[3] ——, *MicroC/OS-II*. Lawrence, KS: R & D Books, 1998.
[4] P. A. Laplante, *Real-Time Systems Design and Analysis: An Engineers Handbook*. Piscataway, NJ: IEEE Press, 1993.
[5] R. Sharma, "Distributed application development with inferno," in *Proc. Design Automation Conf.*, June 1999, pp. 146–150.
[6] D. Stepner, N. Rajan, and D. Hui, "Embedded application design using a real-time OS," in *Proc. Design Automation Conf.*, June 1999, pp. 151–156.
[7] W. Warner, "Non-pre-emptive multithreading performs embedded software's juggling act," *Electron. Design News*, vol. 44, pp. 117–126, July 8, 1999.
[8] L. Benini and G. De Micheli, *Dynamic Power Management: Design Techniques and CAD Tools*. Norwell, MA: Kluwer, 1997.
[9] A. R. Chandrakasan and R. W. Brodersen, *Low Power Digital CMOS Design*. Norwell, MA: Kluwer, 1995.
[10] G. Yeap, *Practical Low Power Digital VLSI Design*. Norwell, MA: Kluwer, 1998.
[11] J. Monteiro and S. Devadas, *Computer-Aided Design Techniques for Low Power Sequential Logic Circuits*. Norwell, MA: Kluwer, 1996.
[12] J. Rabaey and M. P., Eds., *Low Power Design Methodologies*. Norwell, MA: Kluwer, 1996.
[13] V. Tiwari, S. Malik, and A. Wolfe, "Power analysis of embedded software: A first step toward software power minimization," *IEEE Trans. VLSI Syst.*, vol. 2, pp. 437–445, Dec. 1994.
[14] T. Sato, Y. Ootaguro, M. Nagamatsu, and H. Tago, "Evaluation of architecture-level power estimation for CMOS RISC processors," in *Proc. Symp. Low Power Electron.*, Oct. 1995, pp. 44–45.
[15] C. T. Hsieh, M. Pedram, G. Mehta, and F. Rastgar, "Profile-driven program synthesis for evaluation of system power dissipation," in *Proc. Design Automation Conf.*, June 1997, pp. 576–581.
[16] L. Benini and G. De Micheli, "System-level power optimization: Techniques and tools," in *Proc. Int. Symp. Low Power Electron. Design*, Aug. 1999, pp. 288–293.
[17] B. Dave, G. Lakshminarayana, and N. K. Jha, "COSYN: Hardware-software co-synthesis of embedded systems," in *Proc. Design Automation Conf.*, June 1997, pp. 703–708.
[18] Y. Li and J. Henkel, "A framework for estimating and minimizing energy dissipation of embedded HW/SW systems," in *Proc. Design Automation Conf.*, June 1998, pp. 188–193.
[19] T. Simunic, L. Benini, and G. De Micheli, "Cycle-accurate simulation of energy consumption in embedded systems," in *Proc. Design Automation Conf.*, June 1999, pp. 867–872.
[20] M. Lajolo, A. Raghunathan, S. Dey, L. Lavagno, and A. Sangiovanni-Vincentelli, "Efficient power estimation techniques for HW/SW systems," in *Proc. Alessandro Volta Memorial Wkshp. on Low Power Design*, Mar. 1999, pp. 191–199.
[21] S. Gurumurthi, A. Sivasubramaniam, M. J. Irwin, N. Vijaykrishnan, M. Kandemir, T. Li, and L. K. John, "Using complete machine simulation for software power estimation: The softwatt approach," in *Proc. Int. Symp. High-Performance Comput. Architecture*, Feb. 2002, pp. 141–150.
[22] V. Tiwari, S. Malik, and A. Wolfe, "Compilation techniques for low energy: An overview," in *Proc. Symp. Low Power Electronics*, Oct. 1994, pp. 38–39.
[23] T. Simunic, G. De Micheli, and L. Benini, "Energy-efficient design of battery-powered embedded systems," in *Proc. Int. Symp. Low Power Electron. Design*, Aug. 1999, pp. 212–217.
[24] J. L. da Silva, F. Catthoor, D. Verkest, and H. De Man, "Power exploration for dynamic data types through virtual memory management refinement," in *Proc. Int. Symp. Low Power Electron. Design*, Aug. 1998, pp. 311–316.
[25] Q. Qiu, Q. Wu, and M. Pedram, "Stochastic modeling of a power-managed system: Construction and optimization," in *Proc. Int. Symp. Low Power Electron. Design*, Aug. 1999, pp. 194–199.
[26] L. Benini, A. Bogliolo, S. Cavallucci, and B. Ricco, "Monitoring system activity for OS-directed dynamic power management," in *Proc. Int. Symp. Low Power Electron. Design*, Aug. 1998, pp. 185–190.
[27] I. Hong, D. Kirovski, G. Qu, M. Potkonjak, and M. B. Srivastava, "Power optimization of variable voltage core-based systems," in *Proc. Design Automation Conf.*, June 1998, pp. 176–181.
[28] T. Ishihara and H. Yasuura, "Voltage scheduling problem for dynamically variable voltage processors," in *Proc. Int. Symp. Low Power Electron. Design*, Aug. 1998, pp. 197–202.
[29] T. Pering, T. Burd, and R. Brodersen, "The simulation and evaluation of dynamic voltage scaling algorithms," in *Proc. Int. Symp. Low Power Electron. Design*, Aug. 1998, pp. 76–81.
[30] N. K. Jha, "Low power system scheduling and synthesis," in *Proc. Int. Conf. Computer-Aided Design*, Nov. 2001, pp. 259–263.
[31] *Proc. Workshop Compilers Oper. Syst. Low Power*, L. Benini, M. Kemir, and J. Ramanujam, Eds., Sept. 2002.
[32] A. Vahdat, A. R. Lebeck, and C. S. Ellis, "Every Joule is precious: A case for revisiting operating system design for energy efficiency," in *Proc. ACM SIGOPS Eur. Workshop*, Sept. 2000, pp. 31–36.
[33] R. P. Dick, G. Lakshminarayana, A. Raghunathan, and N. K. Jha, "Power analysis of embedded operating systems," in *Proc. Design Automation Conf.*, June 2000, pp. 312–315.
[34] T. K. Tan, A. Raghunathan, and N. K. Jha, "EMSIM: An energy simulation framework for an embedded operating system," in *Proc. Int. Symp. Circuits Syst.*, May 2002, pp. 464–467.
[35] K. Baynes, C. Collins, E. Fiterman, B. Ganesh, P. Kohout, C. Smit, T. Zhang, and B. Jacob, "The performance and energy consumption of three embedded real-time operating systems," in *Proc. Int. Conf. Compilers Architecture Synthesis Embedded Syst.*, Nov. 2001, pp. 203–210.
[36] *N2C Training Manual*, CoWare, San Jose, CA, 1999.
[37] *MB8683x User's Guide*, Fujitsu Microelectronics, Inc., Tokyo, Japan.
[38] *SPARClite Series 32-Bit RISC Embedded Processor MB86832 Databook*, Fujitsu Microelectronics, Inc., Tokyo, Japan, 1998.
[39] *1995 DRAM Databook*, IBM, White Plains, NY, 1994.

