Improved Crosstalk Modeling for Noise Constrained Interconnect Optimization

Jason Cong, David Zhigang Pan and Prasanna V. Srinivas†
Department of Computer Science, University of California, Los Angeles, CA 90095 *
† Magma Design Automation, Inc., 2 Results Way, Cupertino, CA 95014.

Abstract

This paper presents a much improved, highly accurate yet efficient crosstalk noise model, the 2-π model, and applies it to noise-constrained interconnect optimizations. Compared with previous crosstalk noise models of similar complexity, our 2-π model takes into consideration many key parameters, such as coupling locations (near-driver or near-receiver), and the coarse distributed RC characteristics for victim net. Thus, it is very accurate (less than 6% error on average compared with HSPICE simulations). Moreover, our model provides simple closed-form expressions for both peak noise amplitude and noise width, so it is very useful for noise-aware layout optimizations. In particular, we demonstrate its effectiveness in two applications: (i) Optimization rule generation for noise reduction using various interconnect optimization techniques; (ii) Simultaneous wire spacing to multiple nets for noise constrained interconnect minimization.

1 Introduction

In deep sub-micron (DSM) circuit designs, the coupling capacitance between adjacent nets has become a dominant component as taller and narrower wires are now placed closer to each other [1]. The coupling capacitance not only leads to excessive signal delays, but also causes potential logic malfunctions (see [2] for a tutorial). The latter problem is especially serious for designs with higher clock frequencies, lower supply voltages, and usage of dynamic logic since they have lower noise margin. To make sure a final layout to be noise immune, accurate yet efficient noise models are needed to guide interconnect optimizations at various stages.

Recently, a number of simple crosstalk noise models were proposed. By solving telegraph equations directly, [3, 4] obtained a set of analytical formulae for peak noise of capacitively coupled bus lines. But their approaches handle only fully coupled bus structures, not partially coupled lines or general RC trees. The work in [5] modeled each aggressor and victim net by an L-type RC circuit and obtained closed-form expression for both peak noise upper bound and noise-over-time integral. It showed much improvement on the pure charge sharing model, but it assumed a step input for aggressor. Extensions to [5] were made by [6, 7, 8], to consider a saturated ramp input, or a Pi-type lumped RC circuit. Most of these models, however, did not consider the distributed nature of an RC network, which is needed in DSM designs. In [9], an elegant Elmore-delay like peak noise model was obtained for general RC trees, and it guarantees to be an upper bound. However, [9] assumed an *infinite* (non-saturated) ramp input. Thus, it may significantly over-estimate the peak noise, especially for large victim nets, and small aggressor slews (very likely in DSM). In fact, the peak noise obtained from [9] may even be larger than the supply voltage. Recent work in [8] can handle distributed RC network and saturated ramp input. But it can be shown that the model in [8] has up to 100% over estimation compared to the model in [9] when the aggressor transition time is much larger than the victim net delay (see Section 2).

In this paper, we develop a much improved crosstalk noise model, called the *2-π model*. It overcomes major drawbacks of existing models by taking into consideration many key parameters, such as the aggressor slew at the coupling location, the coupling location at the victim net (near-driver or near-receiver), and the coarse distributed RC characteristics for victim net. Our model is very accurate, with less than 6% error on average compared with HSPICE simulations. Moreover, it has simple *closed-form* expressions for both *peak noise* and *noise width* and provides very clear physical meaning for key noise contribution terms. All these characteristics of our 2-π model make it ideal to guide noise-aware layout optimizations *explicitly*.

The rest of this paper is organized as follows. Section 2 presents the 2-π model and its analytical solutions for time-domain *waveform*, *peak noise* and *noise width*, together with the model validation by HSPICE simulations. In the next two sections, we demonstrate two applications of our 2-π model. Section 3 provides a set of interconnect optimization *rules* to guide effective noise reduction and Section 4 uses the 2-π model in a simultaneous wire spacing problem for noise-constrained area minimization to multiple nets. The conclusion follows in Section 5.

2 An Improved 2-π Crosstalk Noise Model

In this section, we first present the 2-π model and derive its analytical time-domain waveform. Then we focus on two key metrics for the 2-π model, i.e., peak noise (amplitude) and noise width, and derive simple closed-form expressions for them. We then extend the 2-π model to handle general RC trees, followed by extensive validation of the model.

2.1 2-π Model and its Analytical Waveform

For simplicity, we first explain our 2-π model for the case where the victim net is an RC line. We will extend the 2-π model to a general RC tree in Section 2.3. For a victim net with some aggressor nearby, as shown in Fig. 1 (a), let the aggressor voltage pulse at the coupling location be a saturated ramp input with transition time (i.e., slew) being t_r,[1] and the interconnect length of the victim net before the coupling, at the coupling and after the coupling be L_s, L_c and L_e, respectively.

