (b) (c)

Fig. 20. (a) Initial contact grid, (b) reshuffling of contacts at high temperatures, and (c) resulting grid at lower temperatures.

generated from the cumulative impact of all parasitic noise sources acting on each node as in [23].

In step 2) a given placement configuration is mapped onto a fully connected graph $G_S(V, E)$, whose vertices V are the substrate contacts and edges E are weighted by the conductance Y_{ij} or resistance R_{ij} between the corresponding vertices i and j. Fig. 19 shows the mapping procedure. The techniques for the evaluation of the edges have been described in detail in Section II. The calculation of all violations in step 3) to the given constraints is carried out by solving the circuit underlying $G_S(V, E)$ and evaluating the appropriate parameters at each critical node.

At each stage of the annealing, only steps 2) and 3) need be repeated, since step 1) is carried out only once for each chip. The efficiency of a Green's Function-based substrate simulator, though high, is still insufficient for such computationally intensive algorithms as SA, hence, appropriate heuristics must be developed. In SA, at high annealing temperatures, considerable reshuffling is allowed on the components of the layout. Hence, the locations of switching noise generators and receptors can be significantly modified. At lower temperatures, on the contrary, modules move by lesser amounts in average. Hence, the edges of $G_S(V, E)$ change with lower frequency and by a lesser amounts.

As an illustration, consider a regular 36-contact grid shown in Fig. 20(a). The plot of Fig. 21 shows the average variation of the resistive components of the substrate network

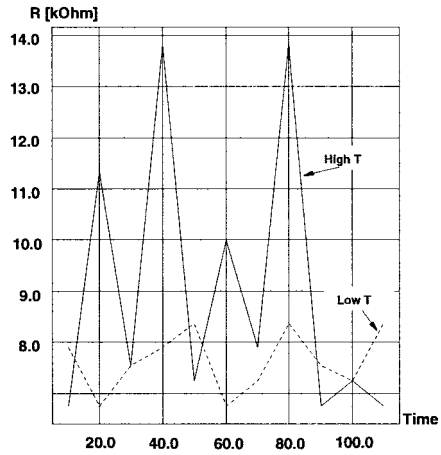


Fig. 21. Resistive network reacting to high-temperature and low-temperature contact reshuffling.

when high-temperature [Fig. 20(b)] and low-temperature [Fig. 20(c)] contact perturbations occur during the unfolding of SA. On the other hand, only when changes in the edges of $G_S(V, E)$ reflect a significant change in any performance measure K_i, should the entire substrate network be evaluated along with the estimate of performance degradation ΔK_i. When a new temperature T_k is reached, the full graph $G_S(V, E)$ is solved, i.e., all the edges in V are evaluated exactly, using the Sherman–Morrison update to obtain the new matrix $\mathbf{P^{-1}}$. After a new move m_k and the associated translation $\mathbf{\Delta v} = [\Delta v_x \ \Delta v_y]^T$ is selected by the annealing algorithm, the sensitivity of the edges of $G_S(V, E)$ can be efficiently computed using the techniques outlined in Section III-C. Suppose subset n_c of all critical receptors has been derived for the circuit; moreover, let n_s be the subset of all noise injecting nodes. Let $[\mathbf{Y_c}]_{m_k}$ be the conductance matrix of all the nodes in n_c and in n_s and let $[\mathbf{\Delta Y_c}]_{m_k}$ be its update. By (32), term $[\mathbf{\Delta Y_c}]_{m_k}$ is estimated as

$$[\mathbf{\Delta Y_c}]_{m_k} \approx \left[\nabla_{\mathbf{v}} \mathbf{Y^T}\right]_0 \mathbf{\Delta v} \tag{47}$$

where term $[\nabla_{\mathbf{v}} \mathbf{Y^T}]_0$ is defined as in (32) for matrix $\mathbf{Y_c}$. After updating $\mathbf{Y_c}$, the resistive network is solved and parameter $\mathbf{\Pi}_j$ can be evaluated for all critical nodes j. By comparing $\mathbf{\Pi}_j$ with the bound $\mathbf{\Pi}_j^{(\text{bound})}$, one can obtain the corresponding violation. If a violation to specifications has occurred, then a precise extraction step must be performed, and the precise value for the violation is used to drive the cost of the annealing in a manner similar to [20]. Otherwise, the contribution of substrate noise to node j in degrading performance K_i is considered negligible, and the cost function will not take it into account. The cost relative to the remaining analog-specific constraints, as well as area and wiring length, will, however, be computed. The placement algorithm is proved to converge to a global minimum under the same conditions of [24] and [25] when it is modified to account for noise substrate transport evaluation [17].

V. Case Study

The circuit used in our experiments is a 140-MHz monitor display controller (RAMDAC) including three D/A converters,

```
foreach temperature T_k
    evaluate_substrate_network_exactly;    // use Sherman-Morrison to
      update
                                           // configuration at equilibrium
    repeat
        m_k = select_move;
        estimate_network_change(m_k);      // use gradient based method
        foreach node j ∈ n_c
            Π_j = estimate_cumulative_noise;
            if Π_j ≥ Π_j^(bound)
                evaluate_substrate_network_exactly;
            go_to_next_node;
        evaluate_cost_function;
        accept_or_reject_move;
    until equilibrium_reached
```

Fig. 22. Heuristic for the combined use of Sherman–Morrisonn and gradient-based methods.

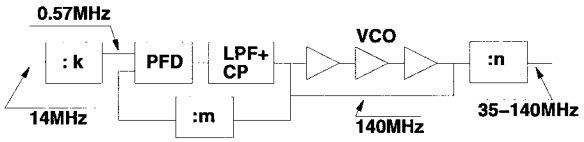


Fig. 23. PLL schematic.

TABLE I
PLL SPECIFICATIONS

Measure	Conditions			Specs
	PLL input freq.	n	VCO freq.	
Stability	0.56 MHz	100	56 MHz	Yes
		250	140 MHz	Yes
Jitter $\Delta T/T$	0.56 MHz	250	140 MHz	≤ 0.007
Ph. Margin	-	-	-	$\geq 45°$

a phase lock loop (PLL) frequency synthesizer, and digital control logic. The circuit was integrated in a Mosis HP 1-μm CMOS technology. The substrate parameters used by SUBRES are similar to Fig. 6(b) with a discretization matching the exponential doping curve. The converters were generated using dedicated silicon compilers [26]. The PLL needed particular care due to its extremely high sensitivity to thermal noise and spurious signals originated within the chip.

