Worst Case RLC Noise with Timing Window Constraints [*]

Jun Chen
Electrical Engineering Department
University of California, Los Angeles
jchen@ee.ucla.edu

Lei He
Electrical Engineering Department
University of California, Los Angeles
lhe@ee.ucla.edu

ABSTRACT

We study the problem of determining the switching patterns and switching times of multiple aggressors to generate the worst-case crosstalk noise (WCN) for a quiet victim in RLC interconnect structures under the constraints of both aggressor switching windows and the victim sampling window. We consider the signal routing direction and show that it has a significant impact under RLC model but not under RC model. We propose a new $SS + AS$ algorithm that considers the timing window constraints and has linear time complexity and high accuracy, only underestimating WCN by 2% on average. We show that although RC model usually severely underestimates WCN with timing window constraints, it *does* overestimate when both the aggressor switching windows and the victim sampling window are small enough. We conclude that RLC model is needed for accurate modeling of WCN in GHz+ design.

1. INTRODUCTION

The coupling induced crosstalk noise gains growing importance in deep-submicron circuits and systems with higher clock frequency. The worst-case noise (*WCN*) defined as the maximum crosstalk noise peak has been studied in [1]. It is assumed that driver and receiver sizes, wire spacings, and net ordering are given, and interconnects can be modeled by distributed RC circuits. Then, the WCN problem is formulated as finding the alignment of switching times for multiple aggressors such that WCN is induced.

As we move to GHz+ designs, the inductive crosstalk noise can no longer be ignored [2, 3]. The WCN problem becomes much more complicated under RLC interconnect model. We need to consider (i) switching pattern generation in addition to alignment of switching times for multiple aggressors, as the same direction switching assumed for the WCN problem under RC model does *not* always lead to WCN under RLC model [4]; (ii) coupling between both adjacent and non-adjacent interconnects, while the WCN problem under RC model only takes into account coupling between adjacent interconnects; and (iii) routing direction of signal wires. It is defined as whether the signal is routed from left (top) to right (down) or vice versa, and has a significant impact on WCN under RLC model but not under RC model.

The worst-case noise problem under RLC model has been studied in [4]. In this work, the authors pointed out that different switching directions of the aggressor must be considered in WCN problem under RLC model, and formulate the problem as finding both switching pattern and switching time for the aggressors such that the noise amplitude is maximized. However, the paper assumes the same routing direction for all the interconnects. Furthermore, it did not consider the switching timing window of the aggressors or the sampling window of the victim. It has been shown that ignoring the timing window constraints can significantly overestimate WCN [1] and greatly increase the number of false violations [5]. In this work, we consider the impact of signal net routing direction and study the WCN problem under RLC model with both the aggressor switching windows and victim sampling window. The rest of the paper is organized as follows: In section 2, we discuss the interconnect and device model, and review the WCN algorithms both under RC model and under RLC model. In section 3, we study the impact of the signal routing direction on WCN. In section 4 we develop new algorithms for the WCN problem with timing window constraints. Finally, we conclude our paper in section 5.

2. PRELIMINARIES

2.1 Interconnect and device models

We study the interconnect structure with a quiet victim wire (in short, the victim) and multiple aggressor wires (in short, the aggressors). The signal nets can be routed in either direction (i.e., from left(top) to right (down) or vice versa). Moreover, we assume that aggressors may have arbitrary switching patterns (i.e., switching high or switching low).

We assume that all drivers (receivers) have a uniform size, and are cascaded inverters. For the best accuracy, we use BSIM model [6] for the predicted ITRS $0.10\mu m$ technology to model all the drivers and receivers. BSIM model is a nonlinear device model. In contrast, there are linearized device models, such as the effective switching resistance model [7] and C_{eff} model [8]. The effective switching resistance model uses a fixed-value resistor to model a device. Interconnects with drivers and receivers become linear circuits under this model, leading to inaccurate estimation of WCN [4]. The C_{eff} model is able to catch the device nonlinearity for a single RC or RLC tree, and has been used for the worst-case delay problem under RC model [9]. We plan to study its applicability to the WCN problem under RLC model in the

---

[*] This paper is partially supported by NSF CAREER award CCR-0093273, SRC grant 1100, a UC MICRO grant sponsored by Analog Devices, Fujitsu Laboratories of America, Intel and LSI Logic, and a Faculty Partner Award by IBM. Address comments to lhe@ee.ucla.edu.

future works.

