

# TACO: Timing Analysis With COupling\*

Ravishankar Arunachalam, Karthik Rajagopal<sup>1</sup> and Lawrence T. Pileggi

Department of Electrical and Computer Engineering,  
Carnegie Mellon University, Pittsburgh, PA 15213  
{ravia,pileggi}@ece.cmu.edu

<sup>1</sup>Microprocessor Products Group  
Intel Corporation, Santa Clara, CA 95052  
karthik.rajagopal@intel.com

**Abstract:** The impact of coupling capacitance on delay is usually estimated by scaling the coupling capacitances (often by a factor of 2) and modeling them as grounded. This simple approach has been shown to be overly pessimistic in some cases, while somewhat optimistic in others. This paper introduces TACO, a timing analysis methodology that produces tight bounds on worst- and best-case timing for circuits with dominant coupling capacitance. The methodology utilizes a coupled Ceff gate model for capturing the provably worst- and best-case delays as a function of the timing-window inputs to the gates.

## 1. Introduction and Motivation

With aggressive scaling and routing congestion for deep submicron designs, coupling capacitance to neighboring nets has been found to be as high as 70-80% of the total capacitance. This implies that coupled nets' activity will significantly impact performance parameters such as delay and noise. Some research progress has been made recently for the problem of noise in circuits[1][3][5]; But the effect of crosstalk on timing is not well studied, and can be a potential source of problems even when the noise doesn't cause circuit failure. A common technique used to compute worst-case delay due to crosstalk is to scale the coupling capacitances by a factor of 2 and model them as grounded. While being overly pessimistic in some cases, this 2x approach has also been shown to underestimate the actual worst-case delay in others[2], which is unacceptable. As our examples will show, the exact scaling factor is design and layout dependent, and is impossible to predict *a priori*.

In this paper we present TACO, a crosstalk-centric timing analysis approach capable of handling coupling effects. We use the model presented in [6] to determine worst-case delay due to switching aggressors. What remains is to determine if the aggressors can indeed switch in conjunction with the victim to increase the victim delay. The timing characteristics of the aggressor can be used for this purpose, but the aggressor's timing window could depend on the victim's output, leading to a classical chicken and egg problem. We overcome this problem by starting with a worst case assumption for the switching windows. Under this assumption, the delay engine produces a pessimistic delay bound if the

\* This work was supported in part by the Semiconductor Research Corporation under contract 98-DC-068, a grant from Intel Corporation and a fellowship from IBM.

timing windows of the aggressors are not known *a priori*. The pessimism incurred in this process is reduced iteratively by using timing windows computed earlier, and including only aggressors which are actually capable of affecting the victim delay. Importantly, we prove convergence for this iterative process. This approach was implemented in the commercial timing analyzer Einstimer from IBM, and results were obtained for real industry designs with extracted coupling. The results will show that while the scaling approach can cause a 40% error in stage delay, the error incurred in path delay is not proportionate, and is unpredictable.

The rest of the paper is organized as follows. Section 2 gives a brief background on static timing analysis and the coupled delay model that we use. We formulate the coupled timing analysis problem in Section 3, and describe our timing analysis with coupling (TACO) approach. Proof of convergence of the methodology is described in Section 4. Section 5 gives some implementation details and results obtained using our approach. Conclusions and future work comprise Section 6.

## 2. Background

### 2.1. Static Timing Analysis

Static timing analysis (STA) is the most popular technique used to verify the timing behavior of integrated circuits[8]. STA works by analyzing the circuit for the earliest and latest possible signal arrival times on each logic path/node, regardless of what is happening on other paths. The arrival times combined with the signal transition time, also referred to as the "slew" of a signal, are expressed in the form of "windows", as shown in Figure 1. Comparing the

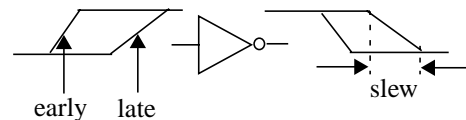


FIGURE 1: Typical arrival time windows for a gate

arrival times at a particular node in the logic with the required arrival time, we get the slack at that node in the logic. By assuming that the earliest and latest signals can be propagated through a particular gate, STA can be completely independent of the logic function of the gate, and can be prone to the false path problem. In this paper, we only focus on computing the topological best and worst case delays in the presence of coupling.

