Semi-Analytical Techniques for Substrate Characterization in the Design of Mixed-Signal ICs

Edoardo Charbon, Ranjit Gharpurey[†],
Robert G. Meyer[‡], and Alberto Sangiovanni-Vincentelli[‡]

Cadence Design Systems Inc., San Jose, CA [†] Texas Instruments Inc., Dallas, TX
[‡] Department of EECS, University of California, Berkeley, CA

Abstract

A number of methods are presented for highly efficient calculation of substrate current transport. A three-dimensional Green's Function based substrate representation, in combination with the use of the Fast Fourier Transform, significantly speeds up the computation of sensitivities with respect to all parameters associated with a given architecture. Substrate sensitivity analysis is used in a number of physical optimization tools, such as placement and trend analysis for the estimation of the impact of technology migration and/or layout re-design.

1 Introduction

Rapid increase of chip complexity and device density have resulted in a dramatic reduction of the distance between high-swing high-frequency noise sources and sensitive devices. In this new scenario, substrate becomes a major carrier of spurious and potentially destructive signals, especially in mixed-signal and RF designs. To alleviate the problem, heavily over-designed structures are generally laid-out, thus seriously limiting the advantages of innovative technologies.

The effects of thermally inhomogeneous substrate on circuit performance were first studied in [1]. According to this approach, the substrate was first discretized into a resistive/capacitive mesh. DC/steady-state analysis was carried out by solving directly the system of simultaneous thermal and electrical equations. Transient analysis was performed by using variable time-step trapezoidal integration techniques on the system of simultaneous equations. More efficient techniques for DC and transient analysis have been proposed in [2]. Direct LU factorization was replaced with the Incomplete Choleski Conjugate Gradient iterative method in the DC solution. In the transient simulation the RC mesh was reduced to a macro-model that could be used efficiently inside a Asymptotic Waveform Evaluation (AWE) simulator.

Extensive literature has recently appeared on modeling and analysis of silicon substrate. Compact and accurate macro-models for both lightly and heavily doped substrates have been proposed by several authors, e.g. [3, 4]. Recently, attempts to introduce the effects of substrate on medium-sized integrated circuits have been made using a numerical finite-difference (FD) method. This technique is versatile and general in nature, since it 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 non-uniform grids are often used. The grid size is made fine in areas close to the substrate contacts and coarse in distant regions. The use of non-uniform or coarse grids usually involves a speed-accuracy trade-off. Boundary element methods can also be used for parasitic extraction. In [5] the use of the Green's Function was proposed for a finite uniform medium, with Neumann 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. Despite their efficiency however, these techniques are still too computational intensive to be suited for optimization.

Traditionally, substrate-aware optimization has not been as important as substrate analysis. Substrate noise analysis has been generally addressed *a posteriori*, i.e. after completion of schematic design and physical assembly. In many design problems however, a *dynamic* substrate noise analysis would be preferable. Recently, a number of authors have addressed the problem of performing these tasks efficiently within physical assembly phases e.g. [6, 7].

Common to these approaches is the use of a FD method 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. However, these methods often cannot guarantee the accuracy needed for reliable performance estimation, due to the extremely coarse grids used. Moreover, even if dense or non-uniform grids were used, at no extra cost in computation, the alignment requirements of grid and layout objects would be very stringent. Thus, unless specific tessellation [8] or analytical approximations [4, 9] were used, iterative algorithms based on progressive and often minimal modifications could not efficiently take advantage of the algorithms.

In substrate-aware tools one has to take into account the *global* effects of small changes in the layout. Traditional approaches, consisting of solving coarse FD based analyses, may reach such inaccuracy levels that the insights gained applying this method might not be beneficial but misleading, thus possibly resulting in sub-optimal solutions.

In this paper we propose non-FD techniques for substrate analysis and optimization in analog and mixed-signal circuit applications. The methods are based on efficient computation of the Green's Function in multi-layered substrates by means of Fast Fourier Transform (FFT). A resistive network is generated accurately representing the substrate with arbitrarily-shaped doped regions. Sensitivities of all the network components with respect to a number of technology parameters are efficiently computed using semi-analytical techniques and at little cost in storage requirements.

Computing sensitivities of substrate coupling is useful for a number of reasons. First, it allows to evaluate the effects of slight imperfec-

ICCAD '96
1063-6757/96 $5.00 © 1996 IEEE

tions in the fabrication process on the performance of a circuit 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 small modifications are made on the substrate geometry and/or technology parameters *after* the design is completed. Third, the technique can be used during optimization to help the decision process providing a trend 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.

The paper is organized as follows. In section 2 the sensitivity evaluation techniques for a Green's Function based method are described. In section 3 sensitivity-based optimization is discussed for substrate-aware problems. In section 4 the suitability of the approach is illustrated with a medium-sized mixed-signal IC designed using substrate-aware techniques and fabricated on a standard CMOS process.

2 Sensitivity Analysis

In general, silicon substrate in ICs is 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. These are junctions with the substrate and may therefore be considered equipotential. Calculating resistances between any contact locations on the substrate requires the computation of electric potential $\Phi(\mathbf{r},t)$ at any location $\mathbf{r} = [x\ y\ z]^T$ in the bulk. From Maxwell's equations one can show that

$$\frac{1}{\rho}\nabla \bullet \nabla\Phi(\mathbf{r},t) + \epsilon\frac{\partial}{\partial t}(\nabla \bullet \nabla\Phi(\mathbf{r},t)) = 0 \qquad (1)$$

holds, where ϵ and ρ are respectively the local dielectric constant and resistivity of the substrate.

At frequencies upto several GHz, the substrate susceptance is typically much smaller than the substrate conductance, and hence it may be ignored. The substrate may be treated as a distributed resistance. In the electrostatic case, the problem can be related to that of distributed capacitance in a low-frequency range by replacing the current vectors with charge vectors and layer resistivities with appropriately ratioed dielectric constants. Under these assumptions (1) can be solved almost analytically by use of the Green's Function. The Green's Function in a medium with prescribed boundary conditions is the potential at any point $\mathbf{r}$ due to a point charge placed at a location $\mathbf{r}'$. Assuming zero potential in the chip's backplane, and vanishing normal electric field on the other faces, the potential due to an arbitrary charge distribution simplifies to

$$\Phi(\mathbf{r}) = \int_V \rho(\mathbf{r}')G(\mathbf{r},\mathbf{r}')dv' \ , \qquad (2)$$

where V is the chip's volume region. The potential of a contact is computed as the result of averaging all internal contact partitions. Hence, using (2) the potential of contact i due to a uniform charge Q_j present at contact j is derived as

$$\bar{\Phi}_i = \frac{Q_j}{V_j V_i}\int_{V_i}\int_{V_j} G dv_j dv_i \ , \qquad (3)$$

