

# Worst-Case Circuit Delay Taking into Account Power Supply Variations \*

Dionysios Kouroussis  
 Department of ECE  
 University of Toronto  
 Toronto, Ontario, Canada  
 diony@eecg.utoronto.ca

Rubil Ahmadi  
 Department of ECE  
 University of Toronto  
 Toronto, Ontario, Canada  
 rubil@eecg.utoronto.ca

Farid N. Najm  
 Department of ECE  
 University of Toronto  
 Toronto, Ontario, Canada  
 f.najm@utoronto.ca

## ABSTRACT

Current Static Timing Analysis (STA) techniques allow one to verify the timing of a circuit at different process corners which only consider cases where all the supplies are low or high. This analysis may not give the true maximum delay of a circuit because it neglects the possible mismatch between drivers and loads. We propose a new approach for timing analysis in which we first identify the critical path(s) of a circuit using a power-supply-aware timing model. Given these critical paths, we then take into account how the power nodes of the gates on the critical path are connected to the power grid, and re-analyze for the worst-case time delay. This re-analysis is posed as an optimization problem where the complete operation of the entire circuit is abstracted in terms of current constraints. We present our technique and report on the implementation results using benchmark circuits tied to a number of test-case power grids.

## Categories and Subject Descriptors

B.7 [Integrated Circuits]: Design Aids

## General Terms

Design, Algorithms, Verification

## Keywords

Static timing analysis, power grid, voltage fluctuations

## 1. INTRODUCTION

Timing of modern integrated circuits is becoming increasingly sensitive to supply voltage fluctuations. Circuit timing is highly effected by even the smallest drop in  $V_{dd}$ . Thus,

\*This research was supported in part by Micronet, with funding from ATI and from Altera, and by the SRC under contract 2003-TJ-1070.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

DAC'04, June 7–11, 2004, San Diego, California, USA.  
 Copyright 2004 ACM 1-58113-828-8/04/0006 ...\$5.00.

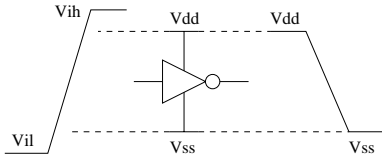
in the analysis and verification of high performance chip design, it is essential that timing analyzers take into account power supply variations. This is a formidable task due to the increasingly large size of power grids and the difficulty of obtaining all possible circuit behaviors that may be used to set up the right simulation in order to compute an accurate worst-case delay for a circuit.

Voltage fluctuations within on-chip power grids, are a result of many sources such as  $IR$ -drop,  $Ldi/dt$  drop, and resonance between the grid and the package. It is common, especially at frequencies below a GHz, that inductance is neglected and simulation of the power grid is focused on only  $IR$ -drop given the  $RC$  structure of the grid. Most simulation techniques for power grid analysis model the power grid as an  $RC$  network and use some form of circuit simulation to complete the simulation. Unfortunately, given the very large number of possible circuit behaviors, one needs to simulate the circuit (for the currents) and the grid (for the voltage drops) for a large number of clock cycles or vector sequences, which is impractical.

We are interested in the possibility of determining how the voltage fluctuations of the grid will affect circuit delay *without* having complete knowledge of the circuit currents. We will assume that only *incomplete* information of circuit currents is available in terms of current constraints [5]. These constraints essentially are the upper bounds on the currents.

Furthermore, we study the effect of variations of the grid voltages on the circuit timing, and develop a static timing analysis (STA) approach that takes these variations into account. We begin by assuming that the exact voltage drops are not known, but that the *ranges* of voltage drops are specified. As well, we first assume that the power supply nodes of the gates in the circuit are totally independent in order to identify the *worst-case* voltage configuration that causes a logic circuit to exhibit its absolute worst-case delay. Then, considering how the supplies of the gates on a critical path are connected to the power grid, we re-analyze the delay on that path, based on knowledge of the grid and the circuit current constraints. The delay time re-analysis is done by using the delay model developed in [1]. This approach has been formulated as a nonlinear programming problem, which we solve for the maximum delay subject to the current constraints.

