Contact Merging Algorithm for Efficient Substrate Noise Analysis in Large Scale Circuits

Emre Salman
Renatas Jakushokas
Eby G. Friedman
Department of Electrical
and Computer Engineering
University of Rochester
Rochester, New York 14627 USA
[salman,jakushok,friedman]@ece.rochester.edu

Radu M. Secareanu
Olin L. Hartin
MMSTL
Freescale Semiconductor
Tempe, Arizona 85284 USA
[r54143,lee.hartin@freescale.com]

ABSTRACT

A methodology is proposed to efficiently estimate the substrate noise generated by large scale aggressor circuits. Small spatial voltage differences within the ground distribution network of an aggressor circuit are exploited to reduce the overall number of input ports *before* the substrate extraction process. Specifically, the substrate of an aggressor circuit is partitioned into voltage domains where each domain is represented by a single substrate contact. The remaining ports of the substrate within that domain are ignored to reduce the computational complexity. A linear time algorithm is developed to identify these voltage domains and generate an equivalent contact. A reduction of more than four orders of magnitude in the number of extracted substrate resistances is demonstrated while introducing 20% error in the peak-to-peak value of the substrate noise voltage.

Categories and Subject Descriptors

B.7.m [**Integrated Circuits**]: Miscellaneous—*Mixed signal circuits, substrate coupling noise, noise analysis*

General Terms

Algorithms, design, verification

1. INTRODUCTION

Substrate noise coupling continues to be a primary concern in mixed-signal circuits such as a transceiver where digital and analog/RF functions are placed on the same monolithic substrate. The demand for higher integration exacerbates this issue due to the reduced physical distances between the aggressor digital and sensitive analog/RF blocks.

This research is supported in part by the Semiconductor Research Corporation under Contract No. 2004-TJ-1207, the National Science Foundation under Contract No. CCF-0541206, grants from the New York State Office of Science, Technology & Academic Research to the Center for Advanced Technology in Electronic Imaging Systems, and by grants from Intel Corporation, Eastman Kodak Company, and Freescale Semiconductor Corporation.

GLSVLSI'09, May 10–12, 2009, Boston, Massachusetts, USA.

A variety of noise reduction and isolation techniques exist to alleviate substrate noise coupling. The evaluation of these techniques and quantification of the substrate noise at the boundary of the sensitive circuit require a computationally efficient analysis methodology which simultaneously considers the circuit activity, power/ground network, and substrate network.

Existing substrate network extraction techniques fail when analyzing large scale circuits due to increasing computational complexity, prohibiting the efficient estimation of the substrate noise. A methodology is proposed in this paper to reduce the computational complexity of existing substrate extraction techniques by reducing the overall number of input ports. The methodology is based on identifying voltage domains within the substrate of the aggressor circuit. A voltage domain represents a region within the substrate that is biased with approximately the same voltage by substrate contacts, and is therefore shorted by the ground network. These voltage domains are determined from the transient voltage differences among the substrate contacts, where each domain is represented by a single equivalent contact, thereby reducing the overall number of input ports for extraction.

The rest of the paper is organized as follows. In Section 2, existing substrate modeling approaches are summarized and the concept of voltage domains on the substrate is introduced. The proposed methodology and algorithm are described in Section 3. Simulation results validating the algorithm are provided in Section 4, followed by a discussion in Section 5. Finally, the paper is concluded in Section 6.

2. PROBLEM FORMULATION

Existing substrate modeling techniques and associated limitations are reviewed in Section 2.1. The process of identifying the *voltage domains* across the substrate in order to reduce the computational complexity is introduced in Section 2.2.

2.1 Existing Substrate Modeling Approaches

Current approaches to model the substrate can be divided into two classes. The first class includes those techniques that discretize the substrate into a 3-D $R(C)$ mesh to determine the impedances such as the finite difference method (FDM) [1], [2] and the boundary element method (BEM) [3], [4]. The substrate volume can be discretized in *differential* form using FDM, resulting in a huge, sparse matrix. Although the non-uniformities distributed throughout the substrate can be included using FDM [5], the overall accuracy is a strong function of the discretization resolution, making the extraction of bulk-type substrates challenging [6].