The PLL architecture, shown in Fig. 23, was derived from [27]. Device sizing was performed using a modified version of the supporting hyperplane algorithm and SPICE for circuit evaluation [28]. The circuit consists of a digital section, i.e., three divide-by-n modules and a phase-frequency detector (PFD), and a number of analog components, i.e., an analog low-pass filter (LPF) and a charge pump (CP). The interface between analog and digital sections is represented by the voltage-controlled oscillator (VCO), which generates a digital output at a frequency proportional to the input voltage. Typical frequencies of operation are shown in the various branches of the circuit in Fig. 23.

The specifications for the PLL are summarized in Table I. The jitter $\frac{\Delta T}{T}$ is defined as the ratio between the variation from nominal of oscillation period ΔT and period T. Due to the time-variance of $\frac{\Delta T}{T}$, it is generally measured in terms of its peak-to-peak or RMS value.

A. Physical Design

The jitter performance of the PLL is entirely dependent on the jitter produced by the VCO. Using this fact, a sensitivity-based model of the PLL could be constructed relating the PLL jitter performance to the level of the noise voltage peak-to-peak present at some 85 critical locations in the VCO. All critical substrate noise receptors were identified in the delay elements and in the two bias circuits using SPICE simulations accounting for both impact ionization and capacitive coupling. Interconnect parasitics and IR drops were also identified (see Fig. 24). Sensitivities with respect to all parasitics (RC for interconnect and V_0 for substrate receptors) were computed. Then, constraint generator PARCAR [23], [19] was used to derive a minimal set of constraints on the maximum admissible noise voltage in each one of the receptors and on the maximum R/C values for the interconnect parasitics in the VCO. The CPU time needed for the sensitivity analysis and constraint calculation was in total 2545 s, the results are shown in Table II. Interconnect parasitic constraints were exploited by a constraint-based module generator VCOGEN to synthesize the VCO. TIMBERWOLFSC-4.1 was used for the internal divider. The module generation step required a total of 163 s on a DEC AlphaServer 2100 5/250.

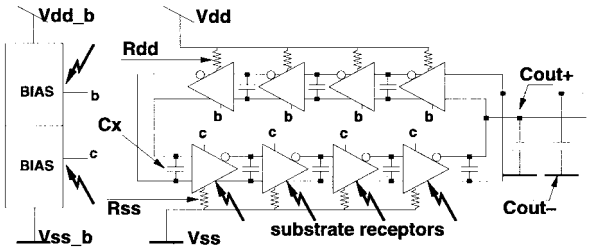


Fig. 24. Interconnect parasitics and substrate noise receptors.

TABLE II
CONSTRAINTS OBTAINED FROM PARCAR

Critical Parasitic	Constraint	Extracted value
C_{out+}	$12.34fF$	$11.60fF$
C_{out-}	$12.34fF$	$11.60fF$
C_x	$15.84fF$	$1.12fF$
$V_0\|_{VCO}$	$110mV$	$109mV$
$V_0\|_{LPF}$	-	-

Note: only about 10% of all the parasitics need be constrained. The remaining 90% cannot be higher than the upper-bound ensured by the technology.

The next step was the placement of the component blocks of the PLL and of the other circuits in the RAMDAC. The placement was carried out using PUPPY-A. In the circuit there exist three major switching noise injectors, corresponding to the dividers. In order to accurately verify if the constraints on the maximum admissible noise voltage were violated, an accurate model was constructed of the injectors using the tool SUBWAVE [1]. SUBWAVE generates simplified substrate noise models, accounting for currents injected via capacitive coupling and impact ionization from active device areas and supply lines. For the capacitive coupling models used in SUBWAVE, we refer to [15] and [1].

Assuming that the substrate shows a purely resistive behavior, the calculation of the peak-to-peak voltage at each node of the surface can be carried out by performing a simple dc analysis on the positive and negative peak values of the current of the injector. The placement was performed using the heuristics summarized in Fig. 22. The constraints

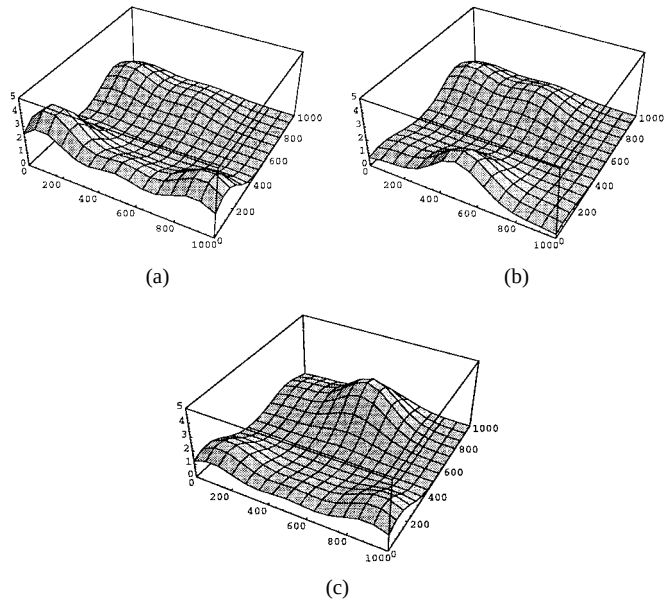


Fig. 25. Estimated switching noise signal amplitude resulting from cumulative divider injection during SA. The signal was normalized with respect to the lowest constraint over the entire 1000 × 1000 μm chip. Violations at (a) high, (b) medium, and (c) low temperature.

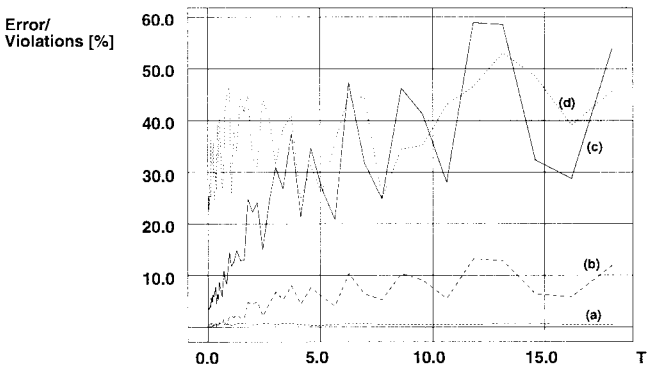


Fig. 26. Error in substrate injection estimation using (a) combined heuristic and (b) gradient-based method only. All substrate violations using (c) combined heuristic and (d) no substrate control.