Consider the diagram in Fig. 1, where an inverter is shown with its input and output waveforms. The power supply nodes of the inverter are considered, the reference  $V_{dd}$  and



**Figure 1: Modeling parameters**

$V_{ss}$ , and the input is assumed to rise from  $V_{il}$  to  $V_{ih}$ . The output of the inverter, as does the output of its fanout interconnect network, falls from  $V_{dd}$  to  $V_{ss}$ . It is instructive to consider what is a practical range of variations of the power supply values. In order for the circuit to function properly, the transistors must be able to turn off, which sets a limit on how large the supply variations may be. For one thing, we should have  $|V_{ss} - V_{il}| < V_{tn}$  and  $|V_{ih} - V_{dd}| < |V_{tp}|$ . In the worst-case, if we consider opposite variations for  $(V_{ss}, V_{il})$  and  $(V_{ih}, V_{dd})$ , then:

$$|\Delta V_{ss}| + |\Delta V_{il}| < V_{tn} \Rightarrow \text{roughly, } |\Delta V_{ss}| < V_{tn}/2 \quad (1)$$

$$|\Delta V_{dd}| + |\Delta V_{ih}| < |V_{tp}| \Rightarrow \text{roughly, } |\Delta V_{dd}| < |V_{tp}|/2 \quad (2)$$

Throughout this work, we have used  $0.13\mu\text{m}$  CMOS technology, with a nominal supply voltage of  $1.2V$ , and we assumed  $\pm 12.5\%$  variation around  $V_{dd}$  and  $V_{ss}$ . This is equivalent to  $0.15V$  drop or increase around the nominal power supply and ground respectively. Therefore,  $V_{ih}$  and  $V_{dd}$  can vary from  $1.05V$  to  $1.35V$ , and  $V_{il}$  and  $V_{ss}$  can vary from  $-0.15V$  to  $+0.15V$ .

Thus, in order for the analysis to be meaningful, one must first check whether the voltage drops resulting from the specified current constraints satisfy these voltage drop ranges. In our work, once we have found a critical path and determined the supply nodes to the gates on that path, we perform voltage optimization on the grid to find the worst-case voltage drop on those nodes as in [5]. If the maximum voltage drop on those nodes does not exceed the allowable ranges, we proceed to our timing re-analysis. If a voltage exceeds the range then the grid should be corrected and rechecked for maximum voltage drop.

## 2. METHODOLOGY DEFINED

As an overview, our proposed timing verification flow is as follows:

1. Abstract the entire behavior of circuit in terms of current constraints at the power taps of all the gates in the circuit.
2. Extract the critical path(s) of the circuit using specified upper/lower bound ranges of supply variations.
3. Verify that the voltage of the supply nodes of the critical path(s) are within bounds.
4. Solve for the maximum delay of path(s) given current constraints.

Our approach integrates some of the work presented in [1] and [5], leading to an approach where timing analysis is performed taking the power grid into account. In the following, after introducing the power grid model in section 3, we will summarize elements from [5] in section 4 and some of the concepts from [1] in section 5. This material is needed to describe our approach. We then discuss the new contribution of this paper starting with section 6.

## 3. POWER GRID MODELING

We consider an RC model of the power grid, where each branch of the grid is represented by a resistor and where there exists a capacitor from every grid node to ground. In addition, some grid nodes have ideal current sources (to ground) representing the current drawn by the circuit tied to the grid at that point, and some grid nodes have ideal voltage sources (to ground) representing the connections to the external voltage supply.