The 2-π type reduced RC model is generated as shown in Fig. 1 (b) to compute the crosstalk noise at the receiver. It is called 2-π model because the victim net is modeled as two π-type RC circuits, one before the coupling and one after the coupling. The victim driver is modeled by effective resistance R_d. Other RC parameters C_x, C_1, R_s, C_2, R_e, and C_L are computed from the geometric information from Fig. 1 (a) in the following manner. The coupling node (node 2) is set to be the center of the coupling portion of the victim net, i.e., $L_s + L_c/2$ from the source. Let the upstream and downstream interconnect resistance/capacitance at Node 2 be

*This work is partially supported by Semiconductor Research Corporation under Contract 98-DJ-605 and IBM Research Fellowship. Part of the work was performed while David Pan worked as a summer intern at Magma Design Automation, Inc., in 1999. The current address of David Pan is: IBM T. J. Watson Research Center, Yorktown Heights, NY 10598.

[1] The aggressor net RC characteristics and its driver/load information are incorporated into the slew t_r at the coupling location. It can be estimated from simple slew models or timing analysis tools.

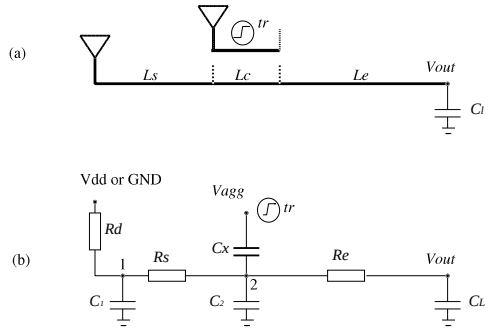

Fig. 1: (a) The layout of a victim net and an aggressor above it. (b) The 2-π crosstalk noise model.

R_s/C_s and R_e/C_e, respectively. Then capacitance values are set to be $C_1 = C_s/2$, $C_2 = (C_s + C_e)/2$ and $C_L = C_e/2 + C_l$. Compared with [5, 6] which only used one lumped RC for the victim net, it is obvious that our 2-π model can model the *coarse* distributed RC characteristics. In addition, since we consider only those *key* parameters, the resulting 2-π model can be solved analytically.

From Fig. 1 (b), we have the impedance at node 1, Z_1 satisfying the following

$$\frac{1}{Z_1} = \frac{1}{R_d} + sC_1$$

Then at node 2, we have

$$\frac{1}{Z_2} = \frac{1}{(Z_1 + R_s)} + sC_2 + \frac{1}{R_e + \frac{1}{sC_L}}$$

Denote the s-domain voltage at node 2 by $V_2(s)$, then

$$V_2(s) = \frac{Z_2}{Z_2 + \frac{1}{sC_x}} \cdot V_{agg}(s).$$

The output voltage V_{out} in the s-domain is

$$V_{out}(s) = V_2(s) \cdot \frac{\frac{1}{sC_L}}{R_e + \frac{1}{sC_L}}. \quad (1)$$

Substituting Z_1, Z_2 and V_2 into $V_{out}(s)$, we have

$$\begin{aligned} V_{out}(s) &= \frac{Z_2}{Z_2 + (\frac{1}{sC_x})} \cdot \frac{1/sC_L}{R_e + 1/sC_L} \cdot V_{agg}(s) \\ &= \frac{a_2 s^2 + a_1 s}{s^3 + b_2 s^2 + b_1 s + b_0} \cdot V_{agg}(s) \quad (2) \end{aligned}$$

where the coefficients are

$$\begin{aligned} a_2 &= K_1/K_2 \\ a_1 &= (R_d + R_s)C_x/K_2 \\ b_2 &= ((C_2 + C_x) \cdot (R_e C_L (R_d + R_s) + R_d R_s C_1) \\ &+ R_d R_e C_1 C_L + C_L R_d R_s C_1)/K_2 \\ b_1 &= ((R_d + R_s)(C_x + C_2 + C_L) + (R_e C_L + R_d C_1))/K_2 \\ b_0 &= 1/K_2 \\ K_1 &= C_x R_d R_s C_1 \\ K_2 &= R_d R_s C_1 C_L R_e (C_x + C_2) \end{aligned}$$

Writing the transform function $H(s)$ into the pole/residue form:

$$H(s) = \frac{a_2 s^2 + a_1 s + a_0}{s^3 + b_2 s^2 + b_1 s + b_0} \equiv \frac{k_1}{s - s_1} + \frac{k_2}{s - s_2} + \frac{k_3}{s - s_3}.$$

The three poles s_1, s_2, and s_3 are the three roots of $s^3 + b_2 s^2 + b_1 s + b_0 = 0$, which can be obtained analytically using standard mathematical techniques (details omitted due to page limitation). After each pole/residue pair is obtained, its corresponding time domain function is just $f_i(t) = k_i e^{s_i t}$ $(i = 1, 2, 3)$.