### 2.2. Coupled Ceff Delay Model

We use the coupled-gate effective capacitance (Ceff) model presented in [6] as the core delay engine. For a system of N coupled lines, each gate is modeled by its Thevenin equivalent and Ceff iterations are performed for each gate in a decoupled manner. The interconnect is modeled by a coupled N-port, and AWE[4] or some similar scheme can be used to obtain a reduced-order macromodel for the N-port. The waveforms obtained from the model agree excellently with those obtained from HSPICE[6].

For computing best and worst case delays due to aggressor switching, it is necessary to align the aggressors with respect to the

victim. This is performed as follows: First the noise waveform is computed, and the peak value of the noise  $V_p$  is found. This waveform is then aligned such that the peak occurs when the victim value is  $V_{dd}/2 - V_p$  (for a falling victim), so as to just cause the victim to reach  $V_{dd}/2$  again. The waveforms can be superposed *within an iteration* due to the linearized nature of the gate models used. This alignment process is explained in greater detail in [6].

### 3. TACO algorithm

Before we discuss the details of our timing analysis with coupling (TACO) approach, we define the terminology that will be used throughout the remainder of the paper.

#### Definitions

1. **Net:** A set of nodes that are resistively connected. A net has one driver node, one or more fanout nodes and may have a number of intermediate nodes that are part of the interconnect.
2. **Gate:** A logic unit or cell whose Thevenin model has been pre-characterized as a function of the input transition time and output load capacitance, for each input. Each gate has an output net whose fanout nodes are also referred to as the fanout nodes of the gate.
3. **Input-output delay:** Defined with respect to a particular input node and fanout node of a gate, it is the time difference between the 50% point of the signal at the input, and the 50% point of the signal at the fanout node.
4. **Best/worst case delay (BCD/WCD):** The input-output delay computed under the condition that the time at which the output node reaches the 50% point is minimized/maximized.
5. **Early/Late arrival times (EAT/LAT):** The EATs and LATs for fanout node  $j$  of a gate with  $n$  inputs are defined as follows:

$$EAT_j = \min_{i=1,2,\dots,n} (EAT_{input\ i} + BCD_{input\ i \rightarrow fanout\ j})$$

$$LAT_j = \max_{i=1,2,\dots,n} (LAT_{input\ i} + WCD_{input\ i \rightarrow fanout\ j})$$

6. **Aggressor net:** A net that has “significant” coupling capacitance to the victim net so as to be able to influence the delay of the victim gate. The criterion for coupling capacitance being “significant” is discussed in Section 3.3.
7. **Aligned aggressor:** An aggressor is said to be an aligned aggressor with respect to a particular victim, if the switching window of the aggressor is such that it can contribute to the BCD/WCD of the victim.

The problem of performing STA on a coupled circuit can be formulated as follows: Given the early and late arrival times at the primary inputs, propagate these arrival times to the primary outputs taking into account the influence of aggressor gates in computing best and worst case delay.

The contribution of aggressor gates to the victim delay can be included using the Ceff model outlined in Section 2, but the relative switching windows of the aggressor nets are not known. In order to obtain the absolute best and worst case delays, we assume initially that all aggressors act as aligned aggressors. To accomplish this, the EAT for all nodes is initially set to zero, and the LAT is set to infinity (or the latest possible switching time, which is the estimated clock period). We then perform one run of STA on the circuit, and compute EATs and LATs at all nodes in the circuit.

Since we assume an infinite initial switching window for all nodes in the circuit, all aggressors are taken into account for their effect on delay if their windows have not been computed. This gives significant scope for pessimism in the analysis. Then, in order to overcome this pessimism, we perform a second run of STA on the

circuit. During this second run, we find that some aggressors which we had assumed were aligned are indeed temporally isolated from the victim. Hence the timing windows of such nodes would constrict and we would obtain tighter bounds for them. This reduction in windows could then propagate along various paths. We can theoretically repeat this process to further tighten the windows, but in practice we find that beyond a second iteration, the windows do not shrink significantly. This process is represented as a flowchart in Figure 2.