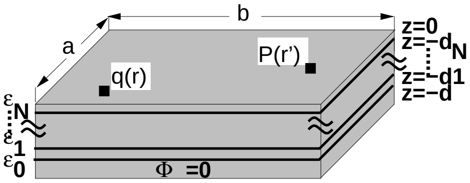

Figure 1: *Multi-layer doping profiles*

where V_i and V_j are the volumes of contacts i and j. The solution to equation (2) for each contact pairs yields the *coefficient of potential* matrix **P**. The relation between matrix **P** and vectors $\bar{\Phi}$, the average potential at each contact, and **Q**, the charge associated with all contacts, is described as

$$\bar{\Phi} = \mathbf{P}\mathbf{Q}. \qquad (4)$$

The matrix of relative capacitances between each pairs of nodes can be obtained by simple manipulations on $\mathbf{P^{-1}}$.

For isotropic medium with uniform dielectric constant ϵ the Green's Function is relatively straightforward [5]. On the contrary, for a multiple-layer substrate, the problem is more complex due to presence of more boundary conditions. Consider the case in which the point-charge at $\mathbf{r} = [x\ y\ 0]^T$ and the observation point at $\mathbf{r}' = [x'\ y'\ 0]^T$ are in the same dielectric ϵ_N. The Green's Function corresponds to

$$\begin{aligned} G(\mathbf{r},\mathbf{r}') = G_0|_{z=z'=0} + \textstyle\sum_{m=0}^{\infty}\sum_{n=0}^{\infty} f_{mn}C_{mn} \times \\ cos(\tfrac{m\pi x}{a})cos(\tfrac{m\pi x'}{a})cos(\tfrac{n\pi y}{b})cos(\tfrac{n\pi y'}{b}) \ , \end{aligned} \qquad (5)$$

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 $m, n \geq 0$. Parameters a, b and d are the dimensions of substrate in x-, y- and z-direction in Figure 1. Formulae for G_0, f_{mn}, Γ_N and β_N can be found in Appendix.

From equation (3), adapted for surface contacts one can derive an expression for the components of **P**

$$p_{ij} = \frac{\bar{\Phi}_i}{Q_j} = \frac{1}{S_i S_j}\int_{S_i}\int_{S_j} G(s_j,s_i)ds_j ds_i \ , \qquad (6)$$

where S_j and S_i are the surfaces of the contacts. Replacing (5) into (6) and integrating , one obtains an explicit form for p_{ij} [10]

$$\begin{aligned} p_{ij} = \tfrac{\Gamma_N}{ab\epsilon_N\beta_N} + \textstyle\sum_{m=0}^{\infty}\sum_{n=0}^{\infty} k_{mn} \times \\ \frac{\left[sin(m\pi\frac{a_2}{a}) - sin(m\pi\frac{a_1}{a})\right]\left[sin(m\pi\frac{a_4}{a}) - sin(m\pi\frac{a_3}{a})\right]}{(a_2-a_1)(a_4-a_3)} \times \\ \frac{\left[sin(n\pi\frac{b_2}{b}) - sin(m\pi\frac{b_1}{b})\right]\left[sin(m\pi\frac{b_4}{b}) - sin(m\pi\frac{b_3}{b})\right]}{(b_2-b_1)(b_4-b_3)}, \end{aligned} \qquad (7)$$

where $k_{mn} = \frac{a^2 b^2 f_{mn} C_{mn}}{m^2 n^2 \pi^4}$. Parameters ($a_1$, a_2; b_1, b_2) and (a_3, a_4; b_3, b_4) are the x- and y-coordinates of nodes i and j respectively.

The doubly infinite series of (7) tends to converge slowly. The problem can be eliminated by rewriting the second term of (7) after proper scaling, as a cosine series

$$\sum_{m=0}^{\tilde{P}-1}\sum_{n=0}^{\tilde{Q}-1} k_{mn}\ cos(m\pi\frac{\tilde{p}_{1,2} \pm \tilde{q}_{3,4}}{\tilde{P}})\ cos(n\pi\frac{\tilde{q}_{1,2} \pm \tilde{q}_{3,4}}{\tilde{Q}})\ . \qquad (8)$$

where the ratios of contact coordinates and substrate dimensions were replaced with ratios of integers $a_k/a = \tilde{p}_k/\tilde{P}$ and $b_k/b = \tilde{q}_k/\tilde{Q}$

and the upper limits were substituted with finite integers $\tilde{P}$ and $\tilde{Q}$ respectively. Equation (8) is a compact representation for a sum of 64 terms forming all possible combinations of signs and indices.

One can immediately see that equation (8) is almost identical to the two-dimensional Discrete Cosine Transform (DCT) of k_{mn}

$$K(\tilde{p},\tilde{q}) = \sum_{m=0}^{\tilde{P}-1}\sum_{n=0}^{\tilde{Q}-1} k_{mn} cos(m\pi\frac{\tilde{p}}{\tilde{P}})\, cos(n\pi\frac{\tilde{q}}{\tilde{Q}}). \qquad (9)$$

In fact one can easily show that equations (8) and (9) can be interchanged [11]. Hence, the computation of p_{ij} ultimately requires only a simple DCT. Several efficient 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 need be derived **only once**. Any modification in the relative position of one or more nodes is captured completely by the DCT. Non-abrupt doping profiles can be analyzed at low CPU cost by simply discretizing in z-direction with a gradually changing value of conductivity. The complete method for substrate analysis is described as following.

```
$k_{mn}$ = compute_kmn (substrate description);   // compute $k_{mn}$
compute_DCT($k_{mn}$);                             // DCT
store_DCT_values;
**foreach** contact
    contact_part = get_partitions (contact);      // get contact partitions
    integer_representation (contact_part);        // $x, y \rightarrow p, q$ coordinates
**foreach** $cp_i$ = contact_part(i)                 // all contact partitions
    **foreach** $cp_j$ = contact_part(j)
        $p_{ij}$ = compute_pij ($cp_i$, $cp_j$);        // build **P**
**C** = invert_matrix (**P**);
**R** = construct_resistive_network(**C**);
```

Algorithm 1: *Pseudo-code of the substrate resistance extraction*

The DCT of k_{mn} is computed for each location in a Manhattan grid covering the whole substrate surface. To generate matrix **P**, it is necessary to compute the parameter p_{ij} for all the pairs of partition elements composing each contact. The inversion of **P**, due to its dense nature, is the most time consuming operation of the whole algorithm. Several inversion techniques, both direct and iterative, have been implemented. Among the direct methods, a LU decomposition based algorithm of complexity $O(|\mathbf{P}|^3)$ has been used for relatively small configurations ($\leq$ 1000 partitions). Larger circuits required the use of various accuracy-driven simplification schemes, discussed in [11].