For the aggressor with saturated ramp input with normalized $V_{dd} = 1$ and transition time t_r, i.e.,

$$v_{agg}(t) = \begin{cases} t/t_r & 0 \le t \le t_r \\ 1 & t > t_r, \end{cases}$$

its Laplace transformation is

$$V_{agg}(s) = \frac{1 - e^{-st_r}}{s^2 t_r} \quad (3)$$

Then for each pole/residue pair, the s-domain output $V_{out_i}(s) = \frac{k_i}{s - s_i} \cdot V_{agg}(s)$, and its inverse Laplace is just the convolution of $f_i(t)$ and $g(t)$,

$$\begin{aligned} v_{out_i}(t) &= f_i(t) * g(t) = \int_0^t f_i(t-u) g(u) du \\ &= \begin{cases} -\frac{k_i(1 + s_i t)}{s_i^2 t_r} + \frac{k_i e^{s_i t}}{s_i^2 t_r} & 0 \le t \le t_r \\ -\frac{k_i e^{s_i (t - t_r)}}{s_i^2 t_r} + \frac{k_i e^{s_i t}}{s_i^2 t_r} - \frac{k_i}{s_i} & t > t_r \end{cases} \quad (4) \end{aligned}$$

Therefore, the final noise voltage waveform is simply the summation of the voltage waveform from each pole/residue pair.

$$v_{out}(t) = v_{out_1}(t) + v_{out_2}(t) + v_{out_3}(t). \quad (5)$$

The 2-π model has been tested extensively and its waveform from (5) can be shown to be almost identical compared to HSPICE simulations. Detailed model validation results will be presented in Section 2.4.

2.2 Closed-Form Noise Amplitude and Width

Although the closed-form noise waveform has been derived in the previous subsection, the solution by itself is still quite complicated. Moreover, it provides little intuition about some key measurements for crosstalk noise, such as noise peak amplitude and noise width, which are very important to guide noise reduction by interconnect optimizations. Simple closed-form expressions for these measurements are highly desired, since they provide more insight about how various interconnect parameters affect the crosstalk noise and to what extent. In this subsection, we will further simplify the original 2-π model and derive closed-form formulae for noise amplitude and noise width.

Using dominant-pole approximation method in a similar manner like [7, 10, 11], we can simplify (2) into

$$V_{out}(s) \approx \frac{a_1 s}{b_1 s + b_0} \cdot V_{agg}(s) = \frac{t_x (1 - e^{-st_r})}{s t_r (s t_v + 1)} \quad (6)$$

where the coefficients are

$$\begin{aligned} t_x &= (R_d + R_s)C_x \quad (7) \\ t_v &= (R_d + R_s)(C_x + C_2 + C_L) + (R_e C_L + R_d C_1) \quad (8) \end{aligned}$$

It is interesting to observe that t_x is in fact the RC delay term from the upstream resistance of the coupling element times the coupling capacitance, while t_v is the distributed Elmore delay of victim net. We will further discuss their implications later.

Computing the inverse Laplace transform of (6), we can obtain the following simple time domain waveform

$$v_{out}(t) = \begin{cases} \frac{t_x}{t_r}(1-e^{-t/t_v}) & 0 \le t \le t_r \\ \frac{t_x}{t_r}(e^{-(t-tr)/t_v} - e^{-t/t_v}) & t > t_r \end{cases} \quad (9)$$

It is easy to verify that in the above noise expression, v_{out} monotonically increases at $0 \le t \le t_r$, and monotonically decreases at $t > t_r$. So the peak noise will be at $t = t_r$, with the value of

$$v_{max} = \frac{t_x}{t_r}(1 - e^{-t_r/t_v}). \quad (10)$$

The above expression of v_{max} can be degenerated to some special cases to encapsulate noise models derived in previous works. As $t_r \to 0$ (i.e., a step input), $v_{max} \to \frac{t_x}{t_v}$, which is in the same form as in [5] (without interconnect resistance) and [8] (with interconnect resistance). In the case of $t_r >> t_v$ (actually $t_r > 3t_v$ is enough), $v_{max} \to \frac{t_x}{t_r}$, which is in the same form as [9].

It is also interesting to compare with the recent work by [8], where the peak noise with saturated ramp input can be written as $v'_{max} = \frac{t_x}{t_v + t_r/2}$. Although obtained from a totally different approach, v'_{max} from [8] is indeed a first-order approximation of our v_{max} in (10), since

$$\frac{t_x}{t_r}(1 - e^{-t_r/t_v}) = \frac{t_x}{t_v}[1 - \frac{1}{2}\frac{t_r}{t_v} + \ldots] \quad (11)$$

$$\approx \frac{t_x}{t_v}\frac{1}{1 + \frac{1}{2}\frac{t_r}{t_v}} = \frac{t_x}{t_v + t_r/2} \quad (12)$$

However, such approximation is only valid when $t_r < t_v$. It will be much off when $t_r >> t_v$, since it throws away larger terms. This explains why v'_{max} in [8] gives twice peak noise of Devgan model when $t_r >> t_v$, i.e., 100% over estimation. It also explains the results in Table II of [8] that as t_r gets larger (from 100ps to 500ps), the average error of peak noise expression from [8] gets larger (from 6% to 10%).