Interconnects can be modeled by either RC or RLC circuits. In this work, we assume that all wires are aligned and have uniform width and spacing, and construct a π-type circuit for every 200μm long wire segment for both RC and RLC models. We only consider the coupling capacitance between adjacent wires because coupling capacitance between nonadjacent wires is negligible. For RC model, both self inductance and mutual inductance are ignored. For RLC model, the accurate model considers self inductance for each wire segment, and mutual inductance between *any* two wire segments, even though they may belong to the same net. Such a RLC circuit model is called full model in [10]. The full model is accurate and applicable to either aligned or unaligned buses. It has been shown that for aligned buses, a normalized model with a much reduced complexity achieves a similar accuracy when compared to the full model [10, 11]. In this work, we study aligned buses, so we use the normalized model for RLC models.

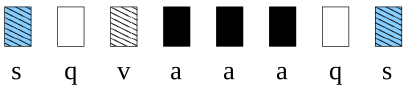


Figure 1: Six-bit aligned bus with two shields

The cross section of the aligned interconnect structure on which we carry out experiments is shown in figure 1, where **v** is the quiet victim, **q** is a quiet wire, **a** is an aggressor, and **s** is a shield. We assume the two shields because they are realistic due to the power/ground wires in the same or adjacent layers and they affect the noise value greatly [4]. We carry out SPICE simulations on the RLC circuits of the interconnects with the BSIM models of the drivers and receivers to validate our WCN algorithms to be presented in this paper. We use the predicted ITRS 0.10μm technology presented in table 1. We assume that the input rising time is 33ps, which is 10% of the clock period of the predicted 3GHz clock. We assume uniform receiver size and driver size. We measure noise at the inputs of receivers and report noise normalized with respect to VDD. It is worthwhile to point out that our algorithms can be applied to any accurate interconnect analysis methods.

Technology	ITRS 0.10μm
Signal rising time	33ps
Wire length	1000μm
Wire thickness	0.75μm
Wire width	0.6μm
Driver size	100x to 150x
Receiver size	10x

Table 1: Experiment settings

2.2 WCN under RC model

If only capacitive coupling is considered, there is no resonance in the noise waveform. When an aggressor switches, there is only one noise peak on the victim with the polarity same as that of the aggressor signal. To achieve the maximum noise, all the noise peaks should have a same polarity, and so do all the aggressor signals. Therefore, the WCN problem under RC model can be simplified as the alignment of aggressor switching times to maximize the noise on the victim, without considering aggressor switching patterns.

The following algorithms have been widely used: (i) *Simultaneous switching* (SS): All the aggressors switch simultaneously. WCN is approximated by the maximum noise value on the victim. And (ii) *Superposition* (SP): Find the maximum noise peak when only one aggressor switches, then approximate WCN by the sum of all such noise peaks. The *Aligned Switching* (AS) has been proposed in [1], where we find the *peak time* as the time of the maximum noise peak when only one aggressor switches, then simulate the interconnect structure with all aggressors switching at the times *aligned* according to the above peak times (see an alignment example in Figure 2). The maximum noise in the last simulation is WCN. According to [1], AS closely approximates WCN with underestimation less than 5%, SS always underestimates WCN, and SP can severely overestimate or underestimate WCN.

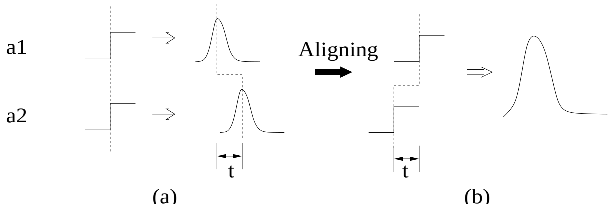


Figure 2: Alignment operation illustrated using two aggressors. (a) We simulate the interconnects with only one aggressor switching in each simulation, and find the skew t between noise peaks. (b) We simulate the interconnects with both aggressors switching. When their switching times are aligned by t, the overall noise due to the two aggressors is likely maximized [1].