Assuming appropriate scaling of the *coefficient of induction* matrix $\mathbf{c} = \mathbf{P^{-1}}$, the conductive network **Y**, is computed as

$$Y_{ii} = \sum_{j=1}^{N} c_{ij}\ , \quad Y_{ij} = -c_{ij}\ , \quad \text{or} \quad \mathbf{Y} = \mathbf{X^T}\,\mathbf{c}\,\mathbf{X}\,, \qquad (10)$$

where Y_{ij}, the (i, j) entry of **Y**, is the mutual conductance of contacts i and j for $i \neq j$ and the substrate conductance towards the backplane, otherwise. **Y** can be also represented compactly using mapping **X**. The elements of the resistive network, denoted by matrix **R** are the reciprocal of those of matrix **Y**.

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

$$\triangle\, K \approx \sum_{\ell} \frac{\partial K}{\partial T_\ell} \triangle\, T_\ell, \text{ with } \frac{\partial K}{\partial T_\ell} = \sum_{i,j} \frac{\partial K}{\partial Y_{ij}} \frac{\partial Y_{ij}}{\partial T_\ell}\ , \qquad (11)$$

where (i, j) represent a contact pair, Y_{ij} the substrate conductive coupling between i and j, and T_ℓ a technology parameter.

Assume that $\partial K/\partial Y_{ij}$ exists and that the equivalent conductive network has been computed from matrix **c** using (10). Consider technology parameter $T_\ell, \forall \ell = 1, \ldots, N_T$. Let us define $\partial\mathbf{Y}/\partial T_\ell$ as the matrix of all sensitivities of matrix elements $\mathbf{Y_{ij}}$ with respect to T_ℓ. The terms are computed as

$$\frac{\partial Y_{ii}}{\partial T_\ell} = \sum_{j=1}^{N} \frac{\partial c_{ij}}{\partial T_\ell}, \text{ and } \frac{\partial Y_{ij}}{\partial T_\ell} = -\frac{\partial c_{ij}}{\partial T_\ell}\ , \qquad (12)$$

where N is the the size of matrix c. Differentiating (4) on both sides and using the fact that $\partial\bar{\mathbf{\Phi}}/\partial T_\ell$ vanishes by construction, one obtains

$$\frac{\partial \mathbf{Q}}{\partial T_\ell} = -\mathbf{P^{-1}} \left(\frac{\partial \mathbf{P}}{\partial T_\ell} \mathbf{Q} \right). \qquad (13)$$

Furthermore, using the definition of c_{ij}, it follows that

$$\frac{\partial c_{ij}}{\partial T_\ell} = \frac{1}{\mathbf{\Phi}_j} \frac{\partial Q_i}{\partial T_\ell}\ , \qquad (14)$$

where $\partial Q_i/\partial T_\ell$ is computed using equation (13).

Now, only the term $\partial\mathbf{P}/\partial T_\ell$, i.e. $\left[\partial p_{ij}/\partial T_\ell\right]_{i,j=1,\ldots,N}$, remains to be computed. From equation (7), assuming zero-depth contacts and $T_\ell \neq \epsilon_N,\ d,\ a$ or b

$$\frac{\partial p_{ij}}{\partial T_\ell} = \frac{\dot{\Gamma}_N \beta_N - \Gamma_N \dot{\beta}_N}{ab\epsilon_N {\beta_N}^2} + \sum_{m=0}^{\infty}\sum_{n=0}^{\infty} \dot{k}_{mn} \times$$
$$\frac{\left[sin(m\pi\frac{a_2}{a}) - sin(m\pi\frac{a_1}{a})\right]\left[sin(m\pi\frac{a_4}{a}) - sin(m\pi\frac{a_3}{a})\right]}{(a_2-a_1)(a_4-a_3)} \times \qquad (15)$$
$$\frac{\left[sin(n\pi\frac{b_2}{b}) - sin(m\pi\frac{b_1}{b})\right]\left[sin(m\pi\frac{b_4}{b}) - sin(m\pi\frac{b_3}{b})\right]}{(b_2-b_1)(b_4-b_3)},$$

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 these derivatives have been derived in Appendix. The extension of (15) for contacts with finite depth c can also be found in Appendix.

The first term of (15) can be easily calculated from the formulae in Appendix, while the second term can be efficiently computed using the DCT by replacing k_{mn} with $\dot{k}_{mn}$ in equation (9). The DCT can be computed for each location in the grid and repeated for all parameters $T_\ell, \ell = 1, \ldots, N_T$, where N_T is the number of technology parameters considered. The method is summarized hereafter.

```
$\dot{k}_{mn}$ = compute_derivative_kmn ($k_{mn}$);          // equation (27)
compute_DCT($\dot{k}_{mn}$);                               // DCT
store_DCT_values;
**foreach** $cp_i$ = contact_part(i)                       // all contact partitions
    **foreach** $cp_j$ = contact_part(j)
        $\partial p_{ij}/\partial T_\ell$ = compute_derivative_pij ($cp_i$, $cp_j$);
$\partial$**P**$/\partial T_\ell$ = compose_matrix ($\partial p_{ij}/\partial T_\ell$);       // equation (15)
$\partial$**c**$/\partial T_\ell$ = get_c_sensitivities ($\partial$**P**$/\partial T_\ell$);     // equation (14)
get_resistance_sensitivities($\partial$**c**$/\partial T_\ell$);         // equations (12)
```

Algorithm 2: *Pseudo-code of the sensitivity computation*

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 additional sensitivity requires additional NxN storage, where N is the number of points in the grid of the DCT. As an example, assume N_T = 10, i.e. ten technology parameters T_ℓ are considered, moreover assume that a grid of 1024x1024 points is used. Then, the total storage needed by our approach is 41.9 MByte, which is relatively low considering that a 1μm resolution would be achieved on a 1x1mm chip size.

3 Iterative Substrate Evaluation

Traditional approaches, consisting of solving coarse FD based analyses at each step of the optimization, may reach such inaccuracy levels that the insights gained might not be beneficial but misleading, thus possibly resulting in sub-optimal solutions. Alternatively, we propose two methods based on the Green's Function approach, exploiting the fact that small adjustments in the position and orientation of layout elements result in a small change in the potential matrix **P** [11].