Let the power grid consist of  $n + p$  nodes, where nodes  $1, 2, \dots, n$  have no voltage sources attached, and nodes  $(n + 1), (n + 2), \dots, (n + p)$  are the nodes where the  $p$  voltage sources are connected. Let  $c_k$  be the capacitance from every node  $k$  to ground. Let  $i_k(t)$  be the current source connected to node  $k$ , where the direction of positive current is from the node to ground. We assume that  $i_k(t) \geq 0$  and that  $i_k(t)$  is defined for every node  $k = 1, \dots, n$  so that nodes with no current source attached have  $i_k(t) = 0, \forall t$ . Let  $\mathbf{i}(t)$  be the vector of all  $i_k(t)$  sources,  $k = 1, \dots, n$ . Let  $u_k(t)$  be the voltage at every node  $k$ ,  $k = 1, \dots, n$ , and let  $\mathbf{u}(t)$  be the vector of all  $u_k(t)$  signals,  $k = 1, \dots, n$ . Applying Modified Nodal Analysis (MNA) [7], leads to:

$$\mathbf{G}\mathbf{u}(t) + \mathbf{C}\dot{\mathbf{u}}(t) = -\mathbf{i}(t) + \mathbf{G}\mathbf{V}_{dd} \quad (3)$$

where  $\mathbf{G}$  is an  $n \times n$  conductance matrix,  $\mathbf{C}$  is an  $n \times n$  diagonal matrix of node capacitances, and  $\mathbf{V}_{dd}$  is a constant vector each entry of which is equal to  $V_{dd}$ . The matrix  $\mathbf{G}$  has several useful properties. It is symmetric positive definite [4] and can be shown to be an M-matrix [2] which means, among other things, that its inverse consists of only non-negative values.

Defining  $v_k(t) = V_{dd} - u_k(t)$  to be the voltage drop at node  $k$ , and let  $\mathbf{v}(t)$  be the vector of voltage drops, then the system equation can be written as:

$$\mathbf{G}\mathbf{v}(t) + \mathbf{C}\dot{\mathbf{v}}(t) = \mathbf{i}(t) \quad (4)$$

This is a *revised* system equation which one can solve directly for the voltage drop values. Notice that the circuit described by this equation consists of the original power grid, but with all the voltage sources set to zero (short-circuit) and all the current source directions reversed. In the following, we will mainly be concerned with this *modified power grid* and the revised system of equations (4).

Based on the monotonicity property of the power grid, [8, 6, 5], we can now make a couple of statements that will be useful below. Let  $I_k$  be an upper bound on  $i_k(t)$  over the time period of interest, say  $0 \leq t \leq \infty$ . Let  $I_1, I_2, \dots, I_n$  form a  $n \times 1$  vector  $\mathbf{I}$  and let  $\mathbf{V}$  be the solution of the system when the DC currents  $\mathbf{I}$  are applied as inputs, which may be found by solving the DC system:

$$\mathbf{G}\mathbf{V} = \mathbf{I} \quad (5)$$

Then, from the monotonicity property, it is clear that  $\mathbf{i}(t) \leq \mathbf{I}, \forall t \geq 0$  leads to  $\mathbf{v}(t) \leq \mathbf{V}, \forall t \geq 0$ . Finally, another related result is that, considering the DC system (5), if  $\mathbf{I}^* \geq \mathbf{I}$ , then  $\mathbf{V}^* \geq \mathbf{V}$ .

## 4. PEAK CURRENT CONSTRAINTS

In order to abstract the behavior of the entire chip we will use the two related notions of an incomplete current specification, referred to as *current constraints: local constraints*, and *global constraints*.