Accurate Crosstalk Noise Modeling for Early Signal Integrity Analysis

Li Ding, David Blaauw, and Pinaki Mazumder

***Abstract*—In this paper, we propose an accurate and fast method to estimate the crosstalk noise in the presence of multiple aggressor nets for use in physical design automation tools. Since noise estimation is often part of the inner loop of optimization algorithms, very efficient closed-form solutions are needed. Previous approaches model aggressor nets one at a time, assuming that the coupling capacitance to all quiet aggressor nets are grounded. They also model the load from interconnect branches as a lumped capacitor, the value of which is the sum of interconnect and load capacitances of the branch. Finally, previous works typically use simple lumped 2–4-node circuit templates and employ a so-called dominant pole approximation to solve the template circuit. While these approximations allow for very fast analysis, they may result in significant underestimation of the noise. In this paper, we propose a new and more comprehensive fast noise estimation method. We propose a novel reduction technique for modeling quiet aggressor nets based on the concept of coupling point admittance. We also propose a reduction method to replace tree branches with effective capacitors which models the effect of resistive shielding. Furthermore, we model the simplified single aggressor net crosstalk noise problem using a 6-node template circuit and propose a new double pole approach to solve the template circuit. We have tested the proposed method on noise-prone interconnects from an industrial high-performance processor. Our results show a worst case error of 7.8% and an average error of 2.7%, while allowing for very fast analysis.**

***Index Terms*—Crosstalk noise, digital CMOS circuits, interconnect, noise estimation, signal integrity.**

Manuscript received August 15, 2002; revised November 18, 2002. This work was supported in part by the Office of Naval Research, by the National Science Foundation, and in part by the Semiconductor Research Corporation under Contract 2001-HJ-959. This paper was recommended by Associate Editor S. S. Sapatnekar.

The authors are with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109 USA (e-mail: lding@eecs.umich.edu; blaauw@eecs.umich.edu; mazum@eecs.umich.edu).

Digital Object Identifier 10.1109/TCAD.2003.810741

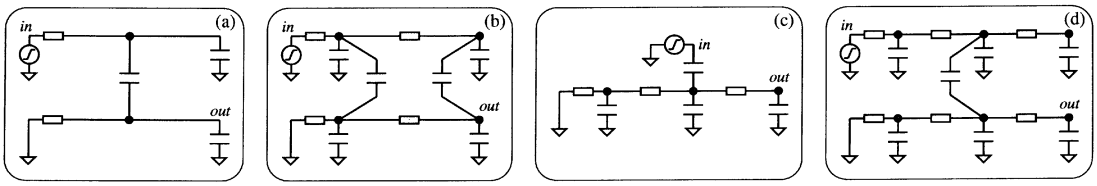

Fig. 1. Circuit templates for crosstalk noise estimation. (a) 2-node symmetric template [14]. (b) 4-node symmetric template [15]. (c) 3-node nonsymmetric template [16]. (d) 6-node symmetric template (this work).

I. Introduction

Crosstalk noise between signal wires has become a major source of failures in modern high-performance very large scale integrated (VLSI) systems [1]–[7]. This is especially the case in deep submicron circuit designs due to the aggressive interconnect scaling in the lateral dimensions with relatively unchanged vertical dimensions. The coupling capacitance among adjacent wires can be significantly larger than wire ground capacitance. In such strongly coupled systems, the state of a wire strongly depends on the states of its neighboring wires. The switching of a first net, referred to as the aggressor net, may affect the state of a second nearby net, referred to as the victim net.

The coupling among adjacent wires has made it necessary to analyze a victim net together with all its coupled aggressor nets. However, efficient and accurate analysis of the coupling noise is difficult since: 1) the number of coupling nets is typically large, ranging from tens to hundreds of nets and 2) the aggressor and victim nets may have a large number of branches. Due to the size of the coupled interconnect, it is impractical to use SPICE simulation methods for large VLSI chips and a number of approaches have been proposed in the literature to improve the efficiency of crosstalk noise analysis. These crosstalk noise modeling approaches can be loosely classified into two categories based on their tradeoff between accuracy and efficiency. The first class of methods aims to achieve maximum modeling accuracy while gaining substantial speedup over standard SPICE simulation. A number of methods use linear model order reduction techniques [8]–[10] to reduce the original large but sparse matrix to a much smaller and denser matrix which is then used as a macromodel for crosstalk simulation. These approaches, such as PRIMA [10], can achieve orders of magnitude speedup over SPICE simulation while achieving very high accuracy. These methods are useful for post-layout verification where high accuracy is a key requirement and they enable the analysis of large industrial designs in a matter of hours [11]–[13].

However, in deep submicron VLSI chip designs, there is often the need to assess and avoid crosstalk noise in the early stages of the chip design flow. Therefore, the second class of crosstalk noise modeling methods aims to further improve the efficiency of noise analysis, such that they can be used in the inner loop of physical design automation tools. These methods [14]–[16] reduce the interconnect topology into a very simple template circuit with a known and fixed topology. The simple template circuit is then analytically modeled. Four different template circuits are shown in Fig. 1. The simplicity of those second class methods leads to an analysis efficiency that is another several orders of magnitude faster than model order reduction-based methods. The approaches proposed in this paper address this second class of crosstalk noise modeling methods.

Since the number of aggressor nets to a victim net is potentially large, it is very difficult to properly align the switching times in order to generate the worst case noise [17], [18]. A common approach uses the superposition law to estimate crosstalk noises. In this approach, the complete, coupled network is simulated once for each aggressor driver, while all other aggressor drivers are held quiet. To compute the worst case crosstalk noise of an N-aggressor system, it is, therefore, necessary to calculate the crosstalk noise of an $N+1$-net system N times. When the number of aggressors is large, the $N+1$-net system can be very large, and all previous approaches have, therefore, assumed that either explicitly or implicitly, that coupling capacitances from the victim net to nonswitching or *quiet* aggressor nets are grounded during superposition. This reduces the network from $N+1$ nets to only two nets; hence, the maximum crosstalk noise can be calculated in linear time with respect to the number of aggressors N. However, during superposition, the quiet aggressor nets follow the victim net waveform to a certain extent and their effective load capacitance is always less than the coupling capacitance value. Note that crosstalk noise decreases as the total victim load capacitance increases. By using grounded coupling capacitance, these methods have, therefore, improved the efficiency of the analysis while potentially underestimating the crosstalk noise.

Similarly, the techniques used in the literature for fast crosstalk noise estimation [14]–[16] do not consider the effect of resistive shielding of long interconnects. They typically lump the total wire and load capacitances of a branch at the branching point to simplify the circuit. However, the effect of resistive shielding is becoming more prominent as the process technology scales down due to increasing interconnect resistances. The resistive shielding effect reduces the capacitance that the circuit observes at the branching point. Again, this approximation will result in an underestimation of crosstalk noise. Finally, previous methods use a 2–4-node template circuit which is solved using a so-called *dominant pole* approximation. We will show that the inability of the template circuit to model the waveform characteristic of the switching aggressor and the dominant pole approximation further compromise the accuracy of the existing fast noise analysis methods.