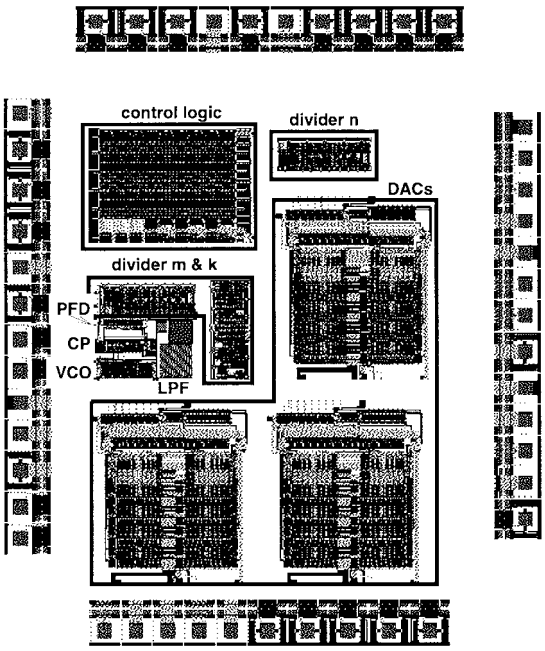


Fig. 27. Placed PLL within the RAMDAC.

TABLE III
PLACEMENT STATISTICS OBTAINED ON A DEC ALPHASERVER 2100 5/250

Mode	CPU (sec)	Area (λ^2)	Est. Jitter
Manual	-	5637 x 6481	-
Parasit.	406.74	6765 x 6528	0.1
Subst.+Parasit.	885.20	7322 x 7716	0.005

TABLE IV
NOISE INJECTOR AND RECEPTOR STATISTICS IN THE COMPONENTS OF THE PLL

Component	Number of receptors	Number of injectors
divider	-	152
PFD	-	23
VCO	85	-
LPF	5	-
CP	46	-
Total	136	175

on the maximum admissible noise voltage at each node of the VCO were used in the cost function of the annealing in a manner identical to [20]. Fig. 25 shows the estimated values of switching noise voltage at each location in the chip at different temperatures during the annealing. Fig. 25(c) shows the substrate noise distribution at the end of the SA run. As expected, the algorithm successfully minimized the noise present in the substrate underlaying the PLL (compare layout in Fig. 27). The plot of Fig. 26 shows the impact of estimation algorithms on the relative error in substrate noise measured at the receptors during the annealing. All relative errors are obtained by comparison with an exact method, i.e., the Sherman–Morrison update. Curves (a) through (c) and (d) show how the constraint violation is driven toward zero depending on whether or not the proposed substrate injection control is used. Fig. 27 shows the final placement performed using PUPPY-A. As expected, divider n was placed at a large distance from the sensitive components of the PLL, namely the CP, VCO, and LPF. On the contrary, the sensitivity of these components with respect to the switching noise produced by divider k is small, hence it could be placed accordingly. For divider m, the placer had to perform a tradeoff between the strength of the switching noise received by it and the parasitics introduced when large interconnect capacitances are introduced. Using the same performance model employed in the constraint derivation, along with noise estimation techniques outlined in Section II, the jitter performance predicted in the PLL is summarized in Table III.

B. Trend Analysis and Technology Scaling

All the potential sources of switching noise in the PLL are localized in the dividers, while the receptors are in the VCO, CP, and LPF. Injection occurs by impact ionization through the active areas of NMOS devices (in an N-well processes) and by capacitive coupling through junctions and interconnect. Receptors are in the active areas of sensitive devices and supply lines. Table IV lists the main sources and receptors of noise in the various components of the design.

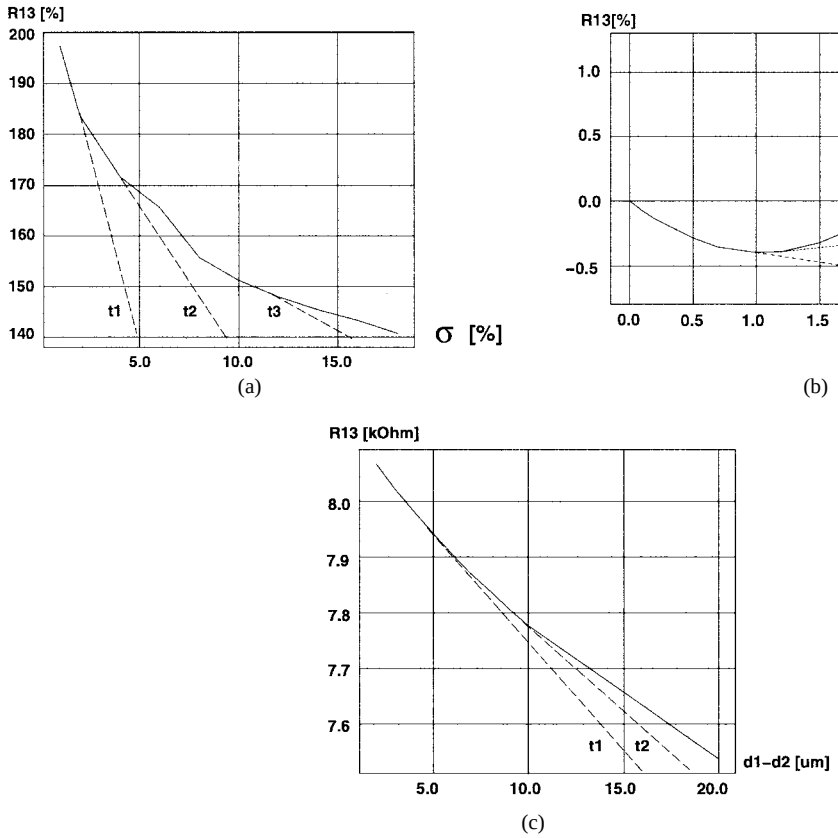


Fig. 28. R_{13} sensitivity with respect to (a) epitaxial doping levels, (b) contact depth, and (c) epitaxial depths.

Suppose one is interested in finding the change of jitter performance if a new lightly doped substrate is to be used instead of the low-resistivity substrate for which the circuit was designed. In this case performance K is the expression $\frac{\Delta T}{T}$. Since sensitivity $\partial\frac{\Delta T}{T}/\partial Y_{ij}$ is known, expression $\partial Y_{ij}/\partial T_\ell$ $\forall i,j$ remains to be calculated. Note that in this case T_ℓ is a particular doping level associated with the layer of interest. The plot in Fig. 28(a) shows the values of the sensitivities of entry $R_{13} = 1/Y_{13}$ at various nominal doping levels $(t_1, \cdots, t_3)$. Substrate impedance R_{13} was chosen as an illustration due to the high sensitivity of jitter with respect to it. The impedance is in fact responsible for approximately 20% of the noise generated in divider n and picked up by the VCO.