Peak noise amplitude v_{max} is not the only metric to characterize noise. Under some circumstance, even the peak noise exceeds certain threshold voltage, a receiver may still be noise immune. This can be characterized by some noise amplitude versus width plots. The noise width is defined as follows.

Definition 1 Noise Width: *Given certain threshold voltage level v_t, the noise width for a noise pulse is defined to be the length of time interval that noise spike voltage v is larger or equal to v_t.*

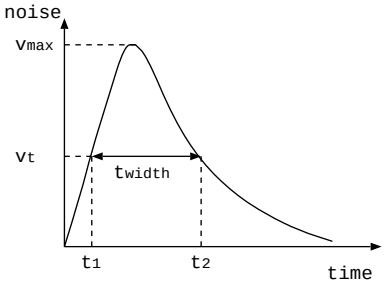

Fig. 2: Illustration of the noise width.

From Eqn. (9), we can compute t_1 and t_2, and thus the noise width

$$t_2 - t_1 = t_v ln\left[\frac{(t_x - t_r v_t)(e^{t_r/t_v} - 1)}{t_r v_t}\right] \quad (13)$$

In this paper, we set the threshold voltage v_t to be half of the peak noise voltage, $v_t = v_{max}/2$. Then, the noise width of (13) is simplified into

$$t_{width} = t_2 - t_1 = t_r + t_v ln\left[\frac{1 - e^{-2t_r/t_v}}{1 - e^{-t_r/t_v}}\right] \quad (14)$$

Note that t_x is cancelled out in (14). One can easily verify the following property for the noise width.

Lemma 1 *The noise width t_{width} is a monotonically increasing function of t_r and t_v, i.e., $\partial t_{width}/\partial t_r > 0$ and $\partial t_{width}/\partial t_v > 0$, and it is bounded by $t_r < t_{width} < t_r + t_v ln 2$.*

2.3 Extension to RC Trees

Our 2-π model can be easily extended to a victim net in general RC tree structures. To compute the crosstalk noise at a certain sink (receiver) S_j, we build the corresponding 2-π model as shown in Fig. 3. It is similar to that shown in Fig. 1, with the same upstream and downstream resistances. The only difference is that we now incorporate the lumped capacitance at each branch on the path from source to sink S_j, i.e., C_{b1}, ... C_{bi}. We will add these C_{bi}'s into C_1, C_2 or C_L in the following weighted manner:

- If a branch B_i is between the source and the coupling center, let its distance to the source be $\alpha(L_s + L_c/2)$. Then $(1-\alpha)C_{bi}$ goes to C_1 and αC_{bi} goes to C_2.
- If a branch B_i is between the sink and the coupling center, let its distance to the sink be $\beta(L_e + L_c/2)$. Then $(1-\beta)C_{bi}$ goes to C_L and βC_{bi} goes to C_2.

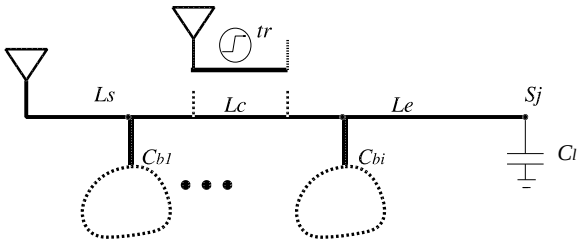

Fig. 3: Extension of the 2-π model for general RC trees.

Actually, it can easily be shown that in the resulting 2-π model of multiple-pin nets, t_x is the same as that in 2-pin nets while t_v is still the Elmore delay from the source to sink S_j, but now with branching capacitances. The analytical solutions of the 2-π model remain the same. Note that for a coupling element (e.g., C_x) not on the path from the source to sink S_j (i.e., coupling with some branching elements), the computation of t_x only takes C_x's upstream resistance common to the path from the source to sink S_j (in the same manner as the Elmore delay computation).

As for the time complexity, since we have the closed-form expressions for the poles, residues, and waveform for each pole/residue pair, the computation time for transfer function and waveform for a given 2-π model can be done in constant time. To reduce the original circuit to the 2-π model, we only need a linear traversal (to compute upstream/downstream interconnect resistance/capacitance at the coupling node) of the victim net, which can be done in linear time as well as in [5, 9]. It is obviously the lower bound of the computational complexity for any reasonable noise model.