In this paper, we present an efficient crosstalk noise estimation framework which maintains the efficiency of past approaches, but significantly improves on their accuracy. We propose a novel quiet aggressor net and tree branch reduction technique which models quiet aggressor nets and tree branches with effective load capacitances. Formulas are derived to calculate the values of these effective capacitances using coupling-point and branching-point admittance together with approximate waveforms at the coupling and branching points. In order to better model the waveform characteristic of the switching aggressor net, we use a 6-node template circuit, which significantly enhances the accuracy of the noise estimation. To solve this more complex template circuit, we propose a new double pole method and confirm its accuracy compared with SPICE simulation. We have applied the proposed methodology on industrial nets that were obtained from a high-performance microprocessor in 0.15-μm technology. The results demonstrate that the proposed methods significantly enhance the accuracy of the noise estimation and eliminate the tendency of prior methods to underestimate the noise level. At the same time, the proposed method maintains the efficiency of previous methods and is linear in runtime with respect to the total number of elements in the circuit.

The rest of the paper is organized as follows. Section II explains the overall framework of the proposed noise estimation methodology. In Section III, we introduce quiet aggressor net reduction and tree branch

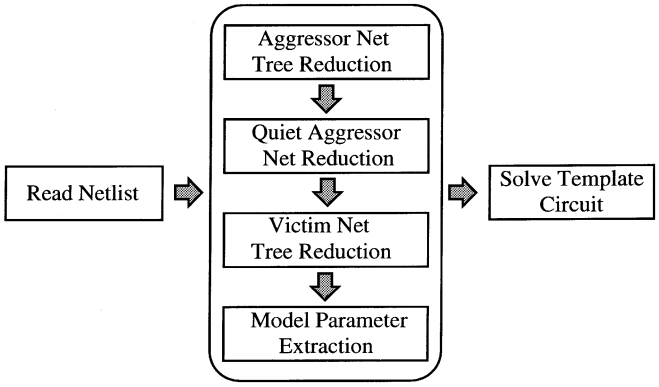

Fig. 2. Flowchart of the proposed crosstalk noise estimation methodology.

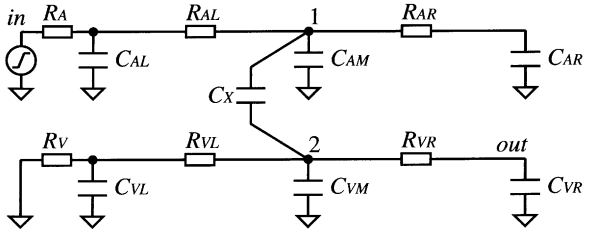

Fig. 3. Proposed six-node template circuit for single aggressor net crosstalk noise model.

reduction techniques based on point admittance matching. The reduced circuit is then analyzed in Section IV, where we proposed the double pole model for efficient yet accurate noise calculation. And in Section V, we present results of proposed methodology on industrial circuits.

II. Methodology

The basic idea of the proposed method is first to reduce a large crosstalk network into a simple template circuit. The template circuit is then solved analytically. The flowchart of the reduction scheme is illustrated in Fig. 2. First, we apply the tree reduction operation on each aggressor net. Second, we apply quiet aggressor net reduction operation on each of the $N-1$ nonswitching aggressors. Third, the branches in the victim net are reduced in a similar manner as those aggressor net branches. At the end of this step, we obtain a simple circuit with only two main wires each corresponding to the victim net and the active aggressor net. Finally, resistance and capacitance values of the reduced template circuit, shown in Fig. 3 are extracted.

The template circuit for crosstalk noise modeling shown in Fig. 3 is an extension to the 2-π model proposed in [16], where the victim net is modeled using the 2-π (3-node) circuit while the aggressor net is simplified as a saturated ramp input at node 1 in Fig. 3. In this paper, we model both victim net and aggressor net as 2-π circuits so that the location of the capacitive coupling can be correctly modeled and overall modeling accuracy is much improved. We have proposed a simple yet accurate double pole model to solve the crosstalk noise estimation problem in the reduced template circuit. Note that this template circuit, however, is only suitable for short to medium interconnects because it uses only one lumped coupling capacitor. More complex template circuits with larger numbers of coupling capacitors should be employed for very long wires. Nevertheless, we will show in Section V that the proposed simple 6-node template circuit works quite well for interconnects up to 3 mm in a 0.15-μm process technology. Furthermore, the reduction methods proposed in this paper are generic for any tree-type circuit topology.

It is quite straightforward to calculate the circuit parameters of the proposed model. Taking the victim net as an example, node 2 is the center of the coupling region, which divides the entire victim wire into two segments. R_{VL} and R_{VR} are the lumped resistances of the left and the right segments, respectively. C_{VL} is half the wire capacitance of the left segment $C_{\rm left}$ plus the driver output diffusion capacitance. C_{VR} is the sum of the load capacitance and half the value of the wire capacitance of the right segment $C_{\rm right}$. Finally, C_{VM} is $(C_{\rm left}+C_{\rm right})/2$, which is half the total wire capacitance. The aggressor net can be treated the same way.

Finally, as an example, we illustrate the steps involved during the reduction for an simple circuit in Fig. 4. In this example, we want to evaluate the effect of *Agg1* on receiver *r1* of the victim. In the first step, two branches of *Agg1* are reduced to effective capacitors. Since *Agg2* has only one fanout, no branch reduction is required. In the second step, we reduce the quiet aggressor, i.e., *Agg2* to an effective capacitor C_3. In the third step, we perform victim net tree reduction where the branch goes downward is reduced. Finally, the resistance and capacitance values of the reduced circuit can be extracted from the circuit shown in Fig. 4(d).

III. Reduction Techniques

Each reduction technique described in this section consists of two phases in sequence. In the first phase, a quiet aggressor net or tree branch is modeled using simple reduced circuits by matching the lower order Taylor series expansion coefficients of the admittance $Y(s)$ at the coupling point or branching point of the circuit. In the second phase, an effective capacitance is derived to replace those reduced circuits to further improve the efficiency.

A. Overview of Point Admittance

Let $Y(s)$ denote the point admittance of a general circuit. We have the following equation based on the Taylor series expansion theorem

$$Y(s)=\sum_{n=0}^{\infty} y_n s^n \tag{1}$$

where y_n is the nth expansion coefficient. For many circuit applications, it is found that the terms up to s^3 are adequate to characterize the transient response of a linear circuit

$$Y(s)=y_0+y_1 s+y_2 s^2+y_3 s^3+O(s^4). \tag{2}$$

Note that the first term y_0 is zero when there is no dc conducting path from the observing point to the ground.

The *coupling-point admittance* or *branching-point admittance* is computed starting from the leaf nodes of a *RC* tree then going back to the coupling or branching point. This is similar to the approaches used in solving the driving-point admittance problem for gate delay calculation [19]. Three basic rules are used in the algorithm to calculate the lower order coefficients. Those rules are presented in (3)–(5) and are illustrated in Fig. 5. Proofs of those rules are straightforward.