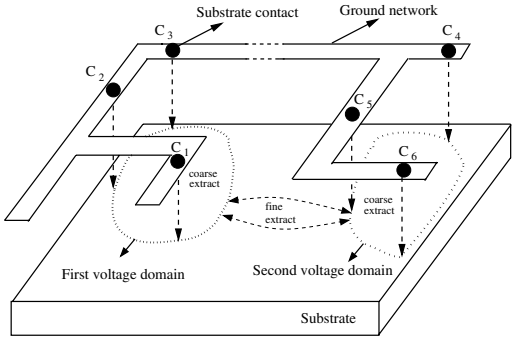

Figure 1: Identifying the voltage domains within the substrate. Assuming $V_{C1} \approx V_{C2} \approx V_{C3}$ and $V_{C4} \approx V_{C5} \approx V_{C6}$, two voltage domains are created by the first and last three contacts. A coarse extraction is performed within each domain to reduce the computational complexity, followed by a fine extraction of those domains where the dominant current flow occurs.

Alternatively, the substrate volume can be discretized in *integral* form using BEM with an appropriate Green's Function. For BEM, the size of the resulting matrix is significantly smaller, yet highly dense, as compared to FDM, since BEM only discretizes the ports into the substrate. As such, BEM does not consider the non-uniformity of the substrate as with channel stop implants.

Several different techniques have been proposed to obtain a more efficient solution of the algebraic equations produced by FDM or BEM to reduce an *RC* network such as moment matching techniques [7], a fast Fourier transform algorithm [8], a fast eigendecomposition technique [5], a numerically stable Green Function [9], and a combination of BEM and FEM techniques [10]. The primary limitation of these approaches, however, is the increase in computational complexity with the size of the circuit, prohibiting the efficient analysis of large scale mixed-signal circuits.

The second class of substrate modeling methods is the use of macromodels to represent the impedance between two ports on a substrate [11], [12]. Although computationally more efficient as compared to FDM and BEM, only limited accuracy can be achieved. Other limitations of these macromodels are the requirement to use process-dependent fitting parameters obtained through empirical data and scaling these models for smaller geometries.

A methodology is proposed in this paper to improve the computational complexity of the substrate extraction process by reducing the number of input ports of the aggressor circuit. The number of input ports is reduced *before* initiating the extraction process by exploiting the small spatial voltage differences within the ground network of the aggressor circuit, as described in Section 2.2.

2.2 Voltage Domains on the Substrate

In a mixed-signal circuit, a common approach to bias the substrate of a digital block is to connect the substrate to the digital ground network with substrate contacts. Due to the parasitic impedance of the ground network, each substrate contact has an $IR + L\,\partial i/\partial t$ voltage bounce which resistively couples into the substrate. As such, if the voltage variation between a set of substrate contacts is sufficiently small, the corresponding area of the substrate is effectively short-circuited by these contacts, as illustrated in Fig. 1.

The transient voltage difference between two contacts C_1 and C_2 is determined by

$$V_{C1} - V_{C2} = V_{12} = i(t)_{12} R_{12} + L_{12} \frac{\partial i(t)_{12}}{\partial t}, \qquad (1)$$

where $i(t)_{12}$ is the transient current on the ground network flowing from C_1 to C_2 injected by the switching gates. R_{12} and L_{12} are, respectively, the parasitic resistance and inductance of the ground network between C_1 and C_2. Referring to Fig. 1, the transient voltage difference among the contacts C_1, C_2, and C_3 and among C_4, C_5, and C_6 are assumed to be sufficiently small such that $V_{C1} \approx V_{C2} \approx V_{C3}$ and $V_{C4} \approx V_{C5} \approx V_{C6}$, respectively. As a result, the corresponding area biased by the first three contacts determines the first voltage domain on the substrate and, similarly, the last three contacts determine the second voltage domain. Since the voltage variations within a domain are sufficiently small, the dominant current flow occurs *among* these voltage domains. The small spatial voltage differences within the ground network can therefore be exploited by coarsely extracting each of these domains and applying a fine extraction among the domains.

An algorithm is described in this paper to identify these voltage domains on the substrate. An equivalent contact is created for each domain while neglecting the other ports within that domain. The number of input ports is reduced, significantly improving the computational complexity of the extraction process.

3. CONTACT MERGING METHODOLOGY

The proposed methodology for efficiently estimating the substrate noise generated by an aggressive digital circuit consists of five steps, as described below.

Step 1. The ground network of the aggressor circuit is extracted to obtain the parasitic resistance and inductance between each substrate contact. This extraction can be achieved using a commercial tool or through the sheet resistance and inductance per length of the technology.