2.3 WCN under RLC model without timing window constraints

The authors of [4] studied the WCN problem under RLC model without timing constraints. The authors extended SS, AS and SP algorithms to consider the switching pattern generation in addition to switching time alignment as follows,

- *Simultaneous switching* (SS): All aggressors switch simultaneously in the same direction. WCN is approximated by the maximum noise on the victim.
- *Superposition* (SP): Find the maximum noise peak for each aggressor when only this aggressor switches. WCN is approximated by the sum of *amplitudes* (absolute values) of all such peaks.
- *Aligned switching* (AS): Obtain *individual* noise waveform by simulating the interconnect structure with only one aggressor switching for each time, then simulate the circuit with multiple aggressors using the following switching times and patterns:
  - (i) *PP alignment:* align the maximum positive peaks of individual noise waveforms, and all aggressors switch in a same direction;
  - (ii) *NN alignment:* align the maximum negative peaks of individual noise waveforms, and all aggressors switch in a same direction;

(iii) *PN alignment:* align the peaks of maximum amplitude, and aggressors have switching directions such that all the aligned peaks have a same polarity.

WCN is approximated by the maximum noise among the above three simulations. Experiments have shown that none of the three kinds of alignments defined above is always better than the others, so all the three alignments are needed by AS algorithm.

The authors of [4] proposed a new algorithm $SS + AS$, where WCN is approximated by the larger one between the results obtained by SS and AS. They showed that $SS+AS$ has linear complexity and high accuracy with an average underestimation of 3%. We will extend the algorithms above to consider the aggressor switching windows and the victim sampling window in section 4.

2.4 $SA + GA$ algorithm

To compare different algorithms, we developed simulated annealing algorithm (SA) and genetic algorithm (GA) for the WCN problem under RLC model. We select the larger noise between the results from SA and GA as the accurate WCN. We call this algorithm as $SA+GA$. In SA algorithm, the value of the cost function is proportional to the maximal noise. There are two types of moves: 1. Adjust the arrival time of a randomly picked aggressor by a random factor from 0 to 10%; 2. Reverse the switching pattern of a randomly picked aggressor. We start the SA at an initial temperature of 50 and terminate it at 0.01. The temperature decreases by a factor of 0.9 and the number of moves at a particular temperature is equal to $100 \times n$, where n is the number of aggressors. For GA algorithm, each individual solution (*chromosome*) is encoded as an ordered array of aggressor switching time and switching pattern pairs. The population of each generation is equal to $4n$. The fitness of each individual is equal to the maximum noise on the victim. Two types of genetic operations are performed: 1. Crossover: produce offspring by exchanging parts of the settings of the aggressors between two parents; 2. Mutation: produce offspring by randomly changing the selected aggressors' switching time and switching pattern of a selected parent. The probability of a parent being selected is proportional to its fitness. The crossover and mutation probabilities are 0.5 and 0.3 respectively. The GA process terminates when there is no improvement for 20 continuous generations. In the following of this paper, we use the result from $SA + GA$ as the comparison base for other algorithms.

3. IMPACT OF SIGNAL ROUTING DIRECTIONS

Signals are routed either from left (top) to right (down) or from right (down) to left (top). The routing direction has little impact on WCN under RC model, but has significant impact under RLC model. In figure 3, we present two signal nets in two different patterns of routing direction. One net is the victim and the other is the aggressor. The wire lengths are $1000\mu m$. We run SPICE simulations to study the noise of the two different settings. In table 2, we summarize the noises under both RC and RLC models. From the table, we can see that the noises for the two topologies are almost the same under RC model with only 0.2% difference, but are much different under RLC model with 29% difference. This can be explained as follows: the noise under RC model is due to the coupling capacitance that is insensitive to the routing direction. Therefore, the routing direction has little impact on RC noise. When we consider the inductance, the different routing directions will result in different current flow directions and in turn different loop inductances (see Figure 3), which results in large difference in the noise waveform even for a single aggressor. Therefore, the routing direction should be considered in the noise analysis under RLC model. In this work, we assume the routing directions are given, but may not be the same for all the signal nets. Considering the routing direction, the WCN problem under RLC model is formulated as,

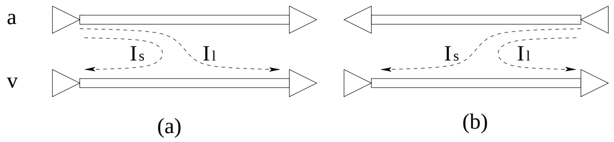


Figure 3: Different routing patterns of two signal wires.