2.4 Validation of the 2-π Model

The 2-π model and its analytical formulae for peak noise (10) as well as noise width (14) have been tested extensively and shown to work remarkably well compared to HSPICE simulations. To obtain high fidelity and to detect the corner scenarios, we run our 2-π model, Devgan model [9], Vittal model [8], and HSPICE simulations on 1000 randomly generated circuits with realistic parameters in a 0.18μm technology (extracted based on NTRS [1]). For the test circuits, the driver resistance R_d is from 20 to 2000 Ω, the loading capacitance C_l is from 4 to 50 fF, the length parameters L_s, L_c, and L_e are from 1 to 2000 μm, the wire width/spacing is either 1x or 2x minimum width/spacing, and the aggressor slew is from 10 to 500 ps. Our experiments show that the average errors for peak noise estimation using Devgan, Vittal and our 2-π model are 589%, 9%, and less than 4%, respectively. Table 1 summarizes the percentage of nets that fall into certain error ranges using the 2-π model with closed-form peak noise and noise width expressions from (10) and (14) compared with those from running HSPICE simulations. We can see that using our model, both peak noise and noise width are on average within 4% error, and almost 95% nets have less than 10% errors.

Table 1: The percentage of nets that fall into the error ranges for peak noise (v_{max}) and noise width (t_{width}) from the 2-π model.

Error range	v_{max}	t_{width}
within +/- 20%	99.9%	98.8%
within +/- 15%	95.8%	96.8%
within +/- 10%	93.5%	94.6%
within +/- 5%	83.1%	84.7%
Average error	3.7%	3.6%

We have also tested the 2-π model on a set of randomly generated multiple-pin nets with general RC tree structures. Our experimental results show that our 2-π model still works surprisingly well for general RC trees. Fig. 4 shows the scatter diagram comparing the 2-π model (y-axis) with HSPICE (x-axis) simulations for 20 randomly generated four-pin nets (i.e., with two branches). The experimental setting is the same as those for 2-pin nets. The branching wire length ranges from 1 to 2000 μm. The branching location can be anywhere from driver to receiver. HSPICE simulations are performed on distributed RC networks by dividing each long wire into every 10μm segment. Again, for all test circuits, the 2-π model gives very good estimation (close to the $y = x$ line in the scatter diagram). The average errors for peak noise and noise width are just 4.3% and 5.89%, respectively.

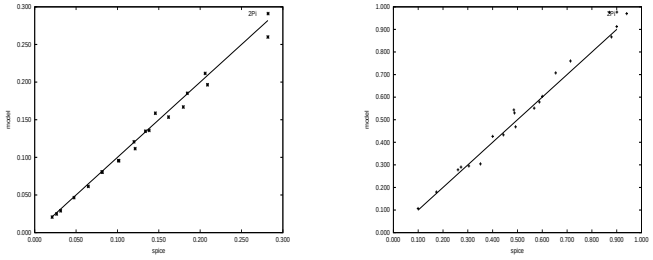

Fig. 4: Comparison of 2-π model versus HSPICE simulation for 20 randomly generated RC trees for (a) peak noise, (b) noise width.

3 Application I: Optimization Rules for Noise Reduction

In this section, we perform some in-depth parametric studies inside the 2-π model and provide a set of optimization rules for noise reduction. Since it is well understood that buffer insertion can help to reduce crosstalk noise [12, 13] and our model can also be used as an internal noise evaluator to guide buffer insertion, we will not include it here. Rather, we will focus on a direct-connected net. Since noise pulse below certain threshold voltage (V_{th}) will not cause a receiver to malfunction, we will mainly focus on the peak noise v_{max} reduction. However, we will consider the noise width when the peak noise exceeds the threshold voltage (Section 3.5).

3.1 Driver Sizing

Intuitively, driver sizing can help to reduce the peak crosstalk noise since a stronger driver has more capability to sustain a noise spike. Although this is true in most cases, our model does indicate some situation under which increasing driver size (i.e., reduce R_d) may not help to reduce the peak noise. Consider

$$\frac{\partial v_{max}}{\partial R_d} = \frac{C_x}{t_r}\left[1 - \frac{1 + \frac{(R_d+R_s)(C_1+C_2+C_x+C_L)}{t_v} \cdot \frac{t_r}{t_v}}{e^{t_r/t_v}}.\right]$$

If $(R_d + R_s)(C_1 + C_2 + C_x + C_L) \leq t_v$ (i.e., $R_sC_1 < R_eC_L$ after substituting t_v), then $\frac{\partial v_{max}}{\partial R_d} > 0$ and sizing up a driver will reduce noise. However, if $(R_d + R_s)(C_1 + C_2 + C_x + C_L) > t_v$ (i.e., $R_sC_1 > R_eC_L$), and $t_r << t_v$, one may have the situation that $\frac{\partial v_{max}}{\partial R_d} \leq 0$. [2] Consider the extreme case of $t_r \to 0$, then the peak noise is

$$\frac{C_x \cdot R_d + R_sC_x}{(C_1 + C_x + C_2 + C_L) \cdot R_d + R_s(C_x + C_2 + C_L) + R_eC_L}.$$

Let $k_1 = \frac{C_x}{C_1+C_x+C_2+C_L}$ and $k_2 = \frac{R_sC_x}{R_s(C_x+C_2+C_L)+R_eC_L}$. It is easy to verify that if $k_1 > k_2$ (i.e., $R_sC_1 < R_eC_L$), $\frac{\partial v_{max}}{\partial R_d} > 0$, while if $k_1 < k_2$, $\frac{\partial v_{max}}{\partial R_d} < 0$. It is also interesting to see that v_{max} in fact bounded by k_1 and k_2, i.e., $min(k_1, k_2) \leq v_{max} \leq max(k_1, k_2)$. That is to say, no matter how one optimally sizes a driver, there is still some noise lower bound, and just doing driver sizing may not help to reduce the peak noise below the desired level. To summarize, we have the following rule.