Step 2. The current injected into each substrate contact by the switching circuit is characterized over a specific time window. For a large digital block, these current profiles can be obtained by precharacterizing each standard cell within each library followed by a behavioral simulation of the circuit to extract the switching time of each cell. The current injected by those cells located between two contacts is shifted to the previous contact to prevent overly optimistic results. Note that for a different window or with a different set of input vectors, these current profiles may change, affecting the merging results. The algorithm, however, can be performed multiple times for different timing windows, resulting in slightly different substrate networks. This solution is computationally possible since the algorithm performs in linear time.

Step 3. The proposed algorithm is performed based on the data obtained from the first two steps and an additional parameter V_{lim} that specifies at what voltage to merge a set of contacts. Note that V_{lim} provides flexibility to exploit accuracy versus complexity tradeoffs, as described in Section 4.

Step 4. For each voltage domain determined in Step 3, an equivalent contact is placed at the geometric mean of the merged contacts. All of the remaining ports into the substrate within that voltage domain, such as the source/drain regions of the devices, are neglected to reduce the computational complexity since the substrate contacts are the primary source injecting noise into the substrate for large scale circuits [13].

Step 5. The substrate is extracted with these equivalent contacts which are also connected to the updated ground network. The resulting netlist is analyzed to determine the substrate noise at the sense node located around the sensitive block.

The contact merging algorithm identifies the voltage domains and creates an equivalent contact for each domain. This algorithm is described in the following section.

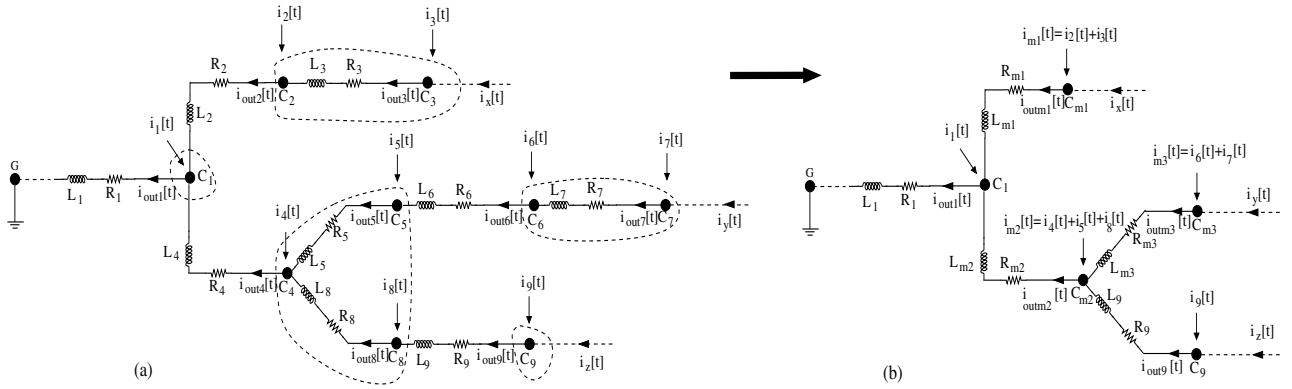

Figure 2: Digital ground network mapped to a tree where each node represents a substrate contact and the root is assumed to be the ideal ground: (a) before merging, (b) after merging. Five voltage domains are identified, as determined by C_1; C_2 and C_3; C_4, C_5, and C_8; C_6 and C_7; and C_9. The reduced network therefore has five substrate contacts. Note that all three elements of a node are updated after merging to maintain the voltage on the node with the least error.

MERGE-CONTACTS(node, V_{lim})
1. **if** node =! leaf
2. **for** x = 1:1:nc(node)
3. MERGE-CONTACTS($C_x(node)$, V_{lim})
4. **end**
5. $i_{out[t]}(node) = i_{[t]}(node) + i_{out[t]}(C_1(node)) +$ $... + i_{out[t]}(C_k(node))$
6. $V_{(diff-max)} = V_{diff}(C_1(node))$
7. bigChild = $C_1(node)$
8. **for** x=2:1:nc(node) % identify bigChild
9. **if** $V_{diff}(C_x(node)) > V_{(diff-max)}$
10. $V_{(diff-max)} = V_{diff}(C_x(node))$
11. bigChild = $C_x(node)$
12. **end**
13. **end**
14. **if** $V_{(diff-max)} < V_{lim}$ % merge the parent and children
15. $R(node) = R(node) + R(bigChild)\frac{max(i_{out[t]}(bigChild))}{max(i_{out[t]}(node))}$
16. $L(node) = L(node) + L(bigChild)\frac{max(\partial(i_{out[t]}(bigChild))/\partial t)}{max(\partial(i_{out[t]}(node))/\partial t)}$
17. $i_{[t]}(node) = i_{[t]}(node) + i_{[t]}(C_1(node)) + ... + i_{[t]}(C_k(node))$
18. Correct(node, bigChild)
19. **end**
20. **end**
21. $V_{diff}(node) = R(node)max(i_{out[t]}(node)) +$ $L(node)max(\frac{\partial(i_{out[t]}(node))}{\partial t})$

Figure 3: Pseudo-code to merge the substrate contacts on the ground network based on spatial transient voltage differences.