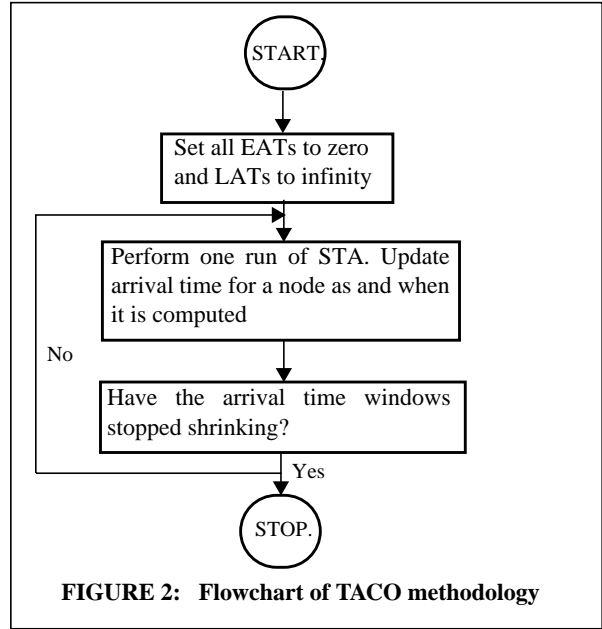


FIGURE 2: Flowchart of TACO methodology

### 3.1. Criterion for aggressor alignment

An aggressor is said to be aligned if it is capable of contributing to the delay of the victim gate. This implies that to have an effect on delay, the window of the noise waveform at the victim fanout should overlap with the window of the victim output without noise. Depending on the exactness of the analysis, a suitable condition can be used to determine if the aggressor can be aligned. The conditions we used in our analysis were:

The aggressor cannot be aligned with the victim if

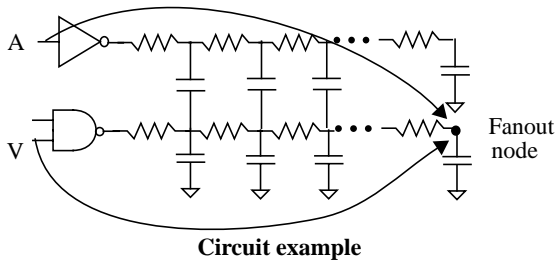
- i) The latest noise peak time occurs before the EAT at the victim driver port.
- ii) The LAT at the victim fanout occurs before the EAT at the aggressor driver port.

It should be noted that these conditions produce a pessimistic bound on the delay.

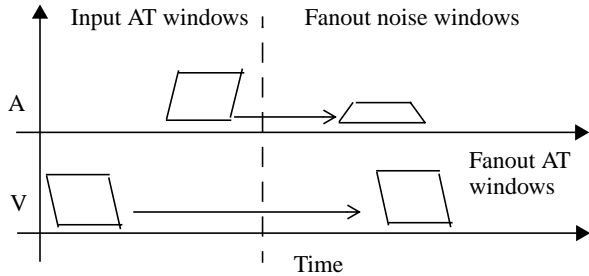
This is illustrated in Figure 3. ‘A’ refers to the aggressor and ‘V’ refers to the victim. The critical node where potential overlap should be checked is the fanout node and not the driver node. As shown in the figure, even though the aggressor switches later than the victim, it can create noise that affects the victim. This effect cannot be observed by using switching windows at either the input or the driver nodes of the gates.

### 3.2. Slew model

The slew of a signal is indicative of how fast the signal transitions from a 0 to a 1 or vice-versa. Traditionally, the slew value has been “tied” to the arrival time value at a particular node. This means that while the EAT and LAT of the window represent the absolute earliest and latest possible times at which the node can undergo a transition, the early slew and the late slew do not necessarily represent the fastest and slowest possible rate of transitions.



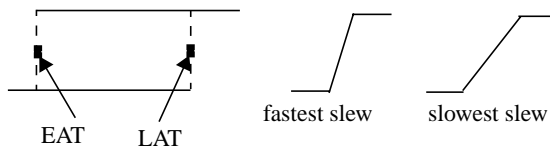
**Circuit example**



**FIGURE 3: Criterion for aggressor alignment**

The difference is especially critical for coupled circuits, where the load for computing EAT and LAT can be significantly different.

We propose a new arrival time window model to handle the slew effects. We separate the slew value from the arrival time value in defining the timing window, and maintain a “slew range” for each node. The slew range represents the absolute fastest and slowest possible transitions at the particular node. The fastest slew is used for computing EAT, and the slowest slew is used for computing LAT. The noise value, however, is always computed using the fastest slew at the aggressor. The new models for the arrival time window and slew are shown in Figure 4. As will be seen later, this slew model facilitates the proof of convergence for our methodology.



**FIGURE 4: New model for arrival time windows**

### 3.3. Pruning of coupled nets

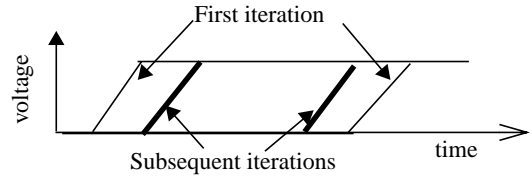
If we consider a long global net from a typical industrial circuit, we find that it is coupled to hundreds of other nets. Most significant coupling capacitance is to same layer metal lines of which there are only a few. A large number of much smaller coupling capacitances are to nets on other metal layers.