The first technique is based on the *Sherman-Morrison* formula. Consider a single partition i within the moving contact, matrix **P** will only change in $2N - 1$ entries at most. In order to recompute the substrate resistive network, it suffices to compute a partial update of the coefficient of induction matrix **c**.

Let $\mathbf{P}'$ be the potential matrix associated with the new configuration. Let $\delta \mathbf{P}_{r.}$ and the rth row and $\delta \mathbf{P}_{.c}$ the cth column of **P** affected by the change, then $\mathbf{P}' = \mathbf{P} + \delta \mathbf{P}_{.c} + \delta \mathbf{P}_{r.}$. Hence, $\mathbf{P}'^{-1}$ can be computed *directly* according to the Sherman-Morrison formula as

$$\mathbf{c}' = \mathbf{P}'^{-\mathbf{1}} = \mathbf{c} + \delta \mathbf{c} \ , \ \text{with} \ \delta \mathbf{c} = \frac{\mathbf{c}_{.r}(\mathbf{c} \ \delta \mathbf{P_r})}{1 + \delta \mathbf{P_r} \ \mathbf{c}_{.r}} \ , \qquad (16)$$

where $\mathbf{c}_{.r}$ is the rth column of **c**. The impedance and admittance networks associated with a substrate configuration can be easily derived using (10). Due to the simple structure of matrix **X**, the complexity of the update is entirely dominated by the Sherman-Morrison computation and is only $O(N^2)$ compared to the $O(N^3)$ complexity required by a full **P** matrix inversion.

The second technique 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 $\mathbf{x}_1, \ldots, \mathbf{x}_{k-1}$. One can 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 **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 **R** and **Y**, contained in $[\delta \mathbf{c}]_1$. Assume that a single contact z is relocated in direction **v** by an amount $|\mathbf{v}| \to 0$. Let us define the vector $\nabla_{\mathbf{v}} \mathbf{Y}$ to be

$$\nabla_{\mathbf{v}} \mathbf{Y} = [\mathbf{A}, \mathbf{B}]^T; \ \ \mathbf{A} = \frac{\partial \mathbf{Y}}{\partial v_x} \ , \ \ \mathbf{B} = \frac{\partial \mathbf{Y}}{\partial v_y} \ , \ \ \mathbf{v} = \left(\begin{array}{c} v_x \\ v_y \end{array} \right).$$

The entries of matrix **A** are defined as $A_{ij} = \partial \mathbf{Y}_{ij} / \partial \mathbf{v}_x$, those of **B** as $B_{ij} = \partial \mathbf{Y}_{ij} / \partial \mathbf{v}_y$. 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 $\delta p_{i\pm1,j}$ as

$$\delta p_{i,j\pm1} = p_{i,j\pm1} - p_{i,j} \ , \ \text{and} \ \ \delta p_{i\pm1,j} = p_{i\pm1,j} - p_{i,j} \ . \qquad (17)$$

Then, each component $\partial Y_{ij} / \partial v_x$ is calculated by replacing term c_{ij} with $\delta c_{i,j+1}$ in (10). Notice that term $\delta c_{i,j+1}$ is derived directly from matrix **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 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 0th step of our incremental algorithm. Call $\left[\partial \mathbf{Y} / \partial v_x\right]_0$ and $\left[\partial \mathbf{Y} / \partial v_y\right]_0$ these matrices.

Assuming that the moving partition, contact or collection of contacts remains *close enough* to its position of step 0, then the conductance matrix at steps $1 \leq n \leq k$ can be approximated as

$$\left[\mathbf{Y}\right]_n \approx \left[\mathbf{Y}\right]_0 + \left[\frac{\partial \mathbf{Y}}{\partial v_x}\right]_0 \triangle v_x + \left[\frac{\partial \mathbf{Y}}{\partial v_y}\right]_0 \triangle v_y = \left[\nabla_{\mathbf{v}} \mathbf{Y}^{\mathbf{T}}\right]_0 \mathbf{v}, \qquad (18)$$

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

The expression in (7) is well-behaved for all finite-sized contacts in the workspace [11]. Hence, the terms will necessarily $\delta p_{i,j\pm1} < \infty$ and $\delta p_{i\pm1,j} < \infty$. In fact, in our experiments the method has shown a 1% accuracy when the move occurred in the vicinity of the position at step 0, while a 10% accuracy was reached when the move was upto one fifth of the chip size.

These techniques have been used during the evaluation phase in a Simulated Annealing (SA) based placement tool called PUPPY-A [12]. The problem of evaluating substrate effects on performance was approached in the following way.

1. generate a model of switching noise injection
2. generate constraints for each noise-sensitive node
3. generate resistive network associated with substrate
4. quantify violations to constraints

For each module j a noise injection model is created, taking into account both impact ionization and capacitive coupling through devices and interconnect lines. The model $V_S(\mathcal{P}_j)$ is based on a bank of independent current noise generators with a unified set of parameters represented by vector $\mathcal{P}_j$. Then, the sensitivity of a given performance K_i is computed with respect to the parameters $\mathcal{P}_j$ related to each noise source j acting on every node in the analog modules being placed. Using constrained optimization techniques [13] and the specification on the maximum positive and negative performance degradation $\overline{\triangle K_i}^{+/-}$, a set of bounds $\mathcal{P}_j^{(bound)}$ is generated only for a reduced set of *critical nodes* n_c. The set n_c is generated based on the cumulative effect of all parasitic noise sources acting on each node similarly as in [13].

In step 3 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. The calculation of all violations in step 4 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 4 and 3 need be repeated, since steps 1 and 2 are carried out only once for each chip.

In SA, at high annealing temperatures, i.e. at the beginning of the cooling, 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 lower amounts. This observation leads us to the following heuristics for the evaluation of substrate effects after each tentative annealing move.