Rule 1 *If $R_sC_1 < R_eC_L$, then sizing up the victim driver strength (i.e., reduce effective R_d) will reduce peak noise. However, if $R_sC_1 > R_eC_L$ and $t_r << t_v$, driver sizing will not help to reduce peak noise. In either situation, there is certain lower bound for peak noise that can be achieved by just doing driver sizing.*

3.2 Near-Driver versus Near-Receiver Coupling

This subsection investigates the effects of different coupling locations on peak noise. From $v_{max} = \frac{(R_d+R_s)C_x}{t_r}(1 - e^{-t_r/t_v})$, we know that as the coupling element move toward the receiver, R_s increases. Meanwhile, the Elmore delay t_v increases because more "lumped" capacitance is now near the receiver, but the increase rate shall be much less than that of R_s and the overall effect to the peak noise is determined by the increase of R_s. This proposition is validated by extensive simulations. As an example, Fig. 5 shows that v_{max} increases monotonically (almost linear) by 40% as an aggressor moves from near-driver ($L_s = 0$) to near-receiver ($L_e = 0$), meanwhile t_v only increases by 9%. This leads to the following interconnect optimization rule for noise reduction.

[2] Intuitively, it corresponds to the situation of a very strong aggressor coupling at a near-receiver location to the victim net.

Rule 2 *During topology generation/routing of a noise-sensitive victim net, one shall avoid near-receiver coupling, especially to its strong aggressors.*

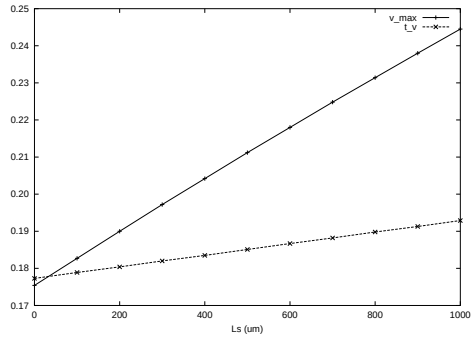

Fig. 5: Variations of v_{max} and t_v to different coupling locations (L_s from 0 to $1mm$) for a victim net of $2mm$ long, $L_c = 1mm$, coupling spacing of $0.33\mu m$, wire width of $0.22\mu m$, $R_d = 300\Omega$, $C_l = 10fF$, $L_c = 1mm$, $t_r = 50ps$.

It shall be pointed out since [5, 6] only have one lumped RC for the victim net, they do not differentiate between near-source and near-sink coupling. The model in [9] also discourages near-sink coupling and confirms Rule 1, but it usually gives too conservative peak noise.

3.3 Shield Insertion

Section 3.2 suggests that we should avoid near-receiver coupling, so we shall insert shielding (non-aggressive) wires close to noise-sensitive receiver. Another aspect for a shielding wire to reduce peak noise is by increasing "lumped" capacitance of the victim net.

Lemma 2 *The peak noise monotonically decreases as C_1, C_2, or C_L increases. For the same amount of capacitance increase, it is most effective at C_L, and least effective at C_1, i.e.,*

$$\frac{\partial v_{max}}{\partial C_L} < \frac{\partial v_{max}}{\partial C_2} < \frac{\partial v_{max}}{\partial C_1} < 0$$

From Lemma 2, the following layout guidance rule to reduce crosstalk noise.[3]

Rule 3 *The placement/insertion of non-aggressive (quiet) neighbors around a victim net will help to reduce the crosstalk noise. The preferred position for shield insertion is near a noise-sensitive net's receiver.*

3.4 Wire Spacing versus Sizing

Lemma 3 *The peak noise monotonically increases as C_x increases, i.e., $\partial v_{max}/\partial C_x > 0$*

Definition 2 Monotone Capacitance Model: *For a wire segment with fixed width, its coupling capacitance monotonically increases while its ground capacitance monotonically decreases as its spacing to neighboring wire decreases.*

[3] [5] also suggests that shield insertion (which they call "intentional overlapping") to reduce noise. However, it does not differentiate between near-driver and near-receiver shield insertion.