## 4.1 Local Constraints

A local constraint relates to a single current source. For instance, one may specify that current  $i_k(t)$  never exceeds a certain fixed level  $I_{L,k}$ , i.e.,  $i_k(t) \leq I_{L,k}, \forall t \geq 0$ . This upper bound may be simply known from prior simulation, if the cell or block is already available, or it may be a best-guess based on the area of the cell or block and on perhaps the *power density* of the design (total power divided by total area). If further information is available on the circuit behavior over time, then the user may be able to specify an upper bound *waveform*, as a time function, so that  $i_k(t) \leq i_{L,k}(t), \forall t \geq 0$ . We assume that every current source tied to the power grid has an upper bound associated with it. If a grid node does not have a current source attached to it, i.e.,  $i_k(t) = 0, \forall t \geq 0$ , then we specify a fixed zero-current upper bound for that node,  $I_{L,k} = 0$ . In this way, we have a local constraint associated with every node of the power grid. We express these constraints in vector form as:

$$\mathbf{0} \leq \mathbf{i}(t) \leq \mathbf{I}_L, \forall t \geq 0 \quad \text{or} \quad \mathbf{0} \leq \mathbf{i}(t) \leq \mathbf{i}_L(t), \forall t \geq 0 \quad (6)$$

## 4.2 Global Constraints

It is useful to express constraints related to all current sources or to sub-groups of current sources. For instance, if the total power dissipation of the chip is known, even approximately, then one may say that the sum of all the current sources is no more than a certain upper bound. In general, a global constraint corresponds to the case when the sum of the currents for a group of current sources is specified to have an upper bound. The upper bound for example, corresponding to the  $j$ th global constraint, may be a fixed bound  $I_{G,j}$ , or a waveform bound  $i_{G,j}(t)$ . If  $m$  is the number of available global constraints, then we express all the global constraints in matrix form as:

$$\mathbf{U}\mathbf{i}(t) \leq \mathbf{I}_G \quad \text{or} \quad \mathbf{U}\mathbf{i}(t) \leq \mathbf{i}_G(t) \quad (7)$$

where  $\mathbf{U}$  is a  $m \times n$  matrix that contains only 0s and 1s.

## 4.3 Combining Constraints

The local and global constraints can be combined into a single matrix inequality, as follows:

$$\mathbf{L}\mathbf{i}(t) \leq \mathbf{I}_m \quad \text{or} \quad \mathbf{L}\mathbf{i}(t) \leq \mathbf{i}_m(t), \\ \text{with } \mathbf{i}(t) \geq \mathbf{0}, \quad \forall t \geq 0 \quad (8)$$

where  $\mathbf{L}$  is an  $(n+m) \times n$  matrix of 0s and 1s, whose first  $n$  rows form an identity matrix (1s on the diagonal and 0s everywhere else) and whose remaining  $m$  rows correspond to the matrix  $\mathbf{U}$ , and where  $\mathbf{I}_m$  and  $\mathbf{i}_m(t)$  are  $(n+m) \times 1$  vectors.

## 4.4 Voltage Formulation

By making use of the relationship  $\mathbf{I} = \mathbf{G}\mathbf{V}$ , we can express the DC constraints in terms of voltages:

$$\mathbf{L}\mathbf{G}\mathbf{V} \leq \mathbf{I}_m, \quad \mathbf{V} \geq \mathbf{0} \quad (9)$$

## 5. TIME DELAY MODELING

In order to develop a timing analysis approach in presence of power supply and ground voltage fluctuations, one needs to first develop a delay model for logic cells that is dependent on these voltages. In this section, we will first define delay in a variable voltage environment and then introduce our delay models.

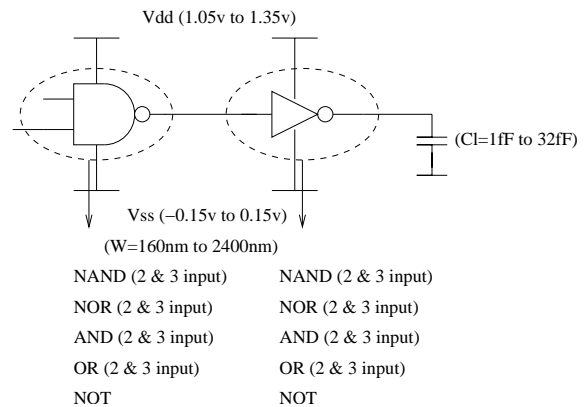


Figure 2: Possible gates and parameters combination.