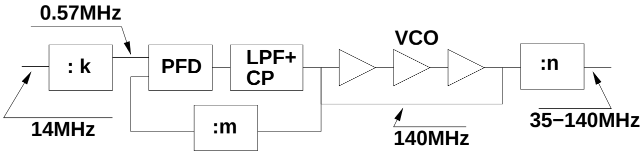

Figure 2: *PLL schematic*

```
foreach temperature $T_k$
    evaluate_substrate_network_exactly;          // Sherman-Morrison update
    repeat
        $m_k$ = select_move;
        estimate_network_change($m_k$);           // gradient-based update
        foreach node $j \in n_c$
            $\mathcal{P}_j$ = estimate_cumulative_noise;
            if $\mathcal{P}_j \geq \mathcal{P}_j^{(bound)}$
                evaluate_substrate_network_exactly;
            go_to_next_node;
        evaluate_cost_function;
        accept_or_reject_move;
    until equilibrium_reached
```

Algorithm 3: *Combined use of Sherman-Morrison and gradient-based methods*

After a new move m_k and the associated translation $\mathbf{v} = [\triangle x, \triangle y]^T$ is selected by the annealing algorithm, the sensitivity of the edges of $G_S(V, E)$ can be efficiently computed. Suppose the set n_c of all critical receipting nodes has been derived for the circuit, moreover let n_s be the set 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 $[\triangle \mathbf{Y_C}]_{m_k}$ be its update. Term $[\triangle \mathbf{Y_C}]_{m_k}$ is estimated using equation (18), where $[\triangle \mathbf{Y_C}]_{m_k} = [\mathbf{Y}]_n$. After updating $\mathbf{Y_c}$, the resistive network is solved and parameter $\mathcal{P}_j$ can be evaluated for all critical nodes j. By comparing $\mathcal{P}_j$ with the bound $\mathcal{P}_j^{(bound)}$ 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 [12]. Otherwise, the contribution of substrate noise to node j in degrading performance K_i is considered negligeable 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 and use to drive the annealing similarly as in [12].

4 Results

In this section experiments are presented using the techniques discussed in this paper, implemented in a C package called SUBRES. The circuit used in our experiments, a 140 MHz monitor display controller (RAMDAC) integrated in 1μm CMOS technology, includes three D/A converters, a Phase Lock Loop (PLL) frequency synthesizer, and digital control logic. The control logic blocks and converters were generated using standard and dedicated silicon compilers [14]. 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 Figure 2, was derived from [15]. Device sizing was performed using a modified version of the supporting hyperplane algorithm and SPICE for circuit evaluation. 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 Figure 2.

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

Table 1: PLL specifications

The specifications for the PLL are summarized in Table 1. The jitter $\frac{\triangle T}{T}$ is defined as the ratio between the variation from nominal of oscillation period $\triangle T$ and period T. Due to the time-variance of $\frac{\triangle T}{T}$, it is generally measured in terms of its peak-to-peak or RMS value.

Physical Design

First, the various components of the PLL were generated using dedicated module generator VCOGEN for the analog sections of the layout and TIMBERWOLFSC-4.1for the digital ones. The module generation step required a total of 163 seconds on a DEC AlphaServer 2100 5/250. Then, the various blocks of the PLL were placed and routed along with the other circuits of the RAMDAC. The placement was carried out using PUPPY-A.

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 critical locations in the VCO. Then, constraint generator PARCAR [13] was used to derive a set of constraints on the maximum admissible noise voltage in each one of the 85 sensitive nodes of the VCO. The CPU time needed for the constraint calculation was 2545 seconds. 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 [16]. SUBWAVE generates simplified substrate noise models, accounting for the current injected via capacitive coupling by power and ground busses, connected to the supply though inductive bonding wiring, as well as current due to impact ionization and capacitive coupling from

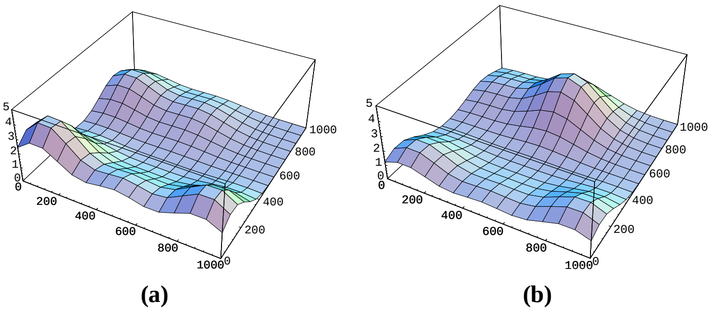

Figure 3: *Estimated switching noise signal amplitude resulting from cumulative divider injection during SA: (a) high; (b) low temperature*

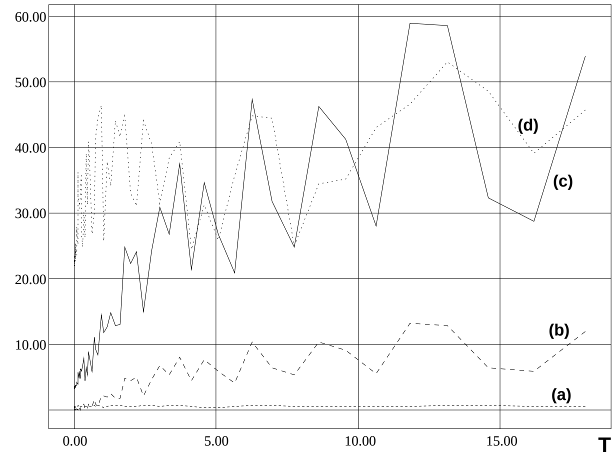

Figure 4: *Error in substrate injection estimation using: (a) combined heuristic; (b) gradient-based method only. All substrate violations using: (c) combined heuristic; (d) no substrate control*

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 2: *Placement statistics obtained on a DEC AlphaServer 2100 5/250*

active device areas.

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 presented in Algorithm 3. The constraints 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 that proposed in [12]. Figure 3 shows the estimated values of switching noise voltage at each location in the chip at different temperatures during the annealing. As expected, the annealing attempted to reduce the switching noise amplitude at critical receptors locations in the VCO, CP and LPF.

Plot 4 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. Sherman-Morrison update. Curve (c) and (d) show how the constraint violation is driven towards zero or not whether or not the proposed substrate injection control is used. Figure 5 shows the final placement performed using PUPPY-A. As expected the 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 can be placed consequently. For divider m the placer had to perform a trade-off between the strength of the switching noise received by it and the parasitics introduced when large interconnect capacitances are introduced. The same performance model used for the constraint derivation, along with the noise estimation techniques outlined in section 2 was used to predict the jitter performance in the PLL at the end of the annealing. See Table 2.

Trend Analysis

For the PLL, all the potential sources of switching noise 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 a N-well processes) and by capacitive coupling through junctions and interconnect. Receptors are in the active areas of sensitive devices and supply lines. Table 3 lists the main sources and receptors of noise in the various components of the design. Using (11) and the sensitivity information $\partial K/\partial Y_{ij}$, performance degradation $\triangle K_i$ can be efficiently computed for PLL, due to small changes in the design and/or in the technology. Suppose one were interested in deriving the trend of the jitter performance if a new lightly doped substrate were to be used instead of the low-resistivity one for which the circuit was designed. Plot 6 shows the values of the sensitivities of one entry of **R** at various nominal doping levels ($t_1, \ldots, t_3$).