Rule 1: Serial resistance

$$\begin{aligned} y_0^* &= p y_0 \\ y_1^* &= p^2 y_1 \\ y_2^* &= p^2 y_2 - p^3 r y_1^2 \\ y_3^* &= p^2 y_3 - 2p^3 r y_1 y_2 + p^4 r^2 y_1^3 \end{aligned} \tag{3}$$

where the parameter p is defined as $p=1/(1+r y_0)$.

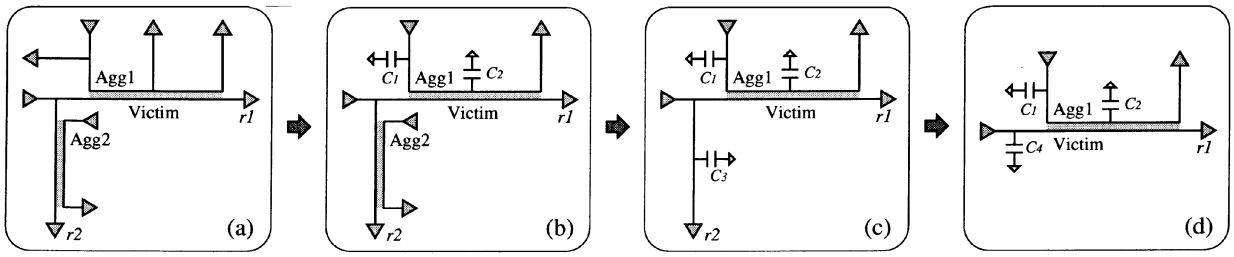

Fig. 4. Example using the tree reduction and quiet aggressor net reduction techniques.

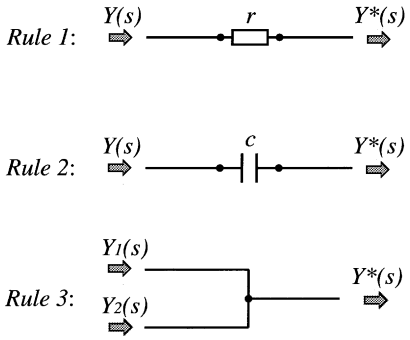

Fig. 5. Rules for calculating point admittance expansion coefficients.

Rule 2: Serial capacitance

$$\begin{aligned} y_0^* &= 0 \\ y_1^* &= c \\ y_2^* &= \frac{-c^2}{y_0} \\ y_3^* &= \frac{c^2(y_1 + c)}{y_0^2}. \end{aligned} \tag{4}$$

Rule 3: Branch join

$$\begin{aligned} y_0^* &= y_{1,0} + y_{2,0} \\ y_1^* &= y_{1,1} + y_{2,1} \\ y_2^* &= y_{1,2} + y_{2,2} \\ y_3^* &= y_{1,3} + y_{2,3} \end{aligned} \tag{5}$$

where $y_{i,0}$, $y_{i,1}$, $y_{i,2}$, and $y_{i,3}$ are the first four Taylor series expansion coefficients of the ith branch ($i = 1, 2$), respectively. Note that *Rule 3* can be applied multiple times when there are more than two joining branches.

The first four terms of the admittance are always exact when these three rules are applied because no higher than y_3 terms appears at the right-hand side of (3)–(5). The time complexity to reduce a subtree using this reduction technique is linear with respect to the number of *RC* elements in the netlist.

B. Quiet Aggressor Net Reduction

Each aggressor net is first modeled using a 2-π circuit. In this subsection, we describe a novel technique that reduces the quiet aggressor net to an efficient capacitance. Consider the 2-π model of a quiet aggressor net shown in Fig. 6(a). We first reduce the aggressor net to a single resistor R_A^* and a single capacitor C_A^* as shown in Fig. 6(b) by matching the first two Taylor series expansion coefficients y_0 and y_1 of the aggressor net at node *A*. Since only y_0 and y_1 appear at the right-hand side of (4), we do not need higher order terms at node *A* to

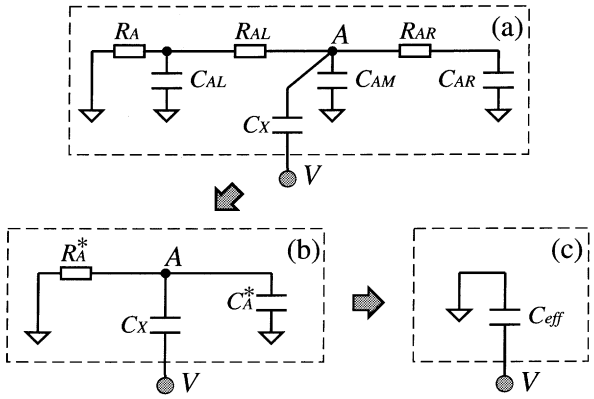

Fig. 6. Quiet aggressor net reduction for crosstalk estimation. (a) Equivalent circuit for the quiet aggressor net. (b) Reduced circuit by matching first two Taylor expansion terms. (c) Effective capacitance for the aggressor.

calculate accurate terms up to y_3^* at node *V*. By applying both *Rule 1* and *Rule 3*, the admittance at node *A* is obtained as

$$Y_A(s) = \frac{1}{R_A + R_{AL}} + \left(\frac{R_A^2}{(R_A + R_{AL})^2} C_{AL} + C_{AM} + C_{AR} \right) s + O(s^2). \tag{6}$$

Therefore, the devices in the simplified circuit shown in Fig. 6(b) have the following values

$$R_A^* = R_A + R_{AL} \tag{7}$$

$$C_A^* = \frac{R_A^2}{(R_A + R_{AL})^2} C_{AL} + C_{AM} + C_{AR}. \tag{8}$$

Next, we derive the formula to estimate the effective coupling capacitance C_{eff} of a quiet aggressor net, based on the simplified circuit shown in Fig. 6(b). To give some intuition, we first study the upper and lower bounds of C_{eff}. When the effective resistance R_A^* of the quiet aggressor net approaches zero, we can consider node *A* as being grounded. Therefore, the effective coupling capacitance is the actual coupling capacitance $C_{\text{eff}}^{\max} = C_X$. On the other hand, as R_A^* approaches infinity, node *A* floats and the coupling point *V* is connected to the ground through two series capacitors C_X and C_A^*. Therefore, the effective coupling capacitance approaches $C_{\text{eff}}^{\min} = C_A^* C_X / (C_A^* + C_X)$. For a realistic R_A^* value, the C_{eff} is somewhere between these two bounds.