Consider now the dependence of impedance R_{13} as a function of another technology-specific parameter, namely the contact layer depth c. See plot in Fig. 28(b). Lines $(t_1, \cdots, t_4)$ in Fig. 28(b) represent the sensitivities of R_{13} at several values of c as computed using the formulae in the Appendix. Let us now consider the effects of changes in the doping profiles in Fig. 2. Assume that the number of layers stays constant but the epitaxy expands toward the ground-plane while the underlaying layer shrinks. The plot in Fig. 28(c) shows the sensitivities (t_1, t_2) of R_{13} as a dependence of the thickness of the epitaxial layer. Table V reports all CPU times for the sensitivities computed in the experiments and the estimated trend of jitter performance degradation calculated using (20).

TABLE V
CPU TIMES ON A DEC ALPHASERVER 2100 5/250 FOR THE TREND ANALYSIS FOR THE PROPOSED EXPERIMENTS ON THE PLL WITH 311 NOISE SOURCES/RECEPTORS. THE CPU TIMES INCLUDE DCT, PARAMETER, AND SENSITIVITY COMPUTATION. FOR THE CALCULATION OF 311 CONTACTS, THE INVERSION MATRIX $\mathbf{P}$ WAS PERFORMED IN 1525.0 S. THE SIZE OF $\mathbf{P}$ WAS 1244 × 1244

Experiment	CPU times (sec)	Jitter trend
Epitaxial doping	3038.88	1.34
Contact depth	2858.83	0.95
Profile change	4005.46	0.55

TABLE VI
SUBSTRATE EXTRACTION IN PRESENCE OF VARYING TECHNOLOGY PARAMETERS USING METHOD I (FULL EXTRACTION) AND METHOD II (SENSITIVITY-BASED EXTRACTION)

	method I	method II	error
R_{12}	8154.03 Ω	8098.60 Ω	0.67 %
R_{13}	1866.15 Ω	1848.10 Ω	0.96 %
R_{23}	3788.62 Ω	3743.80 Ω	1.18 %
R_{10}	893.85 Ω	893.08 Ω	0.08 %
R_{20}	460.40 Ω	458.59 Ω	0.39 %
R_{30}	690.77 Ω	688.59 Ω	0.31 %

configuration of Fig. 13

# contacts	method I	method II	max. error
100	73.8 sec	9.8 sec	3 %

uniform 10 × 10 contact grid

# contacts	method I	method II	max. error
2,500	not completed	57 min	5 %

industrial mixed-signal circuit

C. Accelerated Extraction and Technology Selection

Table VI lists the values of matrix $\mathbf{Y}$ using full and sensitivity-based extraction for two configurations. All CPU times are referred to a DEC AlphaServer 2100 5/250 and relate to all computations except for the Green's Function, which is performed once for a given substrate structure. The

TABLE VII
MEAN AND VARIANCE OF THE ENTRIES OF MATRIX $\mathbf{R}$ AS A FUNCTION OF DEPTH VARIANCE. ALL VALUES ARE REFERRED TO A MEAN DEPTH OF 1 μm. THE EXECUTION TIMES ARE REPORTED FOR A UNIFORM 10 × 10 CONTACT GRID

	$\mu(R)$	$\sigma^2(R)$	
	$\mu(c) = 1\mu m$	$\sigma^2(c) = 0.25\mu m^2$	$\sigma^2(c) = 0.1\mu m^2$
R_{12}	8066.33 Ω	0.0270 Ω^2	0.0109 Ω^2
R_{13}	1836.69 Ω	0.0040 Ω^2	0.0017 Ω^2
R_{23}	3716.29 Ω	0.0324 Ω^2	0.1296 Ω^2
R_{10}	892.11 Ω	$2.5\times10^{-7}\Omega^2$	$10^{-7}\Omega^2$
R_{20}	456.35 Ω	$10^{-4}\Omega^2$	$4\times10^{-5}\Omega^2$
R_{30}	685.69 Ω	$2.25\times10^{-4}\Omega^2$	$9\times10^{-4}\Omega^2$

configuration of Fig. 13

# contacts	$\mu(R)$	$\sigma^2(R)$
100	73.8 sec	16.0 sec

uniform 10 × 10 contact grid

TABLE VIII
SELECTION OF MOST SUITABLE TECHNOLOGY BASED ON THE PROBABILITY OF SATISFYING ALL CONSTRAINTS ON SUBSTRATE COUPLING RESISTANCES

	$R_{ij}^{(bound)}$	$P_{\mathcal{T}}(\mu(c)[\mu m]/\sigma^2(c)[\mu m^2])$	
		$\mathcal{T}_1$: 0 / 0.1	$\mathcal{T}_2$: 1 / 0.25
R_{12}	8066.5 Ω	0.849569	0.948270
R_{13}	1836.8 Ω	0.959005	0.996184
R_{23}	3716.4 Ω	0.729437	0.620028
R_{10}	893.0 Ω	1.0	1.0
R_{20}	457.0 Ω	1.0	1.0
R_{30}	685.8 Ω	1.0	0.999877
Total	-	5.538011	5.564359

error is reported for all configurations. Note that a large circuit with 2500 contacts could not be handled unless an extraction acceleration scheme was used.

Consider now Fig. 13 and a uniform 10 × 10 contact grid. Table VII lists the mean and variance of the entries of matrix $\mathbf{R}$ as a function of depth variance $\sigma^2(c)$, assuming $\mu(c) = 1$ μm. The execution times for the extraction of the mean and variance of $\mathbf{R}$ are also reported. For the example of Fig. 13, suppose that all six substrate resistances R_{ij} and R_{i0} were critical and that constraints on each resistance were set as listed in Table VIII. Clearly, technology $\mathcal{T}_2$ is more likely to meet the above specifications and hence it should be selected as best candidate.

VI. CONCLUSION

Novel techniques for the acceleration of substrate noise analysis in an optimization loop are described. A boundary element method is used to characterize the substrate. Mutual and ground resistances are efficiently computed using a Green's Function for multilayered substrate via the discrete cosine transform. Efficient sensitivity analysis of substrate performance with respect to geometric features and technology parameters is used to accurately assess effects and trends due to design modifications. We have shown the usefulness of the techniques in a number of optimization problems, specifically targeted toward technology migration, selection, and scaling. In particular, the approach has been demonstrated through a medium-sized mixed-signal IC on which a complete analysis of the impact of substrate was performed before fabrication.