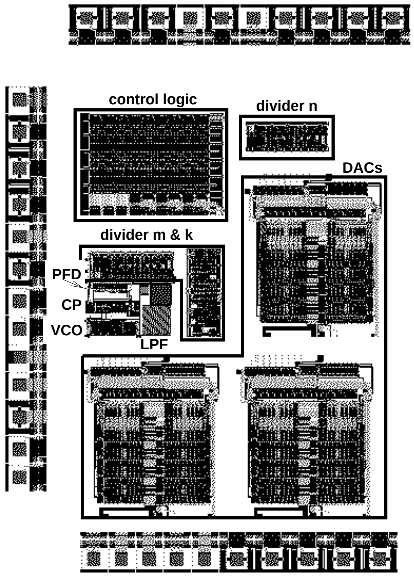

Figure 5: *Placed PLL within the* RAMDAC

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

Table 3: *Noise injector and receptor statistics in the components of the PLL*

Suppose now we were looking at the effects of contact depth c. Assuming that all contacts have similar low-resistivity substrate depth, one can use the expression (28) in Appendix. Plot 7 shows the corresponding sensitivities ($t_1, \ldots, t_4$).

Finally, let us consider the effects of changes in the doping profiles in Figure 1. Assume that the number of layers stays constant but the epitaxial layer expands towards the ground-plane. Plot 8 shows the sensitivity values (t_1, t_2). Table 4 reports the CPU times for the sensitivity analysis performed in the various experiments and the estimated trend of jitter performance degradation computed using (11).

5 Conclusions

Novel techniques based on a Green's Function approach to substrate analysis have been proposed for efficient evaluation of parasitic

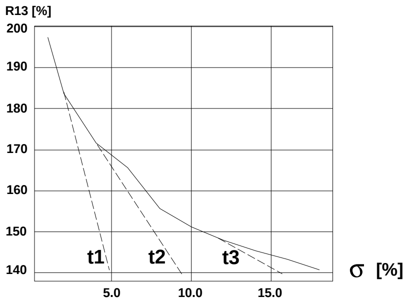

Figure 6: *Dependence from doping levels*

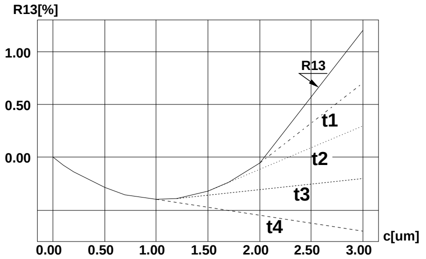

Figure 7: *Dependence from contact depth*

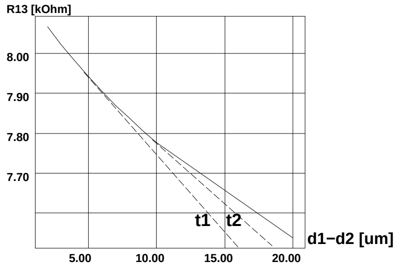

Figure 8: *Dependence from doping profiles*

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

Table 4: *CPU times for trend analysis on a DEC AlphaServer 2100 5/250 with 297 noise sources / receptors.*

switching noise injection. Sensitivity analysis is the basis for the characterization of performance in complex mixed-signal circuits. Efficient sensitivity analysis of the various substrate parasitics with respect to technology parameters are used for making a trend analysis on the effects of technology migration and scaling. The suitability of the approach has been demonstrated through a medium sized mixed-signal IC on which a complete analysis of the impact of substrate was performed.

Appendix

Terms G_0, Γ_N and β_N are computed recursively as

$$G_0 = \frac{1}{ab\epsilon_N}\frac{\Gamma_N}{\beta_N}\ ; \quad \begin{bmatrix} \beta_k \\ \Gamma_k \end{bmatrix} = \begin{bmatrix} \frac{\epsilon_{k-1}}{\epsilon_k} & 0 \\ (\frac{\epsilon_{k-1}}{\epsilon_k} - 1)d_k & 1 \end{bmatrix} \begin{bmatrix} \beta_{k-1} \\ \Gamma_{k-1} \end{bmatrix}, \quad (19)$$

with $\beta_0 = 1,\ \Gamma_0 = d$. Term f_{mn} is computed as follows

$$f_{mn} = \frac{1}{ab\gamma_{mn}\epsilon_N}\frac{\beta_N tanh(\gamma_{mn}d) + \Gamma_N}{\beta_N + \Gamma_N tanh(\gamma_{mn}d)}\ ;\ \gamma_{mn} = \sqrt{(\frac{m\pi}{a})^2 + (\frac{n\pi}{b})^2}\ . \quad (20)$$

Terms β_N and Γ_N for $m \neq 0$ or $n \neq 0$ are computed recursively as

$$\begin{bmatrix} \beta_k \\ \Gamma_k \end{bmatrix} = \mathbf{A}_k \begin{bmatrix} \beta_{k-1} \\ \Gamma_{k-1} \end{bmatrix}, \text{with} \quad (21)$$

$$\mathbf{A}_k = \begin{bmatrix} \frac{\epsilon_{k-1}}{\epsilon_k} cosh^2\theta_k - sinh^2\theta_k & (\frac{\epsilon_{k-1}}{\epsilon_k} - 1)\, cosh\theta_k sinh\theta_k \\ (1 - \frac{\epsilon_{k-1}}{\epsilon_k})\, cosh\theta_k sinh\theta_k & cosh^2\theta_k - \frac{\epsilon_{k-1}}{\epsilon_k} sinh^2\theta_k \end{bmatrix},$$

where $1 \leq k \leq N$, $\theta_k = \gamma_{mn}(d - d_k)$, $\beta_0 = 1$, and $\Gamma_0 = 0$. For the computation of sensitivities, let us consider first parameters Γ_N and β_N as defined in equation (21). Assume $T_\ell = \epsilon_\ell$, then all Γ_k and β_k will not depend on ϵ_ℓ when $0 \leq k < \ell$, hence $[\frac{\partial\beta_k}{\partial T_\ell}\,\frac{\partial\Gamma_k}{\partial T_\ell}]^T = \mathbf{0}\ ,\ \ \forall\ 0 \leq k < \ell$. Consider first the case in which $k = \ell$. Equation (21) becomes

$$\begin{bmatrix} \frac{\partial\beta_\ell}{\partial T_\ell} \\ \frac{\partial\Gamma_\ell}{\partial T_\ell} \end{bmatrix} = \frac{\epsilon_{\ell-1}}{\epsilon_\ell^2}\dot{\mathbf{A}}_\ell \begin{bmatrix} \beta_{\ell-1} \\ \Gamma_{\ell-1} \end{bmatrix}, \text{with} \quad (22)$$

$$\dot{\mathbf{A}}_\ell = \begin{bmatrix} -cosh^2\theta_\ell & -cosh\theta_\ell\ sinh\theta_\ell \\ cosh\theta_\ell\ sinh\theta_\ell & sinh^2\theta_\ell \end{bmatrix}.$$