We find the value of the effective capacitance by matching the current drawn by the circuit in Fig. 6(b) with that taken by the effective capacitor which is grounded at the other end. The current drawn from node *V* to node *A* through the coupling capacitor C_X reads

$$I = C_X \left(\frac{dV_V(t)}{dt} - \frac{dV_A(t)}{dt} \right). \tag{9}$$

Therefore, our task is to find a constant C_{eff} such that

$$C_{\text{eff}}\frac{dV_V(t)}{dt} \simeq C_X\left(\frac{dV_V(t)}{dt} - \frac{dV_A(t)}{dt}\right). \tag{10}$$

The effective capacitance value during the entire input rising period can be estimated by integrating both sides of (10). Since the initial value for both victim and aggressor nets are zero, we obtain the following approximate equation

$$C_{\text{eff}}V_V(t_r) = C_X\Big(V_V(t_r) - V_A(t_r)\Big). \tag{11}$$

Assume the voltage waveform of the victim net is a normalized ramp input $V_V(t) = t/t_r$, $0 \le t \le t_r$. At time t_r, we have $V_V(t_r) = 1$ for the ramp input, therefore, (11) is simplified to

$$C_{\text{eff}} = C_X\Big(1 - V_A(t_r)\Big). \tag{12}$$

Under the ramp input approximation, Kirchoff's current equation at the aggressor node *A* is

$$(C_A^* + C_X)\frac{dV_A(t)}{dt} + \frac{V_A(t)}{R_A^*} = \frac{C_X}{t_r} \tag{13}$$

during the rising period of the signal. Solving the differential equation with the initial condition that $V_A(0) = 0$, we obtain the following equation

$$V_A(t) = \frac{R_A^* C_X}{t_r}\Big(1 - e^{-t/(R_A^*(C_A^*+C_X))}\Big). \tag{14}$$

Inserting (14) at $t = t_r$ into the right-hand side of (12), we obtain the following formula for the effective capacitance

$$C_{\text{eff}} = \left(1 - \frac{R_A^* C_X}{t_r}\Big(1 - e^{-t_r/(R_A^*(C_A^*+C_X))}\Big)\right)C_X. \tag{15}$$

It can be easily verified that the effective coupling capacitance C_{eff} obtained approaches C_X when $R_A^* \to 0$ and that C_{eff} approaches $C_A^* C_X/(C_A^* + C_X)$ as $R_A^* \to \infty$. Experiments on a large number of random circuits have shown that using the proposed effective capacitance results in less than 5% error in most cases while using either $C_{\text{eff}}^{\max}$ or $C_{\text{eff}}^{\min}$ as the effective capacitance may have over 20% error.

C. Tree Branch Reduction

In general, a net has a tree structure instead of being a simple wire. Previous works use a simple method for tree branch reduction, where the total capacitances including wire capacitances and load capacitances of a branch are lumped at the branching point. However, with scaling of VLSI technology, the effect of interconnect resistive shielding can no longer be neglected. When interconnect resistance of a branch is considered, the actual capacitance seen at the branching point is always less than the total capacitance of the branch. Therefore, using total capacitance will result in an underestimation of the crosstalk noise. In this section, we derive a formula to find the value of the effective branching capacitance.

The problem we have here is very similar to the driving point admittance problem for gate delay calculation. However, the trees we consider here are actually branches that are connected to the "main" wires of the aggressor nets or the victim net. We model those branches employing similar approaches as those used in [19] and [20]. First, a general *RC* tree structure is reduced to a simple π-model as shown in Fig. 7(b) by matching the first three moments of the tree. The resulting model is then further reduced to an effective capacitance, shown in Fig. 7(c), for a given signal switching slope at the node *P*.

The difference between the proposed method and the techniques for effective driving point capacitance lies in the interfacing of the π-type

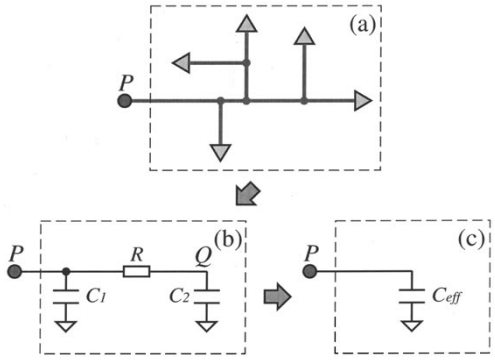

Fig. 7. Tree reduction for crosstalk estimation. (a) General *RC* tree branch. (b) Reduced-order $\boldsymbol{\pi}$-model for the tree. (c) Tree effective capacitance.

circuit with external waveforms. For delay calculation, the effective capacitance tries to match the average capacitance for the period from start to the time when the voltage reaches 50% of the supply voltage. For crosstalk noise estimation, however, we try to match the average capacitance of the branch during the entire signal switching period.

Since there is no direct dc path to the ground in circuit branches, we always have $y_0 = 0$. Once the first three moments y_1, y_2, and y_3 of a general *RC* tree are obtained by repeatedly applying *Rule 1*, we can construct a reduced π-type circuit which matches those three moments. The values of the capacitors and the resistor in the figure are calculated as

$$C_1 = y_1 - \frac{y_2^2}{y_3} \qquad C_2 = \frac{y_2^2}{y_3} \qquad R = -\frac{y_3^2}{y_2^3}. \tag{16}$$

Note that for a *RC* tree, y_2 is always a negative value. Therefore, the resistance R is positive. Similarly, C_1 and C_2 are always positive for realistic circuits and the sum of C_1 and C_2 is the total capacitance of the *RC* tree.

The π-type circuit derived is still not simple enough for our purpose because each branch on the two "main" wires will add one additional node (node *Q*) to the final reduced circuit and the number of such branches can potentially be very large. So the next step is to find an effective capacitance for a given input waveform at node *P* such that this single effective capacitor can approximate the load condition of the π circuit.

Similar to the approach used in the previous subsection, we try to find a constant C_{eff} such that

$$C_{\text{eff}}\frac{dV_A(t)}{dt} \simeq C_1\frac{dV_A(t)}{dt} + C_2\frac{dV_B(t)}{dt} \tag{17}$$

which yields the following approximated equation after integrating from 0 to t_r

$$C_{\text{eff}}V_A(t_r) = C_1V_A(t_r) + C_2V_B(t_r). \tag{18}$$

Referring to Fig. 7(b), the nodal equation at node *Q* is written as

$$C_2\frac{dV_B(t)}{dt} = \frac{V_A(t) - V_B(t)}{R}. \tag{19}$$

Assuming a normalized saturated ramp input at the victim node with a rise time of t_r, (19) can be rewritten as

$$RC_2\frac{dV_B(t)}{dt} + V_B(t) = \frac{t}{t_r} \tag{20}$$

which can be easily solved

$$V_B(t) = \frac{t}{t_r} - \frac{RC_2}{t_r}(1 - e^{-t/RC_2}). \tag{21}$$

The effective capacitance can be derived by combining (18) and (21) noting that $V_A(t_r) = 1$ under the ramp input assumption

$$C_{\text{eff}} = C_1 + \left(1 - \frac{RC_2}{t_r}(1 - e^{-t_r/RC_2})\right) C_2. \tag{22}$$

IV. Template Circuit Analysis

Using tree reduction and quiet aggressor net reduction techniques described in previous sections, the crosstalk noise estimation problem is transformed to a much simpler one of solving the 6-node reduced circuit as shown in Fig. 3. In this section, we derive the analytical expression to calculate the crosstalk noise waveform of the reduced circuit.