3.1 Contact Merging Algorithm

The extracted ground network of the aggressor circuit is mapped to a tree data structure where each substrate contact represents a node and the ideal ground is the root of the tree. Note that substrate contacts are located within the local ground network which can generally be represented as a tree. Each node in the tree is characterized by three elements: the switching current profile injected into the ground network by the switching gates $i_{[t]}(node)$, and the parasitic inductance $L(node)$ and resistance $R(node)$ between the node and the parent of the node. Note that each current profile includes the switching time information obtained at a specific time window and stored in an array at discrete time points. These switching times are obtained through a gate level behavioral simulation of the aggressor circuit.

The algorithm traverses the entire tree starting from the leaf nodes to evaluate the voltage difference between each node and parent. If this voltage difference is smaller than the user specified limit voltage V_{lim}, those nodes are merged into a single node and the node voltage is updated by modifying the resistance and inductance to maintain the absolute voltage with the least error.

An example structure is shown in Fig. 2 to illustrate the inputs and outputs of the algorithm. The algorithm operates on an extracted ground network with current profiles for each substrate contact, producing a ground network with updated impedances and a fewer number of contacts. Each equivalent contact determines a voltage domain. Note that rather than calculating the absolute voltages to decide whether the nodes should be merged, the voltage *difference* between the nodes, based on the current profile and parasitic impedances, is sufficient, significantly decreasing the complexity and memory requirements.

Pseudo-code of the proposed recursive algorithm is provided in Fig. 3. In addition to the three elements ($i_{[t]}(node)$, $R(node)$, and $L(node)$) of a node, $C_x(node)$ represents the x_{th} child of the node, $nc(node)$ represents the number of children of the node, $i_{out[t]}(node)$ represents the total switching current flowing from the node towards the parent of the node, and $V_{diff}(node)$ represents the voltage difference between the node and the parent of the node.

The algorithm calculates the voltage difference $V_{diff}(node)$ between the parent and each child. The child with the greatest voltage difference is identified as *bigChild*. If the peak value of this transient voltage difference is smaller than V_{lim}, all of the children and the parent are merged into one node and the resistance and inductance of the merged node are updated to maintain the original transient voltage. Referring to Fig. 2, after merging C_2 and C_3 into C_{m1}, R_2 and L_2 are incremented, respectively, by ΔR and ΔL to compensate for the voltage loss caused by merging such that

$$i_{out3}[t]R_3 + L_3\frac{\partial i_{out3}[t]}{\partial t} = \Delta R i_{outm1}[t] + \Delta L\frac{\partial i_{outm1}[t]}{\partial t}. \quad (2)$$

Since i_{outm1} is equal to i_{out2}, ΔR and ΔL are given by, respectively,

$$\Delta R = R_3\frac{max(|i_{out3}[t]|)}{max(|i_{out2}[t]|)}, \quad (3)$$

$$\Delta L = L_3\frac{max(|\partial i_{out3}[t]/\partial t|)}{max(|\partial i_{out2}[t]/\partial t|)}. \quad (4)$$

Note that the algorithm maintains the peak value of the absolute voltage after merging. The maximum value of the currents is there-

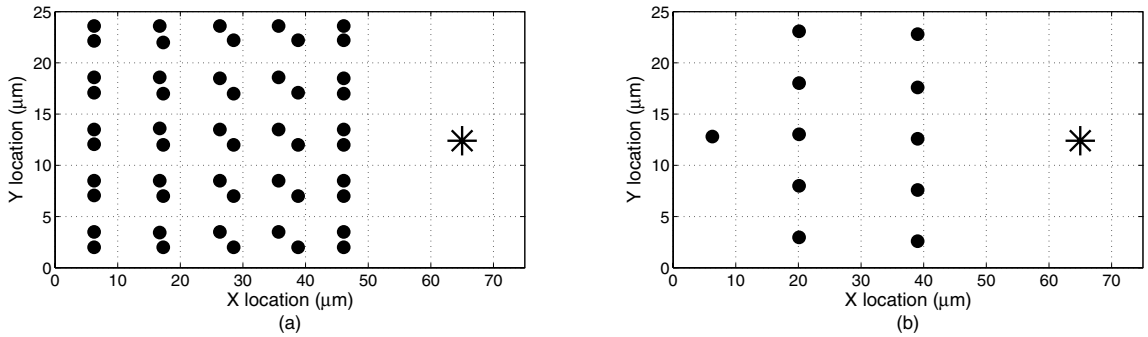

Figure 4: Physical location of the substrate contacts (represented by the circles) and the sense node (represented by the star) where the substrate noise is observed: (a) original 50 contacts before merging (b) eleven equivalent contacts after merging when $V_{lim} = 5$ mV.