Lemma 4 *The peak noise monotonically decreases as wire spacing increases, under the monotone capacitance model.*

Our extensive capacitance extraction shows that Monotone Capacitance Model usually holds for DSM designs, thus wire spacing is always an effective way to reduce noise (especially when other methods reach their limitations, e.g., driver sizing). The penalty is on the area side. For a given area constraint, however, our study shows that wire spacing is consistently more effective than wire sizing for noise reduction. Thus, we have

Rule 4 *Wire spacing is always an effective way to reduce noise, with an area penalty. For a given area constraint, wire spacing is usually more effective than wire sizing for crosstalk noise reduction.*

It shall be noted that previous works (e.g., [14]) showed that proper wire sizing could significantly reduce delay with consideration of coupling capacitance. It will be interesting to explore the wire sizing/spacing tradeoff for both delay and noise consideration in the future.

3.5 On Noise Amplitude-Width Product

Sometimes, a receiver may still be noise-immune even the peak noise exceeds certain threshold voltage. This can be characterized by some noise amplitude versus width plots, which can then be transformed into an *amplitude* (A) versus *amplitude-width* (AW) product (A-AW plot) [5]. This subsection reveals some interesting property on the noise amplitude-width product. From (10) and (14) the AW product can be written as

$$AW \quad = \quad (R_d + R_s)C_x \cdot f(x) \tag{15}$$

where $f(x) = \frac{e^x - e^{-x}}{1 - e^{-x}} ln \frac{e^x - e^{-x}}{1 - e^{-x}}$ and $x = t_r/t_v$. It can be verified that $f(x) \in [ln2, 1]$ (i.e., [0.69, 1]). This important property suggests that AW is essentially determined by $(R_d + R_x)C_x$. Although techniques like increasing C_1, C_2 or C_L (e.g., shield insertion) can reduce AW, they cannot go below the lower bound $ln2 \cdot (R_d + R_x)C_x$. The most effective way to reduce AW is to reduce C_x (e.g., by spacing), R_d (by driver sizing), and R_x (by wire sizing).

Rule 5 *The noise amplitude-width product has a lower bound of $ln2(R_d + R_s)C_x$, and an upper bound of $(R_d + R_s)C_x$. Other parameters such as C_1, C_2, R_e, C_L only play a minor role in it. The effective ways to reduce AW are wire spacing, driver sizing and wire sizing.*

4 Application II: Simultaneous Wire Spacing for Multiple Nets

To demonstrate the effectiveness of our 2-π model, we apply it to a simultaneous wire spacing problem for multiple nets. It is formulated as follows.

Given: (1) The initial layout of multiple nets and their noise constraints; (2) the minimum wire spacing between each coupling pair.

Minimize: The total area or equivalently, the total spacing between all nets.

Subject to: No noise violation for each net.

This problem may be formulated into some nonlinear programming problem under simple formula-based capacitance models. But in DSM designs, table-based capacitance model is usually required for adequate accuracy, which makes the problem difficult to solve due to lack of analytical expressions (possible non-convexity, etc.). Instead, we will use a sensitivity-based spacing algorithm

(SBSA) to solve it, as illustrated in Fig. 6. The *noise reduction sensitivity* Δv_{ij} at some spacing s_{ij} (between two adjacent nets i and j) is defined to be the total noise reduction for those noise-violating receivers in nets i and j, due to some nominal spacing increase to s_{ij}, say Δs_{ij}. The algorithm starts from some minimum spacing as given by the input. As long as there is noise violation, it will check each spacing s_{ij} that is a possible cause of the noise violation, and compute its noise reduction sensitivity. Given the same nominal area increase ΔA, we compute the spacing increase $\Delta s_{ij} = \Delta A / l_{ij}$, where l_{ij} is the coupling length for s_{ij}. We pick the spacing s_{mn} that has the maximum noise reduction sensitivity with the same nominal area increase, and increase it by Δs_{ij}. Then, we update the noise information and iterate until there is no noise violation.

Sensitivity-Based Spacing Algorithm

```
1. initialize spacings;
2. while (there is noise violation) {
3.     Δv_max ← 0;
4.     foreach spacing s_ij between any adjacent nets i and j
           that either i or j has noise violation {
5.         Δs_ij = ΔA/l_ij;
6.         compute noise reduction sensitivity Δv_ij;
7.         Δv_max ← max(Δv_ij, Δv_max);
8.     }
9.     increase s_mn with Δv_max by Δs_mn;
10.    update noise for affected nets;
11. }
```

Fig. 6: A simultaneous wire spacing algorithm for noise-constrained area minimization for multiple nets.