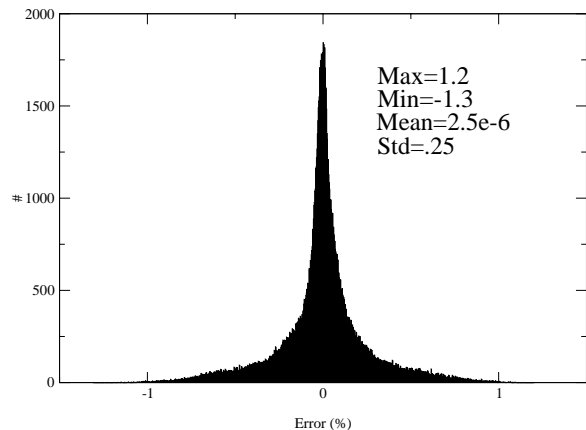


Figure 3: Modeling error.

## 5.1 Gate delay model

Signal delay needs careful attention when the supplies are potentially different between the driver and the load. Consider the typical timing waveforms in Fig. 1. The **gate delay** is defined as  $t_{d1} = t_2 - t_1$ , where  $t_1$  is the time at which the input signal reaches  $(V_{ih} + V_{il})/2$  and  $t_2$  is the time at which the output reaches  $(V_{dd} + V_{ss})/2$ . Based on this gate delay definition, the gate delay model will be introduced.

The gate delay depends on the traditional parameters of input signal slope and output load. In addition, in this work, we model the dependence of gate delay on the four supply voltages defined above, in Fig. 1. Thus, six parameters are considered as part of the gate delay model: the input high signal level ( $V_{ih}$ ), the input low signal level ( $V_{il}$ ), the gate's power supply ( $V_{dd}$ ), the gate's ground ( $V_{ss}$ ), the input slope ( $S_{in}$ ), and the gate's output load ( $C_l$ ). The **input slope** is defined as the slope ( $dV/dt$ ) of the input waveform at the time when it crosses  $(V_{il} + V_{ih})/2$ .

It is instructive to consider how variable the cell delays are, and how strong is their sensitivity to the supply voltages. To this end, we have built a library of cells, containing 2-input and 3-input input NAND, NOR, AND, OR and NOT gates. In our experiments, the load, transistor widths, and four voltage levels of the gates were varied across their valid ranges. Transistor width was allowed to vary from 160nm (the minimum size for 0.13 $\mu$ m technology) to 2400nm, and the loads from 1fF to 32.5 fF (as a comparison, the input capacitance of a minimum size gate for this tech-

nology is near 1fF). Furthermore, different combinations of consecutive gates were tested. Fig. 2 shows all possible gate type combinations along with valid parameters ranges. Normalized delay of gates in our library shows that the delay can change by up to 240% (2.4X) due to a 12.5% variation of the supply and ground around their nominal values.

Modern cell libraries represent the delay of cells using four 2-dimensional tables for each timing arc (a timing arc is an input-output node pair). In case of a falling output, one table gives the propagation delay and another gives the output slope. Another two tables correspond to the rising output case. Each table covers the range of valid input slope and output load values. Simple extension of this model to our case would require 6-dimensional tables, which would be impractical in terms of model size and cost of building the model. In order to simplify the model, we found that the delay dependence on each voltage is near-linear in the (narrow) range of valid voltages. However, to be more accurate, we have used a quadratic polynomial to represent the dependence of delay on each voltage, and made allowance for cross-product terms, by using a template expression for delay as follows:

$$t_d = \sum_k \alpha_k V_{ih}^{a_k} V_{il}^{b_k} V_{dd}^{c_k} V_{ss}^{d_k}$$

where  $\alpha_k \in \mathcal{R}$ , and  $a_k, b_k, c_k, d_k \in \{0, 1, 2\}$  (10)