APPENDIX
SENSITIVITY DERIVATIONS

The term $G_0|_{z=z'=0}$ is computed as

$$G_0|_{z=z'=0} = \frac{1}{ab\epsilon_N}\frac{\Gamma_N}{\beta_N}. \tag{48}$$

The terms Γ_N and β_N are computed recursively as follows:

$$\begin{bmatrix}\beta_k\\ \Gamma_k\end{bmatrix} = \begin{bmatrix}\frac{\epsilon_{k-1}}{\epsilon_k} & 0\\ \left(\frac{\epsilon_{k-1}}{\epsilon_k}-1\right)d_k & 1\end{bmatrix}\begin{bmatrix}\beta_{k-1}\\ \Gamma_{k-1}\end{bmatrix} \tag{49}$$

where the recursion begins with the values $\beta_0 = 1$ and $\Gamma_0 = d$. The term f_{mn} is computed as follows:

$$\begin{aligned} f_{mn} &= \frac{1}{ab\gamma_{mn}\epsilon_N}\frac{\beta_N\tanh(\gamma_{mn}d)+\Gamma_N}{\beta_N+\Gamma_N\tanh(\gamma_{mn}d)}\\ \gamma_{mn} &= \sqrt{\left(\frac{m\pi}{a}\right)^2+\left(\frac{n\pi}{b}\right)^2}. \end{aligned} \tag{50}$$

The terms β_N and Γ_N for $m \neq 0$ or $n \neq 0$ are computed recursively as follows:

$$\begin{aligned}\begin{bmatrix}\beta_k\\ \Gamma_k\end{bmatrix} = &\begin{bmatrix}\frac{\epsilon_{k-1}}{\epsilon_k}\cosh^2(\theta_k)-\sinh^2(\theta_k) & \left(\frac{\epsilon_{k-1}}{\epsilon_k}-1\right)\cosh(\theta_k)\sinh(\theta_k)\\ \left(1-\frac{\epsilon_{k-1}}{\epsilon_k}\right)\cosh(\theta_k)\sinh(\theta_k) & \cosh^2(\theta_k)-\frac{\epsilon_{k-1}}{\epsilon_k}\sinh^2(\theta_k)\end{bmatrix}\\ &\times\begin{bmatrix}\beta_{k-1}\\ \Gamma_{k-1}\end{bmatrix}\end{aligned} \tag{51}$$

where $1 \le k \le N$, $\theta_k = \gamma_{mn}(d-d_k)$, $\beta_0 = 1$, and $\Gamma_0 = 0$.

Assume $T_\ell = \epsilon_\ell$, then all Γ_k and β_k will not depend on ϵ_ℓ when $0 \le k < \ell$, hence

$$\begin{bmatrix}\frac{\partial\beta_k}{\partial T_\ell}\\ \frac{\partial\Gamma_k}{\partial T_\ell}\end{bmatrix} = \mathbf{0}, \quad \forall 0 \le k < \ell.$$

Consider first the case in which $k = \ell$. Equation (51) becomes

$$\begin{aligned}\begin{bmatrix}\frac{\partial\beta_\ell}{\partial T_\ell}\\ \frac{\partial\Gamma_\ell}{\partial T_\ell}\end{bmatrix} = &\frac{\epsilon_{\ell-1}}{\epsilon_\ell^2}\begin{bmatrix}-\cosh^2(\theta_\ell) & -\cosh(\theta_\ell)\sinh(\theta_\ell)\\ \cosh(\theta_\ell)\sinh(\theta_\ell) & \sinh^2(\theta_\ell)\end{bmatrix}\\ &\times\begin{bmatrix}\beta_{\ell-1}\\ \Gamma_{\ell-1}\end{bmatrix}\end{aligned} \tag{52}$$

where $\Gamma_{\ell-1}$ and $\beta_{\ell-1}$ are already known, while $\theta_\ell = \gamma_{mn}(d-d_\ell)$ and $\gamma_{mn} = \sqrt{(\frac{m\pi}{a})^2+(\frac{n\pi}{b})^2}$.

Second, consider the case in which $k = \ell+1$. Equation (51) becomes

$$\begin{aligned}&\begin{bmatrix}\frac{\partial\beta_{\ell+1}}{\partial T_\ell}\\ \frac{\partial\Gamma_{\ell+1}}{\partial T_\ell}\end{bmatrix}\\ &= \frac{1}{\epsilon_{\ell+1}}\begin{bmatrix}\cosh^2(\theta_{\ell+1}) & \cosh(\theta_{\ell+1})\sinh(\theta_{\ell+1})\\ -\cosh(\theta_{\ell+1})\sinh(\theta_{\ell+1}) & -\sinh^2(\theta_{\ell+1})\end{bmatrix}\\ &\quad\times\begin{bmatrix}\beta_\ell\\ \Gamma_\ell\end{bmatrix}\\ &\quad+\begin{bmatrix}\frac{\epsilon_\ell}{\epsilon_{\ell+1}}\cosh^2(\theta_{\ell+1})-\sinh^2(\theta_{\ell+1}) & \left(\frac{\epsilon_\ell}{\epsilon_{\ell+1}}-1\right)\cosh(\theta_{\ell+1})\sinh(\theta_{\ell+1})\\ \left(1-\frac{\epsilon_\ell}{\epsilon_{\ell+1}}\right)\cosh(\theta_{\ell+1})\sinh(\theta_{\ell+1}) & \cosh^2(\theta_{\ell+1})-\frac{\epsilon_\ell}{\epsilon_{\ell+1}}\sinh^2(\theta_{\ell+1})\end{bmatrix}\\ &\quad\times\begin{bmatrix}\frac{\partial\beta_\ell}{\partial T_\ell}\\ \frac{\partial\Gamma_\ell}{\partial T_\ell}\end{bmatrix}.\end{aligned} \tag{53}$$

For $\ell+1<k\leq N$, $\partial\beta_k/\partial T_\ell$ and $\partial\Gamma_k/\partial T_\ell$ are computed as

$$\begin{bmatrix}\frac{\partial\beta_k}{\partial T_\ell}\\ \frac{\partial\Gamma_k}{\partial T_\ell}\end{bmatrix}=\begin{bmatrix}\frac{\epsilon_{k-1}}{\epsilon_k}\cosh^2(\theta_k)-\sinh^2(\theta_k) & \left(\frac{\epsilon_{k-1}}{\epsilon_k}-1\right)\cosh(\theta_k)\sinh(\theta_k)\\ \left(1-\frac{\epsilon_{k-1}}{\epsilon_k}\right)\cosh(\theta_k)\sinh(\theta_k) & \cosh^2(\theta_k)-\frac{\epsilon_{k-1}}{\epsilon_k}\sinh^2(\theta_k)\end{bmatrix}\times\begin{bmatrix}\frac{\partial\beta_{k-1}}{\partial T_\ell}\\ \frac{\partial\Gamma_{k-1}}{\partial T_\ell}\end{bmatrix}\tag{54}$$

recursively, where $\partial\beta_{k-1}/\partial T_\ell$ and $\partial\Gamma_{k-1}/\partial T_\ell$ are obtained directly from (53). The recursion (54) ends when $\partial\beta_N/\partial T_\ell$ and $\partial\Gamma_N/\partial T_\ell$ are found.