Model	a	b	difference
RC	0.1151	0.1149	-0.2%
RLC	0.1658	0.2138	+29.0%

Table 2: Noises of different routing patterns

FORMULATION 1. *Given a quiet victim and multiple aggressors in a pre-routed interconnect structure, find switching patterns and switching times for all aggressors such that the noise in the victim has maximal amplitude.*

We carry out a set of experiments using the six-bit aligned bus structure in Figure 1 but with different routing directions. The driver size is 150x. We compare the algorithms from [4] with $SA + GA$ in table 3. The two opposite directions are marked as '0' and '1' respectively. We take the results from $SA+GA$ as the accurate results. From the table, we can see $SS+AS$ algorithm has the highest accuracy with an average error of 1% and a maximum error of 3%. When aggressors are routed in different directions, SS underestimates the WCN with an error much larger than the error when all the aggressors are routed in the same direction, because the skew between the maximum peaks of aggressors are larger with different routing directions. The SP algorithm underestimates or overestimates the WCN with errors up to 21%. The average of the absolute errors of SP is 12.07%. Therefore SP does not approximate WCN well. We also can see that the RC model severely underestimates the WCN by up to 75%. Furthermore, different routing directions has little impact on the noise values under RC model (the difference is too small to be shown in the table), but it can result in up to 50% difference under RLC model. Therefore, we must consider the routing direction in the WCN problem under RLC model.

4. WCN PROBLEM WITH TIMING WINDOW CONSTRAINTS

In real design practice, there is a switching timing window for each aggressor. The switching timing window is the time interval between the earliest and latest switching times

Direction	RLC					RC
s q v a a a q s	SA+GA	SP	SS	AS	SS+AS	WCN
Space = 0.6						
0 0 0 0 0 0 0 0	0.133	0.112	0.129	0.124	0.129	0.101
0 1 0 1 0 1 0 0	0.193	0.234	0.153	0.191	0.191	0.101
0 1 0 0 1 0 1 0	0.176	0.196	0.0775	0.176	0.176	0.101
0 0 0 1 1 1 0 0	0.200	0.172	0.199	0.198	0.199	0.101
Space = 1.2						
0 0 0 0 0 0 0 0	0.130	0.120	0.126	0.128	0.128	0.043
0 1 0 1 0 1 0 0	0.172	0.196	0.0902	0.171	0.171	0.043
0 1 0 0 1 0 1 0	0.152	0.166	0.0371	0.151	0.151	0.043
0 0 0 1 1 1 0 0	0.151	0.146	0.149	0.150	0.150	0.043
Average Error	0.00%	12.07%	-25.97%	-1.53%	-1.00%	-56.13%
Maximum Error	0.00%	+21.24%	-75.59%	-6.77%	-3.01%	-75.00%

Table 3: Noise on a quiet victim with different routing directions. The average error for SP is calculated based on the absolute difference of noise.

of the aggressor. For the victim, there is a sampling window at the input of its receiver. The sampling timing window is the time interval between the earliest setup time and the latest hold time of the flip-flop at the far end. In this section, we develop the algorithms for WCN problem for RLC interconnects with the timing constraints of both aggressor switching windows and the victim sampling window.

4.1 Algorithm

To find the WCN under timing window constraints, we extend the algorithms in [4]. We still consider three kinds of alignment: PP, NN and PN alignments. We first discuss PN alignment, where we align the aggressors according to the absolute maximum peak of each aggressor. As shown in Figure 4, the specific steps in PN alignment include: (1) Simulation: We simulate the bus with one aggressor switching each time to obtain the individual noise waveform on the victim for each aggressor, and then for each individual noise waveform, we approximate the waveform by a piece-wise linear waveform which consists peak-to-peak straight lines. Because of the oscillation of the noise waveform in RLC circuits, normally the peaks are narrow and sharp and the linear model approximates the waveform very well for the purpose of WCN problem. (2) Depolarization: We construct a new waveform which is the absolute value of the original piece-wise linear waveform. (3) Expansion: We expand the waveform according to the aggressor's timing window. The expansion procedure is shown in figure 5. In this example there is one aggressor with switching timing window of $tw = t2 - t1$. During the expansion, we first expand each noise peak by tw, and then find the contour of all expanded peaks (i.e., the largest values at each time point). We record the peak polarity and switching timing of each region so that we can obtain the switching pattern and switching time of the aggressor later. (4) Summation: To consider the noise contributions from all the aggressors, we sum up the waveform contours of all the aggressors to get an overall waveform contour. We find the time region with the maximum noise value in the waveform within the sampling window of the victim and the correspondent switching pattern and switching time of each aggressor. Finally, we carry out one-time simulation with the determined switching pattern and time, and use the maximum noise from this last simulation as WCN. We summarize the algorithm in table 4.