Terms $\Gamma_{\ell-1}$ and $\beta_{\ell-1}$ are already known, while $\theta_\ell = \gamma_{mn}(d - d_\ell)$ and $\gamma_{mn} = \sqrt{(\frac{m\pi}{a})^2 + (\frac{n\pi}{b})^2}$.
Secondly, consider the case in which $k = \ell + 1$. Equation (21) becomes

$$\begin{bmatrix} \frac{\partial\beta_{\ell+1}}{\partial T_\ell} \\ \frac{\partial\Gamma_{\ell+1}}{\partial T_\ell} \end{bmatrix} = \frac{1}{\epsilon_{\ell+1}}\mathbf{A}' \begin{bmatrix} \beta_\ell \\ \Gamma_\ell \end{bmatrix} + \mathbf{A}'' \begin{bmatrix} \frac{\partial\beta_\ell}{\partial T_\ell} \\ \frac{\partial\Gamma_\ell}{\partial T_\ell} \end{bmatrix}, \text{with} \quad (23)$$

$$\mathbf{A}' = \begin{bmatrix} cosh^2\theta_{\ell+1} & cosh\theta_{\ell+1}\ sinh\theta_{\ell+1} \\ -cosh\theta_{\ell+1}\ sinh\theta_{\ell+1} & -sinh^2\theta_{\ell+1} \end{bmatrix} \text{and}$$

$$\mathbf{A}'' = \begin{bmatrix} \frac{\epsilon_\ell}{\epsilon_{\ell+1}} cosh^2\theta_{\ell+1} - sinh^2\theta_{\ell+1} & (\frac{\epsilon_\ell}{\epsilon_{\ell+1}} - 1)\, cosh\theta_{\ell+1}\ sinh\theta_{\ell+1} \\ (1 - \frac{\epsilon_\ell}{\epsilon_{\ell+1}})\, cosh\theta_{\ell+1}\ sinh\theta_{\ell+1} & cosh^2\theta_{\ell+1} - \frac{\epsilon_\ell}{\epsilon_{\ell+1}} sinh^2\theta_{\ell+1} \end{bmatrix}.$$

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

$$\begin{bmatrix} \frac{\partial\beta_k}{\partial T_\ell} \\ \frac{\partial\Gamma_k}{\partial T_\ell} \end{bmatrix} = \dot{\mathbf{A}}_k \begin{bmatrix} \frac{\partial\beta_{k-1}}{\partial T_\ell} \\ \frac{\partial\Gamma_{k-1}}{\partial T_\ell} \end{bmatrix}, \quad (24)$$

$$\dot{\mathbf{A}}_k = \begin{bmatrix} \frac{\epsilon_{k-1}}{\epsilon_k} cosh^2\theta_k - sinh^2\theta_k & (\frac{\epsilon_{k-1}}{\epsilon_k} - 1)\, cosh\theta_k\ sinh\theta_k \\ (1 - \frac{\epsilon_{k-1}}{\epsilon_k})\, cosh\theta_k\ sinh\theta_k & cosh^2\theta_k - \frac{\epsilon_{k-1}}{\epsilon_k} sinh^2\theta_k \end{bmatrix},$$

where $\partial\beta_{k-1}/\partial T_\ell$ and $\partial\Gamma_{k-1}/\partial T_\ell$ are obtained directly from equation (23). The recursion (24) 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. Similarly as before, consider first the case in which $k = \ell$. Equation (21) becomes

$$\begin{bmatrix} \frac{\partial\beta_\ell}{\partial T_\ell} \\ \frac{\partial\Gamma_\ell}{\partial T_\ell} \end{bmatrix} = -\gamma_{mn}\ \left(\frac{\epsilon_{\ell-1}}{\epsilon_\ell} - 1\right) \dot{\mathbf{A}}_\ell \begin{bmatrix} \beta_{\ell-1} \\ \Gamma_{\ell-1} \end{bmatrix}, \text{with} \quad (25)$$

$$\dot{\mathbf{A}}_\ell = \begin{bmatrix} 2\ sinh\theta_\ell\ cosh\theta_\ell & cosh2\theta_\ell \\ -cosh2\theta_\ell & -2\ sinh\theta_\ell\ cosh\theta_\ell \end{bmatrix},$$

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

$$\begin{bmatrix} \frac{\partial\beta_k}{\partial T_\ell} \\ \frac{\partial\Gamma_k}{\partial T_\ell} \end{bmatrix} = \dot{\mathbf{A}}_k = \begin{bmatrix} \frac{\partial\beta_{k-1}}{\partial T_\ell} \\ \frac{\partial\Gamma_{k-1}}{\partial T_\ell} \end{bmatrix}, \text{with} \quad (26)$$

$$\dot{\mathbf{A}}_k = \begin{bmatrix} \frac{\epsilon_{k-1}}{\epsilon_k} cosh^2\theta_k - sinh^2\theta_k & (\frac{\epsilon_{k-1}}{\epsilon_k} - 1)\, cosh\theta_k \; sinh\theta_k \\ (1 - \frac{\epsilon_{k-1}}{\epsilon_k})\, cosh\theta_k \; sinh\theta_k & cosh^2\theta_k - \frac{\epsilon_{k-1}}{\epsilon_k} sinh^2\theta_k \end{bmatrix},$$

where $\partial\beta_{k-1}/\partial T_\ell$ and $\partial\Gamma_{k-1}/\partial T_\ell$ are obtained directly from equation (25). The recursion (26) 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 equations (7) and (20); after full expansion of its terms, it becomes

$$k_{mn} = \frac{ab\, C_{mn}}{m^2n^2\pi^4\gamma_{mn}\epsilon_N} \frac{\beta_N tanh\gamma_{mn} d + \Gamma_N}{\beta_N + \Gamma_N tanh\gamma_{mn} d},$$
$$\text{with } \gamma_{mn} = \sqrt{(\tfrac{m\pi}{a})^2 + (\tfrac{n\pi}{b})^2}\,.$$