To avoid complexity in the coupled Ceff iterations, we limit the number of aggressors in our analysis to 3 to generate the results below. It is important to note that there is not much accuracy lost in analyzing only the “major” coupled nets. This is because the coupling capacitance as a percentage of the total capacitance for all the other nets is typically very small. Even so, these capacitances are not ignored in the analysis and are modeled as grounded.

## 4. Convergence

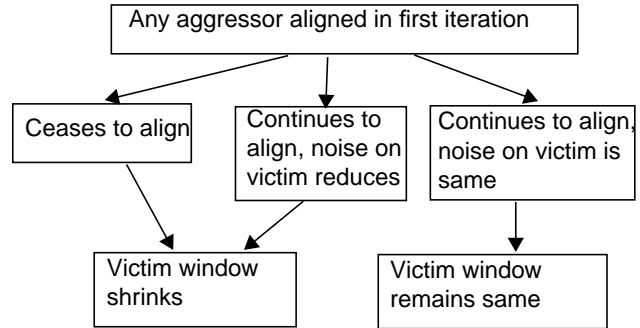
The proof of convergence for our approach hinges primarily upon starting with the worst-case assumption. The timing window we obtain during the first pass is a bound on the actual timing window, for each node in the circuit. Specifically, the EAT computed during the first iteration is the absolute earliest possible for that

node, and the LAT is the absolute latest possible. This is illustrated in Figure 5.



**FIGURE 5: Shrinking arrival time windows**

Part of the proof for this is summarized in Figure 6. If an initially aligned aggressor ceases to be aligned in a later iteration, the victim window shrinks because the WCD(BCD) will decrease(increase) without the noise contribution from the aggressor. If it continues to



**FIGURE 6: Convergence of TACO algorithm**

align and the noise contribution to delay reduces, the victim window still shrinks. There is an assumption here that the noise due to an aggressor cannot increase with subsequent iterations. Our choice for the slew model ensures this. Since we use the fastest slew possible at the aggressor during the first iteration, it is guaranteed that the noise value does not increase with subsequent iterations.

Now, consider an aggressor that is not aligned in the first iteration. What we have shown in Figure 6 also holds good for any aggressor’s timing window (since it will be treated as a victim at some point in the analysis). In conjunction with the alignment criterion defined in Section 3.1, and the fact that we start with the worst-case assumption, this implies that an aggressor that is not aligned in the first iteration can never do so in subsequent iterations. Hence the timing window of the victim corresponding to this aggressor remains unchanged.

It should be noted that the shrinking of victim timing windows cannot go on forever. Once the alignment status of each aggressor is established as either “aligned” or “unaligned”, all timing windows have converged. In practice however, we find that this status does not change after the first iteration.

## 5. Results

### 5.1. Prototype TACO implementation

We implemented a prototype version of TACO in 5500 lines of C++ code and interfaced it with RICE libraries to do the model order reduction. We ran the tool on two 0.35μ industrial examples from a microprocessor. Transistor level circuit descriptions were converted to gate level netlists using the pattern recognition tool, tranalyze[7].

Table 1 shows results obtained by comparing the delays obtained from TACO to those obtained by just modeling the coupling caps to ground. The maximum increase in stage delay is approximately 40% for a strongly coupled gate. The change in path delays is indicative of the change in slack through the critical path in the circuit

ckt	No. of Gates	Max. coupling cap ratio	Max% error in stage delay	Max% error in path delay
1	57	17%	15.1	9.2
2	126	70%	39.5	12.5

**TABLE 1: TACO vs. coupling caps to ground**

due to crosstalk. This information can be used by the designer to resize and/or resynthesize those parts of circuit affected by crosstalk.

The increase in delays due to coupling shown above is after one pass of block based timing analysis. TACO performs a second run of STA to reduce pessimism in the timing windows. Table 2 shows the percentage reduction in the stage delays and path delays after the second run. This change is primarily due to the reduction in the

Circuit	Max.% change in stage delay	Max.% change in path delays	% of aligned aggressors	
			Iteration 1	Iteration 2
1	26.1	6.6	50.7	38.9
2	6.0	5.1	10.7	0.0

**TABLE 2: Effect of a second run of STA**

number of aligned aggressors. The pessimistic windows obtained in the first run are crucial in determining exact alignment in the second run. If we had instead performed uncoupled timing analysis, then it is possible that we would have missed some aggressors in the alignment process.