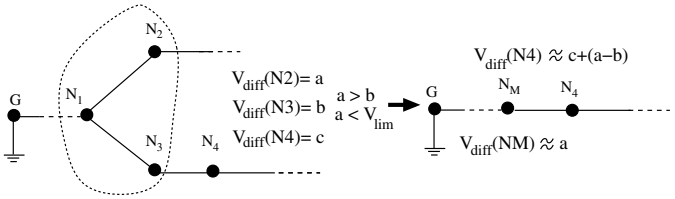

Figure 5: Illustration of the *Correct* function. After the nodes N_1, N_2, and N_3 are merged based on $V_{diff}(N2)$, $V_{diff}(N4)$ shifts by $a-b$ which is compensated by the *Correct* function.

Correct(node, bigChild)
1. **for** k = 1:1:nc(node)
2. $V_{corr}(R)$ = max[$R(bigChild)\ i_{out[t]}(bigChild)$ - $R(C_k(node))\ i_{out[t]}(C_k(node))$]
3. $V_{corr}(L)$ = max[$L(bigChild)\frac{\partial i_{out[t]}(bigChild)}{\partial t}$ - $L(C_k(node))\frac{\partial i_{out[t]}(C_k(node))}{\partial t}$]
4. **for** p = 1:1:$nc(C_k(node))$
5. $R(C_p(C_k(node))) = R(C_p(C_k(node)))$ - $\frac{V_{corr}(R)}{max[i_{out[t]}(C_p(C_k(node)))]}$
6. $L(C_p(C_k(node))) = L(C_p(C_k(node)))$ - $\frac{V_{corr}(L)}{max[\frac{\partial i_{out[t]}(C_p(C_k(node)))}{\partial t}]}$
7. **end**
8. **end**

Figure 6: Pseudo-code of the *Correct* function to prevent error accumulation after merging a set of contacts.

fore considered when merging the contacts and updating the impedance. Another option is to consider the rms value rather than the maximum value. The rms value, however, produces a larger error in the substrate noise.

Since the nodes are merged based on the voltage difference of *bigChild*, a correction is required to maintain the original transient voltage for other children to prevent error accumulation, which is achieved by the *Correct* function. An example of this process is shown in Fig. 5, illustrating the use of this function. Assuming that $V_{diff}(N2) = a$ is greater than $V_{diff}(N3) = b$ and smaller than V_{lim}, nodes $N1$, $N2$, and $N3$ are merged where *bigChild* is $N2$. After merging, the impedance of the merged node NM is adjusted to make $V_{diff}(NM)$ approximately equal to a. After merging, the new parent of $N4$ is NM, and $V_{diff}(N4)$ shifts by $a-b$. Note that the error for $N4$ accumulates with additional merging. In order to prevent this error accumulation, the impedance of $N4$ is updated by the *Correct* function to compensate the error $a-b$. Pseudo-code for the *Correct* function is shown in Fig. 6. The voltage required to compensate this error is calculated, and the resistance and inductance are correspondingly updated.

4. SIMULATION RESULTS

The algorithm has been evaluated on a digital core located close to a sensitive block in an industrial transceiver circuit, designed in a 90 nm CMOS technology on a bulk type substrate. The circuit originally contains 50 substrate contacts. The current profile for each contact is obtained as described in Step 2 for a specific time window. The parasitic resistance between each contact is determined from the sheet resistance (72 mΩ). The parasitic inductance is extracted using Q3D Extractor [14]. For the vertical ground line with a 1 μm width, the parasitic inductance is 0.79 pH/μm. For the horizontal lines with a width of 0.14 μm, the inductance is extracted as 1.14 pH/μm.

The CONTACT-MERGE algorithm, implemented in Matlab, determines the voltage domains on the substrate. Five different values (3, 5, 10, 15, and 20 mV) are used for V_{lim} to evaluate the complexity versus accuracy tradeoff. For $V_{lim} = 5$ mV, eleven voltage domains are identified. Each of these domains is represented by an equivalent substrate contact placed at the geometric mean of the merged contacts. The original physical location of the 50 substrate contacts and eleven equivalent contacts after merging when $V_{lim} = 5$ mV are illustrated in Fig. 4.