The regression coefficients  $\alpha_k$  are found by using a standard Least Mean Square (LMS) regression method [9]. The regression is performed for each grid point in the [slope, load] table, so that each cell in the [slope, load] table contains the values for a number of coefficients  $\alpha_1, \alpha_2, \dots, \alpha_m$ . A similar model to this gives the output slope in terms of all four voltages and input slope and output load. We characterized (built the delay models for) all the cells in our library, then tested the error in delay between HSPICE and the library model. The results are shown in Fig. 3 for the propagation delay. It is seen that the model has very good accuracy. The output slope component of the model was also tested, and it shows an error of under 3%, which also is good.

## 5.2 STA Worst-case delay

Given a logic gate with variable supplies, it is important to look for the supply configuration that gives the worst-case gate delay. The situation is complicated, due to the number of variables involved, especially for complex CMOS gates. Analysis of empirical data shows that delay has a monotonous curve versus each of voltages; therefore, the sensitivity of the delay to a given voltage variable does not change sign as that voltage is varied across its whole range [1].

It was also shown in [1] that, for a multi-gate path composed of all inverting gates with independent supplies, the worst-case voltage setting for a falling input is given by the staggered arrangement:

$$\begin{pmatrix} H \\ H \end{pmatrix} \begin{pmatrix} L \\ L \end{pmatrix} \begin{pmatrix} H \\ H \end{pmatrix} \begin{pmatrix} L \\ L \end{pmatrix} \begin{pmatrix} H \\ H \end{pmatrix} \dots \quad (11)$$

and, for a rising input, it is given by the alternate staggered arrangement:

$$\begin{pmatrix} L \\ L \end{pmatrix} \begin{pmatrix} H \\ H \end{pmatrix} \begin{pmatrix} L \\ L \end{pmatrix} \begin{pmatrix} H \\ H \end{pmatrix} \begin{pmatrix} L \\ L \end{pmatrix} \dots \quad (12)$$

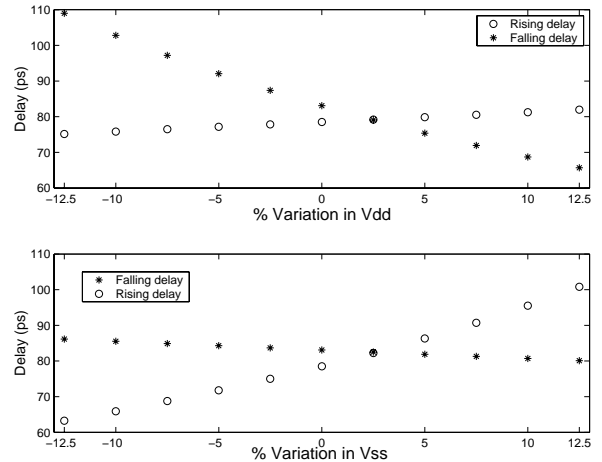


Figure 4: Delay vs. Voltage of a two gate system.

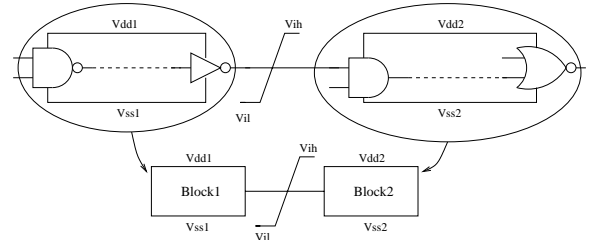


Figure 5: Consecutive blocks with independent supplies

## 6. WORST-CASE DELAY WITH GRID

Having found an accurate model of path delay in terms of supply voltage fluctuations, we are now interested in finding the maximum possible delay, given the current constraints which abstract the behavior of the entire circuit.