The algorithms for PP and NN alignment with timing window constraints are similar. Because in these two alignments all the aggressors have the same switching pattern, we may not need to change the polarity of noise by changing the switching pattern. Therefore, we do not need to use the absolute value of the waveform but instead use the original waveform. In the step of expansion, for PP alignment we get the largest noise (most positive) for the waveform contour, and for NN alignment we get the smallest noise (most negative) for waveform contour. The remaining steps are the same as those in PN alignment. We measure the time complexity in terms of the number of simulations needed for the analysis. The time complexity for the alignment switching algorithm is $n + 3$ because we need n individual simulations for each aggressor and one simulation for each type of alignment.

We also extend the SS algorithm to consider the timing window constraints. We first determine all the overlapped regions for the timing windows of all the aggressors. For each of such regions, we find all the aggressors that can switch in the region, and find the simultaneous switching noise of those aggressors within the sampling window of the victim. The largest noise among the simultaneous switching noises of all the overlapped regions is WCN. The time complexity of SS algorithm is $2n - 1$, where n is the number of aggressors, because each switching window has two ends and thus there are at most $2n - 1$ overlapped regions. For each overlapped region, one simulation is required, so the worst case is $2n - 1$.

After we obtain the maximal noise values from AS and SS, the $AS + SS$ algorithm approximate WCN by the larger one of the two. The worst-case time complexity of the extended $AS + SS$ with timing window constraints is $3n + 2$, the sum of the runtime for AS and SS.

4.2 Experiments

To verify our algorithms, we carry out a set of experiments to compare $SS + AS$ algorithm with $GA + SA$ algorithm. In this set of experiments, the timing windows and routing directions are randomly generated for both the victim and the aggressors. We carry out the experiments on the aligned bus structure shown in Figure 1 with different routing directions. The driver size is 100x. We summarize the experiment results in table 5. We do not compare the SP algorithm because it is meaningless to sum the maximum peaks without considering the timing windows. From the results, we can