The substrate is extracted for the pre- and post-merging cases using SubstrateStorm [15]. The noise is observed using Spectre at the sense node located 20 μm from the nearest substrate contact. Note that the parasitic resistance and inductance between the substrate contacts on the ground network and the current profile of each contact are updated after merging based on the CONTACT-MERGE algorithm.

The time domain noise waveforms observed at the sense node before and after merging are compared in Fig. 7. The waveform shape and peak magnitude of the substrate noise at the sense node after merging into eleven contacts match the original noise voltage with a peak-to-peak error of 22% in the noise voltage. Note that the error increases to 160% if V_{lim} is increased to 20 mV, merging all of the contacts into a single contact. The error in the rms value of the noise over one period is 7% for eleven contacts and increases to 134% for a single contact.

Considering the number of substrate resistances, SubstrateStorm extracts 1225 resistors in the original system with 50 substrate contacts. Alternatively, when the number of contacts is reduced to eleven, the number of extracted substrate resistances is 55, corresponding to a 22X reduction. The dependence of the error and number of extracted substrate resistors on V_{lim} is shown in Fig. 8 to

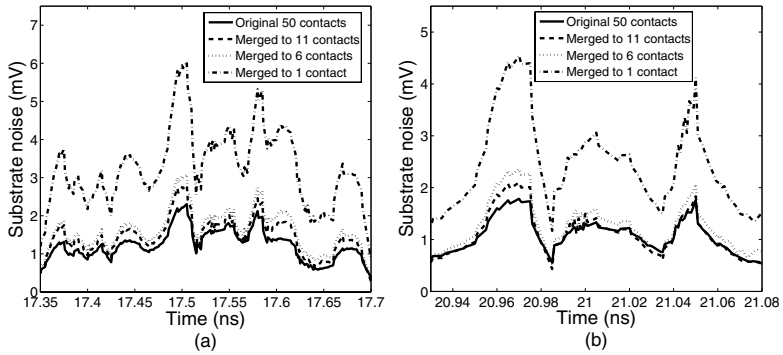

Figure 7: Comparison of the substrate noise at the sense node before and after merging for two different time intervals: (a) from 17.35 ns to 17.70 ns (b) from 20.93 ns to 21.08 ns.

Table 1: Reduction in the number of extracted substrate resistors, substrate noise at the sense node, and the corresponding error in the substrate noise for different values of V_{lim}.

	Number of substrate contacts	Number of extracted substrate resistors	Reduction	Noise at the sense node			Error at the sense node		Estimated / Original
				Peak-to-peak (mV)	RMS (mV)	At 1 GHz (dB)	Peak-to-peak (%)	RMS (%)	At 1 GHz (dB)
Original	50	1225	–	2.31	0.7	-70	–	–	–
$V_{lim} = 3$ mV	17	136	9x	2.81	0.63	-67.5	21.6	10	2.5
$V_{lim} = 5$ mV	11	55	22x	2.82	0.75	-66.5	22.1	7.1	3.5
$V_{lim} = 10$ mV	6	15	82x	3.1	0.85	-65.5	34.2	21.4	4.5
$V_{lim} = 15$ mV	3	3	408x	5.61	1.78	-59	142.8	154.3	11
$V_{lim} = 20$ mV	1	1	1225x	6	1.64	-60	159.7	134.3	10

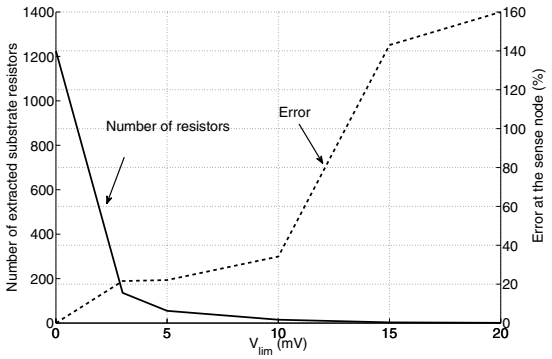

Figure 8: Number of extracted substrate resistors and the error in the peak-to-peak noise voltage at the sense node as a function of V_{lim}.