Next, assume $T_\ell=d_\ell$, the layer thickness. Using a similar reasoning as before, consider first the case in which $k=\ell$. Equation (51) becomes

$$\begin{bmatrix}\frac{\partial\beta_\ell}{\partial T_\ell}\\ \frac{\partial\Gamma_\ell}{\partial T_\ell}\end{bmatrix}=-\gamma_{mn}\left(\frac{\epsilon_{\ell-1}}{\epsilon_\ell}-1\right)\begin{bmatrix}2\sinh(\theta_\ell)\cosh(\theta_\ell) & \cosh(2\theta_\ell)\\ -\cosh(2\theta_\ell) & -2\sinh(\theta_\ell)\cosh(\theta_\ell)\end{bmatrix}\times\begin{bmatrix}\beta_{\ell-1}\\ \Gamma_{\ell-1}\end{bmatrix}\tag{55}$$

where $\Gamma_{\ell-1}$ and $\beta_{\ell-1}$ are already known, while $\theta_\ell=\gamma_{mn}(d-d_\ell)$ and $\gamma_{mn}=\sqrt{(\frac{m\pi}{a})^2+(\frac{n\pi}{b})^2}$.

Second, consider again the case in which $\ell+1\leq k\leq N$, $\partial\beta_k/\partial T_\ell$ and $\partial\Gamma_k/\partial T_\ell$ are computed as

$$\begin{bmatrix}\frac{\partial\beta_k}{\partial T_\ell}\\ \frac{\partial\Gamma_k}{\partial T_\ell}\end{bmatrix}=\begin{bmatrix}\frac{\epsilon_{k-1}}{\epsilon_k}\cosh^2(\theta_k)-\sinh^2(\theta_k) & \left(\frac{\epsilon_{k-1}}{\epsilon_k}-1\right)\cosh(\theta_k)\sinh(\theta_k)\\ \left(1-\frac{\epsilon_{k-1}}{\epsilon_k}\right)\cosh(\theta_k)\sinh(\theta_k) & \cosh^2(\theta_k)-\frac{\epsilon_{k-1}}{\epsilon_k}\sinh^2(\theta_k)\end{bmatrix}\times\begin{bmatrix}\frac{\partial\beta_{k-1}}{\partial T_\ell}\\ \frac{\partial\Gamma_{k-1}}{\partial T_\ell}\end{bmatrix}\tag{56}$$

recursively, where $\partial\beta_{k-1}/\partial T_\ell$ and $\partial\Gamma_{k-1}/\partial T_\ell$ are obtained directly from (55). The recursion (56) ends when $\partial\beta_N/\partial T_\ell$ and $\partial\Gamma_N/\partial T_\ell$ are obtained.

Consider now the sensitivity of the term k_{mn} with respect to parameter T_ℓ. k_{mn} is defined in (12) and (50); after full expansion of its terms, it becomes

$$k_{mn}=\frac{abC_{mn}}{m^2n^2\pi^4\gamma_{mn}\epsilon_N}\frac{\beta_N\tanh(\gamma_{mn}d)+\Gamma_N}{\beta_N+\Gamma_N\tanh(\gamma_{mn}d)}$$
$$\gamma_{mn}=\sqrt{\left(\frac{m\pi}{a}\right)^2+\left(\frac{n\pi}{b}\right)^2}.$$

Hence, assuming T_ℓ is either a doping level, which results in different ϵ_ℓ, or a layer thickness d_ℓ, the sensitivity of k_{mn} with respect to T_ℓ, $\forall 0\leq\ell<N$ is computed as

$$\begin{aligned}\frac{\partial k_{mn}}{\partial T_\ell}=&\frac{abC_{mn}}{m^2n^2\pi^4\gamma_{mn}\epsilon_N}\times\frac{1}{[\beta_N+\Gamma_N\tanh(\gamma_{mn}d)]^2}\\&\times([\dot\beta_N\tanh(\gamma_{mn}d)+\dot\Gamma_N][\beta_N+\Gamma_N\tanh(\gamma_{mn}d)]\\&-[\beta_N\tanh(\gamma_{mn}d)+\Gamma_N][\dot\beta_N+\dot\Gamma_N\tanh(\gamma_{mn}d)])\end{aligned}\tag{57}$$

where the terms $\dot\Gamma_N=\partial\Gamma_N/\partial T_\ell$ and $\dot\beta_N=\partial\beta_N/\partial T_\ell$ are computed from (54) and (56). Similarly, using (54), (56), and, slightly modified, (57), expressions can be easily derived for $T_\ell=d$ or ϵ_N.

Finally, consider the sensitivity of term p_{ij} with respect to contact depth c. Expressions for term p_{ij} in presence of zero depth are shown in (12). Formulae for nonzero depth can be found in [17]. Assume that all contacts have identical depth c, then sensitivity $\partial p_{ij}/\partial c$ is computed as follows:

$$\begin{aligned}\frac{\partial p_{ij}}{\partial c}=&-\frac{2}{3}\frac{\beta_N}{ab\epsilon_N\beta_N}+\sum_{m=0}^{\infty}\sum_{n=0}^{\infty}\frac{\partial k_{mn}}{\partial c}\\&\times\frac{\left[\sin\left(m\pi\frac{a_2}{a}\right)-\sin\left(m\pi\frac{a_1}{a}\right)\right]\left[\sin\left(m\pi\frac{a_4}{a}\right)-\sin\left(m\pi\frac{a_3}{a}\right)\right]}{(a_2-a_1)(a_4-a_3)}\\&\times\frac{\left[\sin\left(n\pi\frac{b_2}{b}\right)-\sin\left(m\pi\frac{b_1}{b}\right)\right]\left[\sin\left(m\pi\frac{b_4}{b}\right)-\sin\left(m\pi\frac{b_3}{b}\right)\right]}{(b_2-b_1)(b_4-b_3)}\end{aligned}\tag{58}$$

where the term $\partial k_{mn}/\partial c$ is computed as

$$\frac{\partial k_{mn}}{\partial c}=-C_{mn}\frac{a^2b^2}{m^2n^2\pi^4}\frac{1}{2ab\epsilon_N}\tag{59}$$

where C_{mn} is defined in Section II. Due to the linearity of the DCT, it is possible to compute the sensitivity of the coefficient of potential by simply calculating $\dot k_{mn}$ and by performing the DCT on it. Several DCT's related to a variety of different depth can be stored and used for the efficient calculation of the effects of technology on a particular circuit.