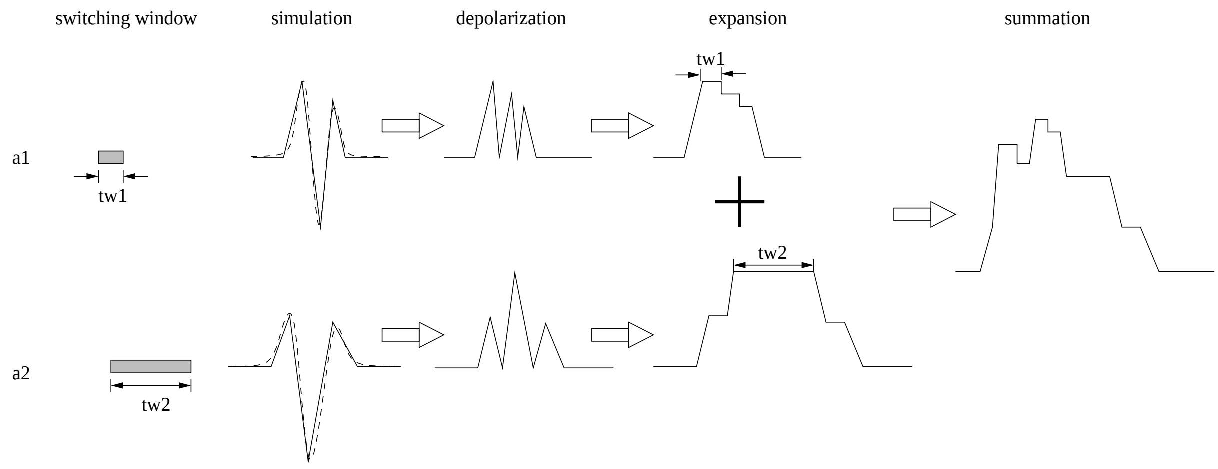


Figure 4: PN alignment with timing windows

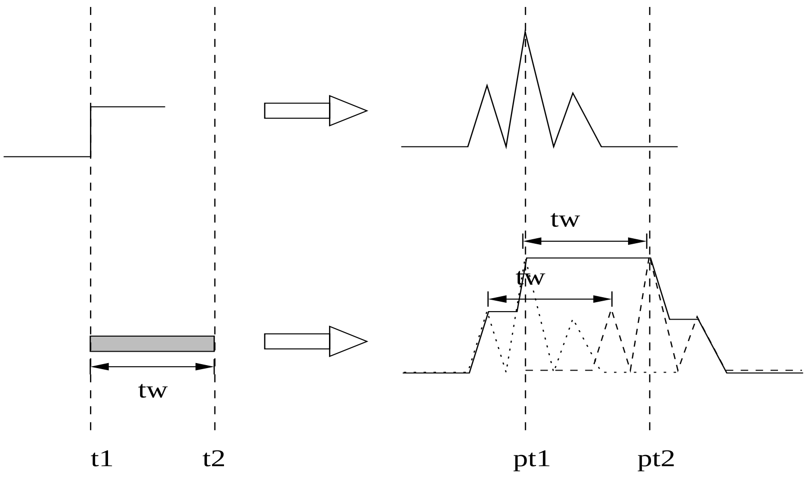


Figure 5: Expansion of noise waveform

Step 1:	Simulation
	For each aggressor simulate with only the aggressor switching and others quiet.
	Proximate the noise waveform by piece-wise linear waveform for each aggressor.
Step 2:	Depolarization
	Obtain the waveform with the absolute value of the original waveform for each aggressor.
Step 3:	Expansion
	Expand each waveform peak by the width of the timing window and obtain the contour of the expanded waveform.
Step 4:	Summation
	Sum the contour waveforms in Step 3 for all the aggressors.
	Find the switching pattern and switching time that generate the maximal noise in the accumulated waveform within the sampling window of the victim.
	Simulate with the determined switching pattern and switching time to obtain WCN.

Table 4: Steps to determine the WCN with time window constraint

see that $SS + AS$ approximates WCN very well with an average error of 2% and a maximum error of 5%. In this set of experiments, the SS algorithm generally behaves worse than the AS algorithm does due to time window constraints of both the aggressors and the victim. However, with certain settings SS still can obtain larger noise than AS as shown in table 5. In table 5, we also present the WCN without the timing window constraints but with the same bus configurations. We can see that the WCN with timing window constraints can be up to 75% smaller than its peer without the timing window constraints. Thus, timing window constraints must be considered in WCN analysis to reduce false crosstalk violations.

Furthermore, we compare WCN under the RLC and RC

Routing Direction	Timing Window (t_{start}, t_{end}) (ps)				Noise				WCN(No window)
s q v a a a a q s	v	a1	a2	a3	SA+GA	SS	AS	SS+AS	
Spacing = 0.6 μm									
0 0 0 0 0 1 0 0	(300,325)	(100, 200)	(100, 275)	(50,150)	0.118	0.112	0.105	0.112	0.163
0 1 0 1 0 1 0 0	(300,350)	(0, 200)	(150, 350)	(50,250)	0.164	0.162	0.163	0.163	0.174
0 1 0 0 1 0 1 0	(350,400)	(50, 250)	(100, 350)	(300,600)	0.156	0.134	0.155	0.155	0.171
0 0 0 1 1 1 0 0	(350,400)	(250, 450)	(100, 300)	(0,200)	0.0510	0.0506	0.0510	0.0510	0.195
Spacing = 1.2 μm									
0 0 0 0 0 1 0 0	(300,325)	(100, 200)	(100, 275)	(50,150)	0.0705	0.0371	0.0695	0.0695	0.131
0 1 0 1 0 1 0 0	(300,350)	(0, 200)	(150, 350)	(50,250)	0.127	0.118	0.121	0.121	0.143
0 1 0 0 1 0 1 0	(350,400)	(50, 250)	(100, 350)	(300,600)	0.110	0.0608	0.109	0.109	0.133
0 0 0 1 1 1 0 0	(350,400)	(250, 450)	(100, 300)	(0,200)	0.0492	0.0481	0.0489	0.0489	0.137
Average Error					0.00%	-14.42%	-2.49%	-1.90%	+79.25%
Maximum Error					0.00%	-47.38%	-11.02%	-5.09%	+282.35%