For template circuits with fixed topology, we can always find all poles and their respective residues. Exact waveform, therefore, can be derived based on those pole/residue pairs. This approach, however, requires to solve a sixth order equation with respect to s in order to obtain the six poles. This has to be done numerically, hence, will have negative impact on the efficiency of the entire crosstalk estimation algorithm. Furthermore, this approach does not give a clear physical picture on the waveform of the noise glitch, nor does it offer a simple way to calculate the maximum noise height and the noise width. For example, to calculate the maximum crosstalk noise voltage, we have to solve the equation $dV_{\text{out}}(t)/dt = 0$, which is a transcendental equation, therefore, has to be solved numerically.

The desire to have simple analytical expressions for the noise voltage waveform and the maximum noise voltage has led to models based on the dominant pole approximation, which have been demonstrated to be fairly accurate for certain template circuits [15], [16]. However, for our template circuit shown in Fig. 3, which is more complex than the template circuits used in previous works, the dominant pole approximation is no longer a good approximation: 1) it predicts the noise voltage peaks at time t_r, which is incorrect; 2) the derivative of the noise voltage expression is not continuous at time t_r; and 3) the calculated maximum noise voltage is not very accurate.

It is the goal of this work to derive a simple yet efficient formula that overcomes the shortcomings of the aforementioned problems. It is clear that the single pole approximation is not adequate for the proposed 6-node template circuit. We will, therefore, use a double pole approximation approach where the first pole mainly models the victim net and the second pole models the aggressor net.

First, we estimate the voltage waveform at the coupling node on the active aggressor net by temporarily treating the victim net as a "quiet aggressor net" and reducing it to an effective capacitance. The resulting circuit has only three nodes, therefore, can be relatively accurately modeled using one pole. The Elmore delay from the input to the coupling node at the aggressor net is estimated as

$$t_{A,0} = R_A C_{AL} + (R_A + R_{AL})(C_{AM} + C_X + C_{AR}). \tag{23}$$

The approximate rise time at the aggressor coupling node is

$$t_{r,0} = t_r + \frac{t_{A,0}}{(1 - e^{-1})}. \tag{24}$$

The effective capacitances for the victim net and for the right segment of the aggressor net can be calculated as follows based on (15) and (22), as well as the approximate rise time of the coupling point on the aggressor net.

$$C^v_{\text{eff}} = \left(1 - \frac{t_X}{t_{r,0}}(1 - e^{-t_{r,0}/t_V})\right) C_X \tag{25}$$

$$C^r_{\text{eff}} = \left(1 - \frac{R_{AR}C_{AR}}{t_{r,0}}(1 - e^{-t_{r,0}/R_{AR}C_{AR}})\right) C_{AR}. \tag{26}$$

A more accurate approximated time constant corresponding to the dominant pole at the aggressor net is calculated, therefore, as

$$t_A = C_{AL}R_A + (C_{AM} + C^v_{\text{eff}} + C^r_{\text{eff}})(R_A + R_{AL}). \tag{27}$$

And the aggressor time-domain voltage waveform is obtained as

$$V_A(t) = \begin{cases} \frac{t}{t_r} - \frac{t_A}{t_r}\left(1 - e^{-t/t_A}\right), & t \le t_r \\ 1 - \frac{t_A}{t_r}\left(1 - e^{-t_r/t_A}\right)e^{-(t-t_r)/t_A}, & t > t_r \end{cases}. \tag{28}$$

Now, instead of using a simple ramp function at node 1 as the aggressor net waveform, we use the more accurate form as shown in (28). The s-domain aggressor waveform can be derived using Laplace transformation of (28)

$$V_A(s) = \left(\frac{1}{s^2 t_r} + \frac{t_A}{t_r}\left(\frac{1}{s + \frac{1}{t_A}} - \frac{1}{s}\right)\right)\left(1 - e^{-st_r}\right). \tag{29}$$

Similarly, we use dominant pole approximation on the victim net and the following equation is obtained

$$V_{out}(s) = \frac{t_X s}{t_V s + 1} V_A(s) \tag{30}$$

where

$$t_X = C_X(R_V + R_{VL}) \tag{31}$$

$$t_V = C_{VL}R_V + (C_{VM} + C_X)(R_V + R_{VL}) + C_{VR}(R_V + R_{VL} + R_{VR}). \tag{32}$$

Time-domain noise voltage output is an inverse Laplace transform of the above equation, which can be divided into the following two regions

1) Region I ($0 \le t \le t_r$)

$$V^I_{\text{out}}(t) = \frac{t_X}{t_r}\left(1 + \alpha e^{-t/t_A} - \beta e^{-t/t_V}\right) \tag{33}$$

2) Region II ($t > t_r$)

$$V^{II}_{\text{out}}(t) = \frac{t_X}{t_r}\left(\alpha\left(e^{-t/t_A} - e^{-(t-t_r)/t_A}\right) - \beta\left(e^{-t/t_V} - e^{-(t-t_r)/t_V}\right)\right) \tag{34}$$

where $\alpha = t_A/(t_V - t_A)$ and $\beta = t_V/(t_V - t_A)$.

It can be easily observed that the noise voltage increases monotonically in Region I and it increases, then decreases in Region II. Therefore, the maximum noise voltage always occurs in Region II. By solving the equation $dV^{II}_{\text{out}}(t)/dt = 0$, we obtain the time noise voltage reaches the peak

$$t_{\text{peak}} = t_r + \frac{t_V t_A}{t_V - t_A} \ln\left(\frac{1 - e^{-t_r/t_A}}{1 - e^{-t_r/t_V}}\right). \tag{35}$$

We compare the noise waveforms generated by the dominant pole and the double pole models with that obtained using SPICE simulation of the 6-node template circuit in Fig. 8. The following circuit parameters are assumed. The driving resistances of the aggressor and the victim are 500 and 1000 Ω, respectively; the wire resistances are 100 Ω each; the ground capacitances are 50 fF each and the coupling capacitance is 150 fF. The rising slope of the input signal is 200 ps. Clearly, the waveform obtained using the double pole approximation is more accurate than that obtained by the dominant pole approximation. First, the noise peak time is very close to the correct value. Second, the derivative of the voltage waveform is continuous throughout the entire range, which is important for many optimization methods. Third, the noise voltage matches the simulated result very well over the entire range.

Peak noise voltage is a metric to determine whether the noise on a signal wire exceeds the static noise margin of the receivers. However, the duration of the noise should also be considered to measure the effect of the noise on the receiver output. In the literature, this is accomplished by using the noise width metric. In the presence of multiple aggressor nets, however, the noise width of the glitches generated by each single

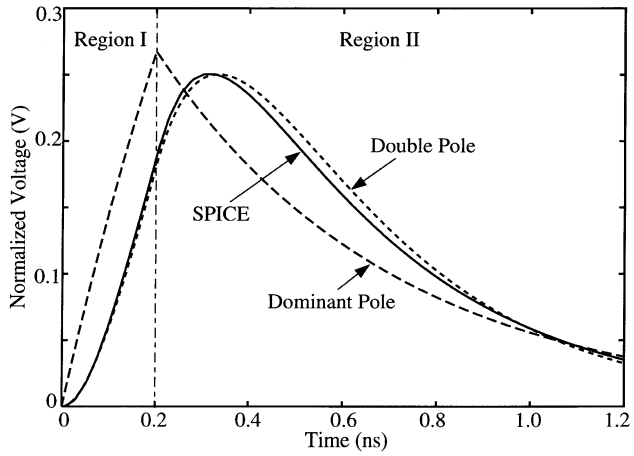

Fig. 8. Comparison of noise waveforms in template circuit modeling.

aggressor cannot be simply combined as we can do with the peak noise voltage. Therefore, in this paper, we use another metric, noise area, in addition to noise height for dynamic noise analysis. Similar to noise height, noise area produced by each aggressor can be simply added to derive the total noise area generated when all aggressors are switching simultaneously. In practice, a simple noise waveform shape (for example, a rectangular or triangular waveform), is usually adopted. In such case, the total noise area can be easily translated back to the noise width metric, if necessary.