REFERENCES

[1] P. Miliozzi, L. Carloni, E. Charbon, and A. L. Sangiovanni-Vincentelli, "SUBWAVE: A methodology for modeling digital substrate noise injection in mixed-signal IC's," in *Proc. IEEE Custom Integrated Circuit Conf.*, May 1996, pp. 385–388.

[2] K. Fukahori and P. R. Gray, "Computer simulation of integrated circuits in the presence of electrothermal interactions," *IEEE J. Solid-State Circuits*, vol. SC-11, pp. 834–846, Dec. 1976.

[3] L. T. Pillage and R. A. Rohrer, "Asymptotic waveform evaluation for timing analysis," *IEEE Trans. Computer-Aided Design*, vol. 9, pp. 352–366, Apr. 1990.

[4] E. Chiprout and M. S. Nakhla, *Asymptotic Waveform Evaluation and Moment Matching for Interconnect Analysis*. Boston, MA: Kluwer, 1994.

[5] N. K. Verghese, D. J. Allstot, and S. Masui, "Rapid simulation of substrate coupling effects in mixed-mode IC's," in *Proc. IEEE Custom Integrated Circuit Conf.*, May 1993, pp. 1831–1834.

[6] B. R. Stanisic, N. K. Verghese, D. J. Allstot, R. A. Rutenbar, and L. R. Carley, "Addressing substrate coupling in mixed-mode ICs: Simulation and power distribution synthesis," *IEEE J. Solid-State Circuits*, vol. 29, pp. 226–237, Mar. 1994.

[7] T. Smedes, "Substrate resistance extraction for physiscs-based layout verification," in *IEEE/PRORISC Workshop on Circuits Systems and Signal Processing*, Mar. 1993, pp. 101–106.

[8] T. Smedes, N. P. van der Meijs, and A. J. van Genderen, "Extraction of circuit models for substrate cross-talk," in *Proc. IEEE Int. Conf. Computer Aided Design*, Nov. 1995, pp. 199–206.

[9] T. Smedes, N. P. van der Meijs, A. J. van Genderen, P. J. H. Elias *et al.*, "Extraction of circuit models for substrate cross-talk," in *ESSDERC'95. Proc. 25th European Solid State Device Research Conf.*, Sept. 1995, pp. 397–400.

[10] S. Mitra, R. A. Rutenbar, L. R. Carley, and D. J. Allstot, "Substrate-aware mixed-signal macro-cell placement in WRIGHT," in *Proc. IEEE Custom Integrated Circuit Conf.*, May 1994, pp. 529–532.

[11] I. L. Wemple and A. T. Yang, "Mixed-signal switching noise analysis using Voronoi-Tessellated substrate macromodels," in *Proc. IEEE/ACM Design Automation Conf.*, June 1995, pp. 439–444.

[12] D. K. Su, M. Loinaz, S. Masui, and B. Wooley, "Experimental results and modeling techniques for substrate noise in mixed-signal integrated circuits," *IEEE J. Solid-State Circuits*, vol. 28, pp. 420–430, Apr. 1993.

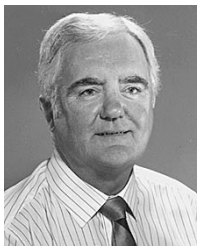
[13] K. Joardar, "A simple approach to modeling cross-talk in integrated circuits," *IEEE J. Solid-State Circuits*, vol. 29, pp. 1212–1219, Oct. 1994.
[14] J. D. Jackson, *Classical Electrodynamics*. New York: Wiley, 1975.
[15] R. Gharpurey, "Modeling and analysis of substrate coupling in IC's," Ph.D. dissertation, Univ. California Berkeley, May 1995.
[16] R. Gharpurey and R. G. Meyer, "Modeling and analysis of substrate coupling in IC's," *IEEE J. Solid-State Circuits*, vol. 31, pp. 344–353, Mar. 1996.
[17] E. Charbon, "Constraint-driven analysis and synthesis of high-performance analog IC layout," Ph.D. dissertation, Univ. California Berkeley, Dec. 1995.
[18] E. Charbon, E. Malavasi, and A. L. Sangiovanni-Vincentelli, "Generalized constraint generation for analog circuit design," in *Proc. IEEE Int. Conf. Computer Aided Design*, Nov. 1993, pp. 408–414.
[19] P. Miliozzi, I. Vassiliou, E. Charbon, E. Malavasi, and A. L. Sangiovanni-Vincentelli, "Use of sensitivities and generalized substrate models in mixed-signal IC design," in *Proc. IEEE/ACM Design Automation Conf.*, June 1996, pp. 227–232.
[20] E. Charbon, E. Malavasi, U. Choudhury, A. Casotto, and A. L. Sangiovanni-Vincentelli, "A constraint-driven placement methodology for analog integrated circuits," in *Proc. IEEE Custom Integrated Circuit Conf.*, May 1992, pp. 2821–2824.
[21] E. Charbon, E. Malavasi, D. Pandini, and A. L. Sangiovanni-Vincentelli, "Simultaneous placement and module optimization of analog IC's," in *Proc. IEEE/ACM Design Automation Conf.*, June 1994, pp. 31–35.
[22] B. R. Stanisic, R. A. Rutenbar, and L. R. Carley, "Mixed-signal noise-decoupling via simultaneous power distribution and cell customization in RAIL," in *Proc. IEEE Custom Integrated Circuit Conf.*, May 1994, pp. 533–536.
[23] U. Choudhury and A. L. Sangiovanni-Vincentelli, "Constraint generation for routing analog circuits," in *Proc. IEEE/ACM Design Automation Conf.*, June 1990, pp. 561–566.
[24] F. Romeo, "Simulated annealing: Theory and applications to layout problems," Ph.D. dissertation, Univ. California Berkeley, Mar. 1989.
[25] B. Hajek, "Cooling schedules for optimal annealing," *Mathematics of Operations Res.*, vol. 13, no. 2, pp. 311–329, May 1988.
[26] R. Neff, P. Gray, and A. L. Sangiovanni-Vincentelli, "A module generator for high speed CMOS current output digital/analog converters," in *Proc. IEEE Custom Integrated Circuit Conf.*, May 1995, pp. 481–484.
[27] I. A. Young, J. K. Greason, and K. L. Wong, "A PLL clock generator with 5 to 110 MHz of lock range for microprocessors," *IEEE J. Solid-State Circuits*, vol. 27, pp. 1599–1607, Nov. 1992.
[28] I. Vassiliou, H. Chang, A. Demir, E. Charbon, P. Miliozzi, and A. L. Sangiovanni-Vincentelli, "A video driver system designed using a top-down, constraint-driven methodology," in *Proc. IEEE Int. Conf. Computer Aided Design*, Nov. 1996, pp. 463–468.