Table 5: Noises on a noisy victim from different algorithms for aligned RLC bus structure

models, both with timing window constraints. We use the WCN algorithm from [1] for the RC model. We use the aligned bus structure in Figure 1 with $0.6\mu m$ wire spacing and routing directions of “01010100” (“0” and “1” represent two opposite directions respectively). The centers of the aggressor switching windows are fixed and decided such that their maximal noise peaks under RLC model are perfectly aligned. In the experiments, we change the position of the victim sampling window and compute the correspondent WCN. In Figure 6, we show examples with a fixed driver size of 120× but with different timing window sizes. From (a) to (c) in the figure, the sizes of the aggressor switching windows are 20ps, 30ps and 50ps respectively and the size of victim sampling window is 10ps, 15ps and 25ps respectively. The X-axis is the position of the victim sampling window center and the original point is the position that has the maximum WCN without the sampling window constraint. Clearly, the WCN under RLC model is much larger than that under RC model when there is no sampling window constraint. When there is a sampling window constraint, the WCN varies with respect to the position of the sampling window, and the RLC model still gives larger WCN than RC model in most cases.

However, in the circled parts of Figure 6(a) and 6(b), RC model produces larger WCN than RLC model does. Because of resonance in the noise waveform, the noise peaks are normally narrower and sharper under RLC model than under RC model, and thus the WCN of RLC model may be smaller than that of RC model when the sampling window is between two adjacent noise peaks in RLC model. When we increase the size of the timing windows as shown in Figure 6(b) and 6(c), the width of the peak increases and the adjacent peaks from RLC model most likely overlap with each other. We can see that the overestimation of RC model gradually vanishes and the region of the overestimation moves away from the origin when the timing window sizes increase. When the sizes of timing windows are large enough, the overestimation of RC model disappears (see figure 6(c)). Overall, RC interconnect model underestimates the WCN in most cases, but it does overestimate the WCN when the timing window sizes are small enough. Whether RC model underestimates or overestimates the WCN depends on the detailed settings of the interconnects and the sizes and locations of the timing windows. We plan to develop efficient metrics to determine the conditions of RC model overestimating WCN in our future work. The underestimation under RC model leads to underdesign which causes circuit failures due to crosstalk violations, and the overestimation under RC model leads to overdesign which causes larger cost. For accurately analyzing the WCN problem of GHz+ interconnects, the RLC model is necessary.

5. CONCLUSION

Previous work on the interconnect worst case crosstalk noise (WCN) under RLC model consider neither timing window constraints nor signal routing direction. In this work, we have presented the first study on WCN under RLC model with consideration of both aggressor switching windows and the victim sampling window. We have shown that the signal routing direction impacts WCN significantly under RLC model but not under RC model. We proposed a new $SS{+}AS$ algorithm to consider the timing constraints. The $SS+AS$ algorithm has linear complexity, and approximates WCN very well with an average underestimation of 2% and an maximum underestimation of 5%. Experiments show that without considering timing windows, the RLC noise can be overestimated significantly. Thus, we must consider timing window constraints in WCN analysis. We further show that RC model underestimates WCN in most cases with timing constraints, but it *does* overestimate WCN when both the aggressor switching window and victim sampling window are small enough. We plan to develop effective matrices determining when the accurate RLC noise model is needed and when more efficient RC noise models can be applied without jeopardizing signal integrity in the future study.