We apply our sensitivity-based wire spacing algorithm to a 4-bit fully parallel bus of 1 mm long, with $R_d = 180\Omega$, $C_l = 23fF$, wire width of $0.44\mu m$, and $t_r = 50ps$. The noise constraint is set to be 0.2 V_{dd}. Table 2 lists the spacings between adjacent bus lines using SBSA. We compare the resulting spacings (s_{12} denotes the spacing between the first and the second bus line, and so on. TS denotes the total spacing) from our metrics with two other metrics [9] (Devgan) and [8] (Vittal). We list results under two different Δs, 0.33 and 0.11 μm, respectively. It can be seen that using Devgan and Vittal models may lead to too conservative spacing by as much as 70% and 31%, respectively, due to their peak noise over-estimation. It is also interesting to see that, comparing with a straightforward equal spacing algorithm (i.e., $s_{12} = s_{23} = s_{34}$, with the total spacing TS_{ES} at the last row of Table 2), our SBSA algorithm will use much less area, with area reduction by up to 11% (total spacing of 5.28 μm versus 5.94 μm for 2-π model with Δs = 0.33 μm). So our SBSA is quite effective in practice.

Table 2: Spacing for noise control of a 4-bit bus, using different noise metrics.

spacing (μm)	Δs = 0.33μm			Δs = 0.11μm		
	Devgan	Vittal	2-π	Devgan	Vittal	2-π
s_{12}	2.64	1.98	1.65	2.42	1.98	1.54
s_{23}	3.63	2.97	1.98	3.52	2.75	2.20
s_{34}	2.64	1.98	1.65	2.42	1.98	1.54
TS	8.91	6.93	5.28	8.36	6.71	5.28
TS_{ES}	8.91	6.93	5.94	8.58	6.93	5.61

5 Conclusion

We have developed in this work a much improved crosstalk noise model, with less than 6% error on average compared with HSPICE simulation, for both peak noise voltage and noise width estimations. Compared to existing models with the same complexity, our model is much more accurate and it provides a unified view for them. The model has been shown to be very effective to guide noise-aware interconnect optimization. In this paper, we assume a saturated ramp input for the aggressor net. We have also obtained the closed-form peak noise formula for the 2-π model under the exponential aggressor input, and our experiments show about the same accuracy as that under the saturated ramp input using HSPICE simulations. Interested reader can refer to [15] for details. We expect that our 2-π model will be useful in many other applications at various levels to guide noise-aware DSM circuit designs.

Acknowledgments

The authors would like to thank Tim Burks from Magma Design Automation, Inc., for helpful discussions.

References

[1] Semiconductor Industry Association, *National Technology Roadmap for Semiconductors*, 1997.

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

[3] T. Sakurai, "Closed-form expressions for interconnection delay, coupling, and crosstalk in VLSIs," *IEEE Trans. on Electron Devices*, vol. 40, pp. 118–124, 1993.

[4] H. Kawaguchi and T. Sakurai, "delay and noise formulas for capacitively coupled distributed RC lines," in *Proc. Asia and South Pacific Design Automation Conf.*, pp. 35–43, 1998.

[5] A. Vittal and M. Marek-Sadowska, "Crosstalk reduction for VLSI," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 16, pp. 290–98, 1997.

[6] S. Nakagawa, D. M. Sylvester, J. McBride, and S.-Y. Oh, "On-chip cross talk noise model for deep-submicrometer ulsi interconnect," *Hewlett-Packard Journal*, vol. 49, pp. 39–45, Aug. 1998.

[7] A. B. Kahng, S. Muddu, and D. Vidhani, "Noise and delay uncertainty studies for coupled rc interconnects," in *IEEE International ASIC/SOC Conference*, pp. 3–8, 1999.

[8] A. Vittal, L. Chen, M. Marek-Sadowska, K.-P. Wang, and S. Yang, "Crosstalk in VLSI interconnections," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 18, no. 2, pp. 1817–24, 1999.

[9] A. Devgan, "Efficient coupled noise estimation for on-chip interconnects," in *Proc. Int. Conf. on Computer Aided Design*, pp. 147–153, 1997.

[10] M. Kuhlmann, S. Sapatnekar, and K. Parhi, "Efficient crosstalk estimation," in *Proc. IEEE International Conference on Computer Design*, pp. 266–272, 1999.

[11] E. Acar, A. Odabasioglu, M. Celik, and L. Pileggi, "S2p: a stable 2-pole RC delay and coupling noise metric IC interconnects," in *Proceedings 9th Great Lakes Symposium on VLSI*, pp. 60–3, 1999.

[12] C. J. Alpert, A. Devgan, and S. Quay, "Buffer insertion for noise and delay optimization," in *Proc. Design Automation Conf.*, pp. 362–7, 1998.

[13] C.-P. Chen and N. Menezes, "Noise-aware repeater insertion and wire sizing for on-chp interconnect using hierarchical moment-matching," in *Proc. Design Automation Conf.*, pp. 502–506, June 1999.

[14] J. Cong, L. He, C.-K. Koh, and Z. Pan, "Global interconnect sizing and spacing with consideration of coupling capacitance," in *Proc. Int. Conf. on Computer Aided Design*, pp. 628–633, 1997.

[15] Z. Pan, *Interconnect Synthesis and Planning for High-Performance IC Designs.* PhD thesis, University of California, Los Angeles, 2000.