The area under the noise voltage waveform can be calculated by integrating the noise voltage equations, which turns out to be a very simple equation

$$\text{Area} = C_X(R_V + R_{VL}). \tag{36}$$

Note that the above expression is normalized to the supply voltage V_{DD}. The noise area equation holds for both dominant pole and double pole-based formulas.

We have tested the proposed formulas using 5000 randomly generated circuits having the 6-node template circuit topology. In those test circuits, the parameter ranges are as follows. The driver resistances are 20–2000 Ω; the wire resistances are 10–300 Ω; the ground capacitances are 20–200 fF; the coupling capacitance is 30–300 fF; and the input signal slope is 20–500 ps. Results are shown in Table I, where the comparison is between the dominant pole method for the 6-node circuit and the double pole method for the 6-node circuit. In terms of peak noise voltage, the dominant pole model has an average error of 8.3% and only 66.8% of the test circuits have errors within 10% of the SPICE result. The double pole model has significantly improved accuracy with an average error of only 2.3% and the 3σ error is 8%. The noise area formula is always exact.

V. Experiments

We have applied the proposed method to industrial circuits to further verify its correctness. The set of circuits available to us is 30 noise-prone nets, having tree structures with distributed *RC* elements from a high-performance processor designed in a 0.15-μm process technology. In those nets, the drivers of the circuits were replaced by linear resistors using the technique described in [12]. Logic correlations and overlapping of timing windows were also considered.

Some information of the circuits and comparison of the model results with SPICE simulation results on noise area as well as noise amplitude are shown in Table II. The second column shows the number of total aggressor nets and number of switching aggressor nets. The third column is the total number of *RC* elements in a given circuit. The average number of all the nets is 128. The lengths of the victim nets in millimeters are shown in the fourth column, and vary from less than 1 mm up to about 3 mm, with an average victim net length of 2.1 mm.

TABLE I
Model Errors in Peak Noise Voltage Based on 5000 Random Circuits

Method	Dominant pole	Double pole
Average error	8.3%	2.3%
Cases $\leq$ 5% error	37.3%	92.6%
Cases $\leq$ 10% error	66.8%	99.9%
3σ error	26%	8%

Noise area calculated using the proposed model is compared with those obtained using SPICE simulation in Table II. According to (36), the noise area is a function of only three variables: C_X, R_V, and R_{VL}, none of which is affected by the reduction techniques described in previous sections. Therefore, both the proposed approach and the simple approach which does not use effective capacitances, generate the same results in terms of noise area. Also, in agreement with the fact that (36) is exact, we observe the model errors are very small with an average value of 1.3%.

In Table II, we also compare the peak noise voltage values of the simple approach and the proposed approach with SPICE simulation results. The simple approach also adopts the 6-node template as well as the double pole model. However, the quiet aggressor nets are grounded during superposition and the resistive shielding effect in the tree branches is not considered. The proposed approach has an average peak noise error of 2.7% and the maximum error is 7.8%. The majority of the circuits (23 out of 30) have an error of less than 5%. In comparison, the simple method has an average error of 11.7% and a maximum error of 21.3%. It underestimates the crosstalk noises by more than 10% in 21 out of 30 circuits.

The proposed crosstalk noise estimation method is also very efficient. Each aggressor net is first reduced to a 2-π circuit and the reduced 2-π circuits for the aggressor nets are then reused each time when we apply the superposition method to calculate the total noise at the victim net. Therefore, the total time complexity of the proposed method is linear with respect to the number of elements in the noise circuit cluster and it is the best possible complexity to obtain the combined noise caused by all the aggressor nets. The CPU time for the 30 industrial circuits ranges from 0.03 to 0.15 ms on a 933-MHz PC running Linux. It may be noted that the set of circuits available to us is extracted for post-layout verification purpose. Much coarser interconnect segmentation can be used for our purpose of fast crosstalk noise estimation. In such scenario, the runtime of the proposed methodology can be greatly reduced.

VI. Conclusion

We have proposed an efficient crosstalk noise estimation framework that maintains the efficiency of prior works and has much improved accuracy. Novel reduction techniques were proposed for quiet aggressor net reduction, which models the effect that the quiet aggressor nets are affected by the victim waveform, and for tree branch reduction, which considers the effect of resistive shielding of branch interconnects. A double pole-based formula is derived for analytical model of the reduced 6-node template circuit. Experimental results on industrial circuits is promising.

The proposed crosstalk noise estimation methodology is very efficient and, therefore, is suitable as a noise estimation engine for various physical design tools such that coupling noise, together with circuit area, speed, power consumption, and others, can be used as a metric for design optimization.