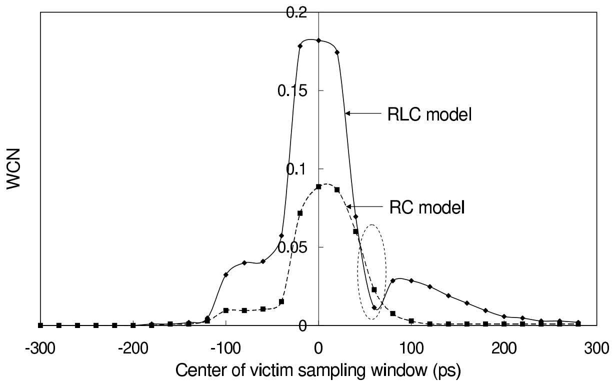
6. REFERENCES

[1] L. H. Chen and M. Marek-Sadowska, “Aggressor alignment for worst-case crosstalk noise,” *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 20, pp. 612 – 621, May 2001.

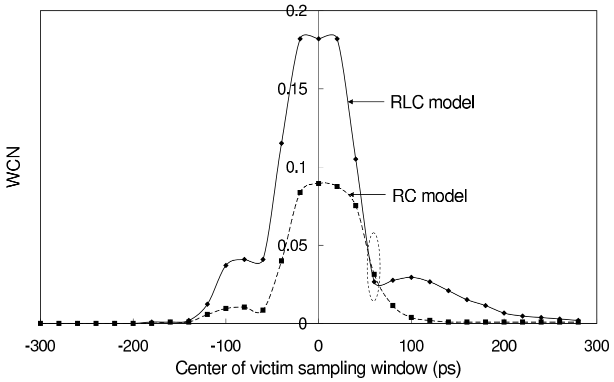
[2] L. He and K. M. Lepak, “Simultaneous shielding insertion and net ordering for capacitive and inductive coupling minimization,” in *Proc. Int. Symp. on Physical Design*, 2000.

[3] C. K. Cheng, L. Lillis, S. Lin, and N. Chang, *Interconnect analysis and synthesis.* John Wiley & Sons, Inc., 1999.

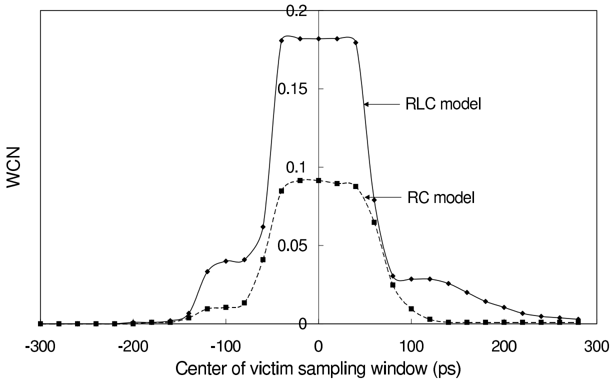
[4] J. Chen and L. He, “Determination ofworst-case crosstalk noise for non-switching victims in ghz+ interconnects,” in *Proc. Asia South Pacific Design Automation Conf.*, 2003.



(a) The aggressor switching window sizes are 20ps and the victim sampling window size is 10 ps.



(b) The aggressor switching window sizes are 30ps and the victim sampling window size is 15 ps.



(c) The aggressor switching window sizes are 50ps and the victim sampling window size is 25 ps.

Figure 6: WCN changes with the position of victim sampling window under RLC and RC models. Driver size is 120×.

[5] K. Tseng and V. Kariat, "Static noise analysis with noise windows," in *Proc. Design Automation Conf*, 2003.

[6] http://www-device.EECS.Berkeley.EDU/ ptm/.

[7] J. K. Ousterhout, "Switch-level delay models for digital MOS VLSI," in *Proc. Design Automation Conf*, pp. 542–548, 1984.

[8] F. Dartu, N. Menezes, J. Qian, and L. T. Pillage, "A gate-delay model for high-speed CMOS circuits," in *Proc. Design Automation Conf*, pp. 576–580, 1994.

[9] P. D. Gross, R. Arunachalam, K. Rajagopal, and L. T. Pileggi, "Determination of worst-case aggressor alignment for delay calculation," in *Proc. Int. Conf. on Computer Aided Design*, 1998.

[10] M. Xu and L. He, "An efficient model for frequency-dependent on-chip inductance," in *Great Lakes Symposium on VLSI*, 2001.

[11] T. Lin and L. Pileggi, "On the efficacy of simplified 2d on-chip inductance models," in *Proc. Design Automation Conf*, 2002.