Edoardo Charbon (S'90–M'92) received the Diploma in electrical engineering from the Swiss Federal Institute of Technology (ETH) at Zurich, Switzerland, in 1988. In 1991, he received the M.S. degree in electrical and computer engineering from the University of California at San Diego, working on telecommunication protocol analysis and optimization. He received the Ph.D. degree in 1995 from the University of California at Berkeley.

Between 1988 and 1989, he worked at the Department of Electrical Engineering of the ETH, where he designed CMOS A/D converters for integrated sensor applications. In 1989, he visited the Department of Electrical Engineering of the University of Waterloo, Canada, where he was involved in the design and fabrication of ultra low-noise, nano-Tesla magnetic sensors. In 1991 he joined the Department of Electrical Engineering and Computer Sciences at the University of California at Berkeley. At Berkeley he worked on performance-directed, constraint-based analog and mixed-signal physical design automation, substrate analysis and optimization, and advanced parasitic extraction. Since 1995 he has been with Cadence Design Systems where he is leading the effort on constraint management in the physical design group. He has published a book and over 30 articles in technical journals and conference proceedings. He has been consulting with Texas Instruments and Hewlett-Packard. His research interests include CAD for RF, analog and mixed-signal IC's, substrate modeling and characterization, RF design, microwave and superconducting parasitic analysis, and micromachined sensor design.

Ranjit Gharpurey received the Bachelor of Technology degree from the Indian Institute of Technology, Kharagpur, India, in 1990 and the M.S. and Ph.D. degrees from the University of California at Berkeley in 1992 and 1995, respectively.

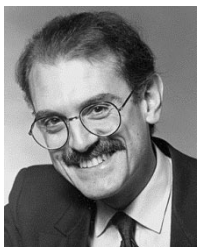
He has since been with the Mixed Signal Design branch at Texas Instruments Inc., Dallas, TX. His research interests are in the areas of RF circuit design, high-speed analog circuit design, and CAD for RF design.



Robert G. Meyer (S'64–M'68–SM'74–F'81) was born in Melbourne, Australia, on July 21, 1942. He received the B.E., M.Eng.Sci., and Ph.D. degrees in electrical engineering from the University of Melbourne in 1963, 1965, and 1968, respectively.

In 1968, he was employed as an Assistant Lecturer in Electrical Engineering at the University of Melbourne. Since September 1968, he has been employed in the Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, where he is now a Professor. His research interests are in high-frequency analog integrated-circuit design and device fabrication with emphasis on nonlinear phenomena and noise performance. He has acted as a consultant on electronic circuit design for numerous companies in the electronics industry. He is co-author of the book *Analysis and Design of Analog Integrated Circuits* (Wiley, 1993, 3rd Ed.) and Editor of the book *Integrated Circuit Operational Amplifiers* (IEEE Press, 1978).

Dr. Meyer was President of the IEEE Solid-State Circuits Council and was an Associate Editor of the IEEE JOURNAL OF SOLID-STATE CIRCUITS and of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS.



Alberto Sangiovanni-Vincentelli (M'74–SM'81–F'83) obtained the electrical engineering and computer sciences degree *summa cum laude* from the Politecnico di Milano, Milano, Italy, in 1971.

He is a Professor of Electrical Engineering and Computer Sciences at the University of California at Berkeley where he has been on the Faculty since 1976. He was Assistant Professor from 1971 to 1974 and Associate Professor from 1974 to 1976 at the Politecnico di Milano. In 1980–1981, he was Visiting Scientist at the Mathematical Sciences Department of the IBM T. J. Watson Research Center. In 1987, he was Visiting Professor at MIT. He has held a number of visiting professor positions at the Universities of Torino, Bologna, Pavia, Pisa, and Rome. In 1996 he was named General Manager of the Cadence European Laboratories and Scientific Director of the Cadence-Magneti Marelli-SGS-Thomson Research Consortium PARADES. His present research interests include design methodologies and tools for mixed signal integrated circuits with particular emphasis on RF components, for embedded controllers including software synthesis and analysis, and control algorithms for hybrid systems. He is a co-founder of Cadence and Synopsys, the two leading companies in the area of electronic design automation. He was a Director of ViewLogic and Pie Design System and Chair of the Technical Advisory Board of Synopsys. He is on the Board of Directors of Cadence, the Chair of its Scientific Advisory Council, the founder of the Cadence Berkeley Laboratories and of the Cadence European laboratories and a member of their Management Boards. He has consulted for a number of U.S., Japanese, and European companies. He is on the Advisory Board of the Lester Center of the Haas School of Business and of the Center for Western European Studies and a member of the Berkeley Roundtable of the International Economy (BRIE). He has published over 450 papers and ten books in the area of design methodologies and tools.

Dr. Sangiovanni-Vincentelli is the recipient of the 1981 Distinguished Teaching Award of the University of California, the world-wide 1995 Graduate Teaching Award of the IEEE, three Best Paper Awards (1982, 1983, and 1990) and a Best Presentation Award (1982) at the Design Automation Conference. He has also been co-recipient of the Guillemin-Cauer Award (1982–1983), the Darlington Award (1987–1988), and the Best Paper Award of the Circuits and Systems Society of the IEEE (1989–1990). He is a member of the National Academy of Engineering. He has been General Chair of the International Conference on CAD and member of the program committee of several major conferences. He has been on the editorial board of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS, the IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN, and of several other archival Journals.