TABLE II
CIRCUIT INFORMATION AND EXPERIMENTAL RESULTS ON 30 INDUSTRIAL CASES

Circuit	NA/NS	#RC	Length	Noise area ($ns \cdot V$)			Noise peak (V)				
			(mm)	SPICE	Model	(err%)	SPICE	Simple	(err%)	Model	(err%)
1	2 / 2	97	1.49	0.535	0.541	(1.3)	0.839	0.772	(8.0)	0.900	(7.4)
2	9 / 9	215	2.80	0.610	0.596	(2.2)	0.793	0.672	(15.3)	0.782	(1.4)
3	4 / 4	132	2.62	0.634	0.644	(1.7)	0.790	0.670	(15.3)	0.812	(2.7)
4	5 / 5	128	2.80	0.535	0.522	(2.6)	0.786	0.650	(17.3)	0.796	(1.3)
5	9 / 9	167	2.79	0.599	0.600	(0.2)	0.765	0.659	(13.9)	0.765	(0.0)
6	9 / 9	201	2.61	0.518	0.520	(0.3)	0.772	0.658	(14.7)	0.731	(5.4)
7	4 / 4	112	1.47	0.396	0.393	(0.7)	0.764	0.704	(7.9)	0.772	(1.0)
8	7 / 6	144	0.71	0.706	0.706	(0.0)	0.716	0.727	(1.7)	0.761	(6.3)
9	7 / 5	121	2.07	1.223	1.220	(0.2)	0.713	0.675	(5.2)	0.727	(2.1)
10	2 / 2	88	2.82	0.483	0.477	(1.2)	0.710	0.559	(21.3)	0.654	(7.8)
11	7 / 7	174	2.70	0.449	0.459	(2.4)	0.704	0.589	(16.3)	0.717	(1.8)
12	3 / 3	116	1.69	0.409	0.404	(1.3)	0.695	0.609	(12.4)	0.682	(1.9)
13	3 / 3	117	1.69	0.407	0.404	(0.7)	0.693	0.609	(12.0)	0.682	(1.5)
14	10 / 10	191	2.58	0.388	0.397	(2.1)	0.682	0.576	(15.5)	0.658	(3.5)
15	3 / 3	106	1.47	0.324	0.329	(1.5)	0.686	0.632	(7.9)	0.702	(2.3)
16	2 / 2	106	1.71	0.410	0.409	(0.4)	0.690	0.612	(11.3)	0.683	(1.1)
17	7 / 4	171	2.82	8.965	8.969	(0.0)	0.686	0.634	(7.6)	0.693	(1.0)
18	3 / 3	112	1.68	0.402	0.407	(1.1)	0.688	0.615	(10.6)	0.687	(0.1)
19	2 / 2	94	1.49	0.272	0.269	(1.2)	0.685	0.597	(12.9)	0.684	(0.2)
20	3 / 3	118	1.70	0.407	0.406	(0.1)	0.685	0.606	(11.5)	0.676	(1.3)
21	2 / 2	98	1.65	0.392	0.397	(1.1)	0.684	0.610	(10.9)	0.682	(0.4)
22	7 / 7	181	2.73	0.647	0.651	(0.6)	0.663	0.559	(15.7)	0.658	(0.7)
23	5 / 3	136	2.72	8.870	8.860	(0.1)	0.662	0.600	(9.4)	0.714	(7.7)
24	2 / 2	95	1.48	0.246	0.250	(1.5)	0.656	0.577	(12.0)	0.657	(0.2)
25	3 / 3	116	1.70	0.404	0.401	(0.7)	0.656	0.581	(11.4)	0.640	(2.5)
26	5 / 5	114	2.41	0.337	0.352	(4.3)	0.632	0.580	(8.2)	0.616	(2.5)
27	10 / 9	247	2.75	0.380	0.385	(1.2)	0.632	0.548	(13.3)	0.585	(7.4)
28	5 / 5	138	2.71	0.389	0.406	(4.5)	0.626	0.562	(10.2)	0.625	(0.1)
29	2 / 2	95	1.47	0.275	0.282	(2.5)	0.626	0.561	(10.4)	0.639	(2.0)
30	9 / 9	187	2.12	0.226	0.224	(1.0)	0.622	0.561	(9.8)	0.585	(5.9)
Average	**5.0 / 4.7**	**128**	**2.1**			**1.3%**			**11.7%**		**2.7%**
Maximum									**21.3%**		**7.8%**

REFERENCES

[1] K. L. Shepard and V. Narayanan, "Noise in deep submicron digital design," in *Proc. Int. Conf. Computer-Aided Design*, Nov. 1996, pp. 524–531.

[2] A. Devgan, "Efficient coupled noise estimation for on-chip interconnects," in *Proc. Int. Conf. Computer-Aided Design*, Nov. 1997, pp. 147–151.

[3] M. Kuhlmann, S. S. Sapatnekar, and K. K. Parhi, "Efficient crosstalk estimation," in *Proc. Int. Conf. Computer Design*, Sept. 1999, pp. 266–272.

[4] J.-S Yim and C.-M Kyung, "Reducing cross-coupling among interconnect wires in deep-submicron datapath design," in *Proc. Design Automation Conf.*, June 1999, pp. 485–490.

[5] P. Chen and K. Keutzer, "Toward true crosstalk noise analysis," in *Proc. Int. Conf. Computer-Aided Design*, Nov. 1999, pp. 132–137.

[6] A. Glebov, S. Gavrilov, D. Blaauw, S. Sirichotiyakul, C. Oh, and V. Zolotov, "False-noise analysis using logic implications," in *Proc. Int. Conf. Computer-Aided Design*, Nov. 2001, pp. 515–521.

[7] C. Visweswariah, R. A. Haring, and A. R. Conn, "Noise considerations in circuit optimization," *IEEE Trans. Computer-Aided Design*, vol. 19, pp. 679–690, June 2000.

[8] 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.

[9] P. Feldmann and R. W. Freund, "Efficient linear circuit analysis by Padé approximation via the Lanczos process," *IEEE Trans. Computer-Aided Design*, vol. 14, pp. 639–649, May 1995.

[10] A. Odabasioglu, M. Celik, and L. T. Pileggi, "PRIMA: Passive reduced-order interconnect macromodeling algorithm," *IEEE Trans. Computer-Aided Design*, vol. 17, pp. 645–654, Aug. 1998.

[11] K. L. Shepard, V. Narayanan, and R. Rose, "Harmony: Static noise analysis of deep submicron digital integrated circuits," *IEEE Trans. Computer-Aided Design*, vol. 18, pp. 1132–1150, Aug. 1999.

[12] R. Levy, D. Blaauw, G. Braca, A. Dasgupta, A. Grinshpon, C. Oh, B. Orshav, S. Sirichotiyakul, and V. Zolotov, "ClariNet: A noise analysis tool for deep submicron design," in *Proc. Design Automation Conf.*, June 2000, pp. 233–238.

[13] V. Zolotov, D. Blaauw, S. Sirichotiyakul, C. Oh, and R. Panda, "Noise propagation and failure criteria for VLSI designs," in *Proc. Int. Conf. Computer-Aided Design*, Nov. 2002, pp. 587–594.

[14] A. Vittal and M. Marek-Sadowska, "Crosstalk reduction for VLSI," *IEEE Trans.Computer-Aided Design*, vol. 16, pp. 290–297, Mar. 1997.

[15] A. Vittal, L. H. Chen, M. Marek-Sadowska, K.-P. Wang, and S. Yang, "Crosstalk in VLSI interconnects," *IEEE Trans. Computer-Aided Design*, vol. 18, pp. 1817–1824, Dec. 1999.

[16] J. Cong, D. Z. Pan, and P. V. Srinivas, "Improved crosstalk modeling for noise constrained interconnect optimization," in *Proc. Asia South Pacific Design Automation Conf.*, Jan. 2001, pp. 373–378.

[17] L. H. Chen and M. Marek-Sadowska, "Aggressor alignment for worst-case crosstalk noise," *IEEE Trans. Computer-Aided Design*, vol. 20, pp. 612–621, May 2001.

[18] S. Sirichotiyakul, D. Blaauw, C. Oh, R. Levy, V. Zolotov, and J. Zuo, "Driver modeling and alignment for worse-case delay noise," in *Proc. Design Automation Conf.*, June 2001, pp. 720–725.

[19] P. R. O'Brien and T. L. Savarino, "Modeling the driving-point characteristic of resistive interconnect for accurate delay estimation," in *Proc. Int. Conf. Computer-Aided Design*, Nov. 1989, pp. 512–515.

[20] J. Qian, S. Pullela, and L. T. Pillage, "Modeling the effective capacitance for the *RC* interconnect of CMOS gates," *IEEE Trans. Computer-Aided Design*, vol. 13, pp. 1526–1535, Dec. 1994.