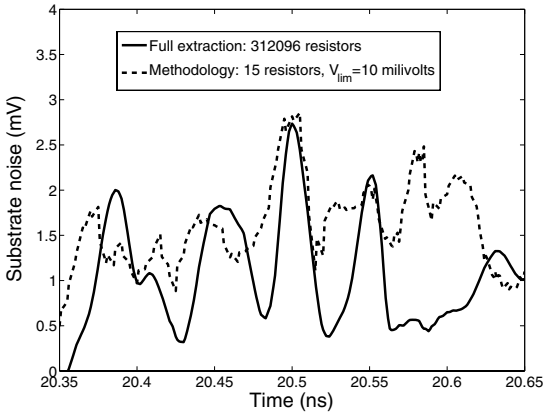

Figure 9: Comparison of the substrate noise at the sense node obtained by simulating the fully extracted circuit and application of the methodology when $V_{lim} = 10$ mV.

illustrate the complexity versus accuracy tradeoff. Note that a rapid reduction in the number of resistors is achieved with a relatively small V_{lim}. Increasing V_{lim} above 10 mV marginally improves the complexity while introducing additional error at the sense node. These results are listed in Table 1 for five different values of V_{lim}.

The complete layout of the circuit including all of the devices is extracted using Assura and SubstrateStorm to compare the results obtained by the proposed methodology with a fully extracted set of impedances. The noise waveforms at the sense node obtained by simulating the fully extracted circuit and applying the methodology when $V_{lim} = 10$ mV are compared in Fig. 9. The fully extracted circuit consists of $312,096$ resistors for the substrate and $55,856$ junction and well capacitances. The full extraction and simulation of the transient noise for this circuit on a dual core 64 bit Sun machine with Linux operating system requires approximately six hours. Alternatively, the proposed methodology reduces the number of extracted substrate resistors to 15 (for $V_{lim} = 10$ mV), achieving more than four orders of magnitude reduction, and requires negligible computational time. The peak-to-peak error of the methodology in estimating the substrate noise voltage is 20% with a highly accurate agreement in the positive peak noise, as illustrated in Fig. 9.

Note that the reduction achieved by the methodology is expected to increase for larger scale circuits due to the increasing number of substrate contacts. A common practice is to increase the density of the contacts near those aggressor blocks that can potentially affect a sensitive circuit. An aggressor digital block may therefore have a significant number of substrate contacts, where a reduction in the number of contacts quadratically reduces the number of extracted substrate resistances.

5. DISCUSSION

The proposed methodology requires a reasonable selection of V_{lim} to obtain sufficiently accurate results while reducing the overall number of extracted substrate resistors. Choosing an excessively large V_{lim} significantly increases the peak-to-peak and rms error.

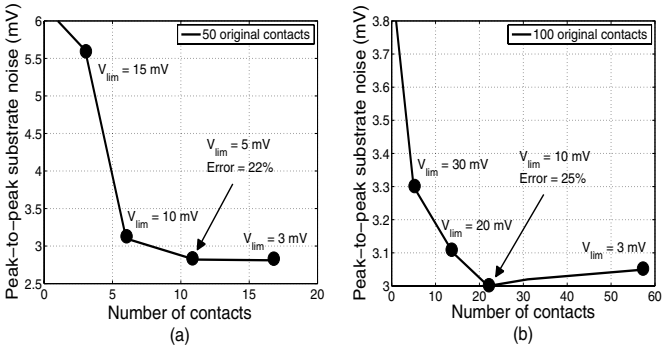

Figure 10: The variation of the peak-to-peak noise with respect to the number of contacts after merging: (a) The simulation circuit with 50 **original contacts. (b) An extension of the circuit with** 100 **original contacts.**

Alternatively, an unnecessarily small V_{lim} limits the reduction in the extracted substrate resistors, and therefore the computational efficiency.

As listed in Table 1, the peak-to-peak and rms value of the estimated peak-to-peak noise is significantly greater for the extreme case when all of the contacts are merged into one contact. As V_{lim} is reduced, the estimated noise voltages decrease, and ultimately saturate. The value where this saturation starts to occur is a good choice for V_{lim}. The corresponding peak-to-peak noise obtained at this V_{lim} is reasonably accurate with respect to the fully extracted noise voltage. The variation of the peak-to-peak noise voltage with respect to the number of contacts is shown in Fig. 10 for 50 and 100 contacts.

As illustrated in Fig. 10, the peak-to-peak noise voltage initially exhibits a rapid decrease, ultimately saturating as V_{lim} is decreased. The value of V_{lim} where the estimated peak-to-peak noise voltage saturates therefore produces sufficiently accurate results for this methodology. Note that the time complexity of the proposed algorithm is linear, allowing these iterations to be performed in a reasonable amount of time. V_{lim} can therefore be effectively selected with this iterative methodology.