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

$$\frac{\partial k_{mn}}{\partial T_\ell} = \frac{ab\, C_{mn}}{m^2n^2\pi^4\gamma_{mn}\epsilon_N} \frac{1}{[\beta_N + \Gamma_N tanh(\gamma_{mn} d)]^2} \times$$
$$[\dot{\beta}_N tanh(\gamma_{mn} d) + \dot{\Gamma}_N][\beta_N + \Gamma_N tanh(\gamma_{mn} d)] -$$
$$[\beta_N tanh(\gamma_{mn} d) + \Gamma_N][\dot{\beta}_N + \dot{\Gamma}_N tanh(\gamma_{mn} d)], \quad (27)$$

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 equations (24) and (26). Similarly, using (24), (26) and, slightly modified, (27), 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 of term p_{ij} for non-zero depth, derived in [11], are reported here.

$$p_{ij} = \frac{1}{c_2 c_4 ab\epsilon_N \beta_N} \left(-\beta_N (\frac{c_s c_g^2}{2} + \frac{c_s^3}{6}) + c_2 c_4 \Gamma_N \right) +$$
$$\sum_{m=0}^{\infty} \sum_{n=0}^{\infty} \frac{\left[sin(m\pi \frac{a_2}{a}) - sin(m\pi\frac{a_1}{a})\right]\left[sin(m\pi\frac{a_4}{a}) - sin(m\pi\frac{a_3}{a})\right]}{(a_2 - a_1)(a_4 - a_3)} \times$$
$$k_{mn} \frac{\left[sin(n\pi\frac{b_2}{b}) - sin(m\pi\frac{b_1}{b})\right]\left[sin(m\pi\frac{b_4}{b}) - sin(m\pi\frac{b_3}{b})\right]}{(b_2 - b_1)(b_4 - b_3)}, \quad (28)$$

where the term k_{mn} is calculated as

$$k_{mn} = C_{mn} \frac{a^2b^2}{m^2n^2\pi^4} \left(f_{mn} - \frac{c_s c_g^2}{2ab\epsilon_N c_2 c_4} \right), \quad (29)$$
$$\text{with } c_g = max(c_2, c_4) \text{ and } c_s = min(c_2, c_4).$$

Assume that all contacts have identical depth c, then sensitivity $\partial p_{ij}/\partial c$ is computed as follows

$$\frac{\partial p_{ij}}{\partial c} = -\frac{2}{3} \frac{\beta_N}{ab\epsilon_N \beta_N} +$$
$$\sum_{m=0}^{\infty} \sum_{n=0}^{\infty} \frac{\left[sin(m\pi\frac{a_2}{a}) - sin(m\pi\frac{a_1}{a})\right]\left[sin(m\pi\frac{a_4}{a}) - sin(m\pi\frac{a_3}{a})\right]}{(a_2 - a_1)(a_4 - a_3)} \times$$
$$\frac{\partial k_{mn}}{\partial c} \frac{\left[sin(n\pi\frac{b_2}{b}) - sin(m\pi\frac{b_1}{b})\right]\left[sin(m\pi\frac{b_4}{b}) - sin(m\pi\frac{b_3}{b})\right]}{(b_2 - b_1)(b_4 - b_3)}, \quad (30)$$

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

$$\frac{\partial k_{mn}}{\partial c} = \frac{\partial k_{mn}}{\partial c} = -C_{mn} \frac{a^2b^2}{m^2n^2\pi^4} \frac{1}{2ab\epsilon_N}, \quad (31)$$

where C_{mn} is defined in section 2. 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. ■

References

[1] K. Fukahori and P. R. Gray, "Computer Simulation of Integrated Circuits in the Presence of Electrothermal Interactions", *IEEE Journal of Solid State Circuits*, vol. SC-11, pp. 834–846, December 1976.

[2] S.-S. Lee and D. J. Allstot, "Electrothermal Simulation of Integrated Circuits", *IEEE Journal of Solid State Circuits*, vol. SC-28, n. 12, pp. 1283–1293, December 1993.

[3] T. A. Johnson, R. W. Knepper, V. Marcello and W. Wang, "Chip Substrate Resistance Modeling Technique for Integrated Circuit Design", *IEEE Trans. on CAD*, vol. CAD-3, pp. 126–134, 1984.

[4] D. K. Su, M. Loinaz, S. Masui and B. Wooley, "Experimental Results and Modeling Techniques for Substrate Noise in Mixed-Signal Integrated Circuits", *IEEE Journal of Solid State Circuits*, vol. SC-28, n. 4, pp. 420–430, April 1993.

[5] T. Smedes, "Substrate Resistance Extraction for Physiscs-Based Layout Verification", in *IEEE/PRORISC Workshop on CSSP*, pp. 101–106, March 1993.

[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 Journal of Solid State Circuits*, vol. SC-29, n. 3, pp. 226–237, March 1994.

[7] S. Mitra, R. A. Rutenbar, L. R. Carley and D. J. Allstot, "Substrate-Aware Mixed-Signal Macro-Cell Placement in WRIGHT", in *Proc. IEEE CICC*, pp. 529–532, May 1994.

[8] I. L. Wemple and A. T. Yang, "Mixed-Signal Switching Noise Analysis Using Voronoi-Tessellated Substrate Macromodels", in *Proc. IEEE/ACM DAC*, pp. 439–444, June 1995.

[9] K. Joardar, "A Simple Approach to Modeling Cross-Talk in Integrated Circuits", *IEEE Journal of Solid State Circuits*, vol. SC-29, n. 10, pp. 1212–1219, October 1994.

[10] R. Gharpurey and R. G. Meyer, "Modeling and Analysis of Substrate Coupling in ICs", *IEEE Journal of Solid State Circuits*, vol. SC-31, n. 3, pp. 344–353, March 1996.

[11] R. Gharpurey, *Modeling and Analysis of Substrate Coupling in ICs*, PhD thesis, University of California at Berkeley, May 1995.

[12] 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 CICC*, pp. 2821–2824, May 1992.

[13] U. Choudhury and A. L. Sangiovanni-Vincentelli, "Constraint Generation for Routing Analog Circuits", in *Proc. IEEE/ACM DAC*, pp. 561–566, June 1990.

[14] R. Neff, P. Gray and A. L. Sangiovanni-Vincentelli, "A Module Generator for High Speed CMOS Current Output Digital/Analog Converters", in *Proc. IEEE CICC*, pp. 481–484, May 1995.

[15] 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 Journal of Solid State Circuits*, vol. SC-27, n. 11, pp. 1599–1607, November 1992.

[16] P. Miliozzi, L. Carloni, E. Charbon and A. L. Sangiovanni-Vincentelli, "SUBWAVE: a Methodology for Modeling Digital Substrate Noise Injection in Mixed-Signal ICs", in *Proc. IEEE CICC*, pp. 385–388, May 1996.