## 5.2. Integration in Einstimer

The TACO approach was also integrated into Einstimer, a commercial timing analysis tool at IBM. It was run on a portion of a 0.5 $\mu$  design with extracted coupling. The circuit was comprised of 3210 gates and a maximum coupling cap ratio of 29%. The delays and arrival times obtained using the TACO methodology were compared with the simple scaling approach. Experiments were performed with scaling factors of 1, 1.5 and 2 for the coupling capacitors, and the results obtained are tabulated in Table 3. A scal-

Scaling Factor	Max difference in stage delay	Max difference in path delay
1	+33.3%	+11.8%
1.5	-34.9%	+8.2%
2	-40.4%	-4.4%

**TABLE 3: Comparison of TACO approach with scaling approach**

ing factor of 1 just means that the coupling caps were modeled as grounded.

A positive difference in the delay indicates that the worst-case delay(WCD) computed by TACO was more than that from the scaling approach, and vice-versa. A unity scaling factor is guaranteed to underestimate the WCD, as is reflected in the results. But more importantly, TACO is able to come up with a much tighter bound on delay compared to the scaling approach with a factor of 2. Since TACO starts with a worst-case assumption for aggressor alignment, the true WCD(BCD) of the circuit cannot be greater(lesser) than the one obtained from TACO. This implies that TACO is able to reduce

a significant amount of pessimism incurred due to the 2x approach. The results also indicate that while a scaling factor of 1.5 can overestimate the stage delay, it can underestimate the path delay. The impact of coupling on path delay depends on the extent of coupling along the path, the strengths of various coupled nets etc. Hence it is impossible to use a single scaling factor for an entire design.

The run times of Einstimer on this example are as follows:

CPU time for plain scaling approach: 25.7 units

CPU time with the TACO approach: 30.3 units

It can be seen that TACO is able to obtain better bounds on the timing windows with only an 18% increase in the run time. For this example, the timing windows did not change significantly during a second iteration, and hence results are shown only for one iteration of TACO. The number of iterations required, however, is layout and design dependent.

## 6. Conclusions and Future Work

We have presented a novel crosstalk oriented timing analysis algorithm that accurately analyzes the timing of coupled logic stages of the DSM era. A strategy to iteratively reduce the pessimism of the timing windows at primary outputs was outlined. The error incurred in using the arbitrary scaling approach was highlighted and results presented on real industrial examples. These results show that a paradigm shift is required in the current static timing analysis methods.

Some future work planned is to include the effect of functional information to eliminate false coupling interactions. This is to further reduce the pessimism in the timing windows.

## Acknowledgments

The authors would like to thank Alex Suess and Khalid Rahmat from IBM EDA, Fishkill for their help and support in the implementation of TACO in Einstimer and for providing test-case examples.

## REFERENCES

- [1]Ashok Vittal and Malgorzata Marek-Sadowska, "Crosstalk reduction for VLSI," IEEE Trans. on Computer-Aided Design, vol. 16, pp. 290-298, March 1997.
- [2]F.Dartu and L.T.Pileggi, "Calculating Worst-Case Gate Delays Due to Dominant Capacitance Coupling," 34th ACM/IEEE Design Automation Conference, pp.576-580, 1997.
- [3]K.L.Shepard et. al., "Global Harmony: Coupled Noise Analysis for Full-Chip RC Interconnect Networks," Proc. of the International Conference on Computer-Aided Design, pp. 139-146, 1997.
- [4]L.T. Pillage and R.A. Rohrer, "Asymptotic Waveform Evaluation Timing Analysis," IEEE Trans. Computer-Aided Design, pp. 352-366, April 1990 4
- [5]O.S.Nakagawa, et al., "Closed form modeling of on-chip crosstalk noise for deep submicron ULSI interconnect", Hewlett Packard Journal, Aug 1998.
- [6]P.Gross, R.Arunachalam, K.Rajagopal and L.T.Pileggi, "Determination of Worst Case Aggressor Alignment for Delay Calculation", Proc. of the International Conference on Computer-Aided Design, November 1998.
- [7]R.E.Bryant,"Extraction of Gate Level Models from Transistor Circuits by Four-Valued Symbolic Analysis", Proc. of the International Conference on Computer Aided Design, 1991.
- [8]R.Hitchcock,"Timing Verification and the Timing Analysis Program", Proc. of the Design Automation Conference, 1982.