6. CONCLUSIONS

A methodology is proposed to efficiently analyze substrate noise generated by an aggressor circuit in large scale circuits. The complexity of existing substrate extraction techniques is significantly reduced by exploiting similarly biased regions on the substrate of the aggressor block. An algorithm with linear time complexity is introduced to identify these voltage domains on the substrate. Each domain is represented by a single input port corresponding to an equivalent substrate contact, while the remaining ports within that domain are ignored to reduce the computational complexity. A reduction of more than four orders of magnitude in the number of extracted substrate resistances is demonstrated with an error of 20% in the peak-to-peak value of the substrate noise at the sense node.

7. REFERENCES

[1] K. Fukahori and P. R. Gray, "Computer Simulation of Integrated Circuits in the Presence of Electrothermal Interaction," *IEEE Journal of Solid-State Circuits*, Vol. 11, No. 6, pp. 834–846, December 1976.

[2] B. R. Stanisic, N. K. Verghese, R. A. Rutenbar, R. Carley, and D. J. Allstot, "Addressing Substrate Coupling in Mixed-Mode IC's: Simulation and Power Distribution Synthesis," *IEEE Journal of Solid-State Circuits*, Vol. 29, No. 3, pp. 226–238, March 1994.

[3] T. Smedes, N. P. van der Meijs, and A. J. van Genderen, "Boundary Element Methods For 3D Capacitance and Substrate Resistance Calculations in Inhomogeneous Media in a VLSI Layout Verification Package," *Advances Engineering Software*, Vol. 20, No. 1, pp. 19–27, 1994.

[4] R. Gharpurey and R. G. Meyer, "Modeling and Analysis of Substrate Coupling in Integrated Circuits," *IEEE Journal of Solid-State Circuits*, Vol. 31, No. 3, pp. 344–352, March 1996.

[5] J. P. Costa, M. Chou, and L. M. Silveria, "Efficient Techniques for Accurate Modeling and Simulation of Substrate Coupling in Mixed-Signal IC's," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, Vol. 18, No. 5, pp. 597–607, May 1999.

[6] N. K. Verghese, D. J. Allstot, and M. A. Wolfe, "Verification Techniques for Substrate Coupling and their Application to Mixed-Signal IC Design," *IEEE Journal of Solid-State Circuits*, Vol. 31, No. 3, pp. 354–365, March 1996.

[7] N. K. Verghese, D. J. Allstot, and S. Masui, "Rapid Simulation of Substrate Coupling Effects in Mixed-Mode ICs," *Proceedings of the IEEE Custom Integrated Circuits Conference*, pp. 18.3.1–18.3.4, May 1993.

[8] E. Charbon, R. Gharpurey, R. G. Meyer, and A. S. Vincentelli, "Substrate Optimization Based on Semi-Analytical Techniques," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, Vol. 18, No. 2, pp. 172–190, February 1999.

[9] A. M. Niknejad, R. Gharpurey, and R. G. Meyer, "Numerically Stable Green Function for Modeling and Analysis of Substrate Coupling in Integrated Circuits," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, Vol. 17, No. 4, pp. 305–315, April 1998.

[10] E. Schrik and N. P. van der Meijs, "Combined BEM/FEM Substrate Resistance Modeling," *Proceedings of the IEEE/ACM Design Automation Conference*, pp. 771–776, June 2002.

[11] D. Ozis, T. Fiez, and K. Mayaram, "A Comprehensive Geometry-dependent Macromodel for Substrate Noise Coupling in Heavily Doped CMOS Processes," *Proceedings of the IEEE Custom Integrated Circuits Conference*, pp. 497–500, September 2002.

[12] H. Lan *et al.*, "Synthesized Compact Models and Experimental Verifications for Substrate Noise Coupling in Mixed-Signal ICs," *IEEE Journal of Solid-State Circuits*, Vol. 41, No. 8, pp. 1817–1829, August 2006.

[13] M. Badaroglu *et al.*, "Evolution of Substrate Noise Generation Mechanisms With CMOS Technology Scaling," *IEEE Transactions on Circuits and Systems I: Regular Papers*, Vol. 53, No. 2, pp. 296–305, February 2006.

[14] "Q3D Extractor^{TM} tool," [Online]. Available: http://www.ansoft.com.

[15] "Assura RCX^{TM}, SubstrateStorm^{TM}, Spectre^{TM} tools," [Online]. Available: http://www.cadence.com.