### 6.1 Time delay problem

Firstly the current constraints have to be checked to make sure that they do not lead to voltages on the grid that exceed the bounds of the worst case voltage allowed in the development of the gate timing model. This is done by first extracting the set of nodes that supply the path  $(v_i, v_j, \dots, v_m)$ . Then we may formulate the following sequential linear program:

$$\begin{aligned} \text{maximize :} & \quad v_i, v_j, \dots, v_m & (13) \\ \text{subject to :} & \quad \mathbf{LGV} \leq \mathbf{I} \\ & \quad \mathbf{V} \geq \mathbf{0} \end{aligned}$$

If any of the the voltages exceed the model bounds then the grid needs to be fixed and the supply nodes verified again. We solve this maximization by using an interior point method that maximizes the voltage drop of the connecting supply nodes of the critical path. Once the supply nodes have been verified, that they are within the bounds of the model, then we may proceed to our nonlinear optimization.

We may now pose the worst case time delay as a nonlinear programming (NLP) problem in the form of:

$$\begin{aligned} \text{maximize :} & \quad t_d & (14) \\ \text{subject to :} & \quad \mathbf{LGV} \leq \mathbf{I} \\ & \quad \mathbf{V} \geq \mathbf{0} \end{aligned}$$

where  $t_d$  is the delay of a specific critical path. An impor-

**Table 1: Experimental Results.**

Circuit	# of path gates	# of path supplies	STA delay (ns)	Powergrid name	# nodes	max. delay (ns)	Difference	NLP Computation time (sec.)
C1345	28	58	4.54	pgrid1.6k-5d	1320	4.13	9.9%	0.61
C1908	49	100	5.90	pgrid1.6k-5d	1320	5.31	11.1%	0.67
C2607	40	82	5.82	pgrid1.6k-5d	1320	5.64	3.2%	0.64
C3540	54	110	7.43	pgrid6.6k-5d	5832	6.96	6.7%	1.37
C432	48	98	7.22	pgrid6.6k-5d	5832	7.10	1.7%	1.54
C499	28	54	3.64	pgrid11k-10d	10073	3.52	2.6%	1.69
C5315	51	104	6.79	pgrid11k-10d	10073	6.28	8.1%	1.71
C7552	44	90	5.48	pgrid28.9k-5d	27055	4.84	13.2%	2.95
C880	27	56	3.92	pgrid28.9k-5d	27055	3.62	8.3%	2.83
S420	12	26	2.13	pgrid28.9k-5d	27055	1.81	11.9%	2.61
S510	12	26	2.14	pgrid45.4k-5d	43106	1.77	20.9%	3.87
C6288	124	250	21.47	pgrid45.4k-5d	43106	18.80	14.2%	4.00

tant property of this optimization problem is that, due to empirically observed sensitivity properties of delay of logic gates, it has no local maxima and only one global maximum. The reasoning for this is as follows.

We assume that a “fine” grid model is in use so that a node on the grid is connected to no more than one logic gate or cell. Thus, the voltage on a node on the grid would appear as a voltage variable in *only* two gates (*or blocks*) in the path, namely the gate it is directly connected to (we call this the driver gate), and the gate that is driven by that gate (we call this the load gate). We will show that, for any pair of gates in a driver-load arrangement, the delay of the combination of the two gates is *monotone* in any of the supply voltages that are connected to the driver gate.

Consider the delay of two connected inverters with independent supplies and grounds in presence of 12.5% voltage variation around the nominal values. Fig. 4 shows that the delay of two consecutive gates is always monotone due to variations of power supply and ground voltages and the sensitivity is positive or negative depending to the signal polarity. Delay sensitivities of all gates and gate combinations in our library (for example as in Fig. 5) have been checked and it is confirmed that the sensitivity of the delay to a given voltage variable does not change sign as that voltage is varied across the whole range.

As a result, the maximum delay always occurs at a corner of the voltage domain and, therefore, any local maximum of the optimization problem is also a global maximum.

The absence of local maxima eliminates the use of heuristics and instabilities associated with many NLP solvers which try to find global solutions. We use the SNOPT solver [3] for our optimization which can only find local solutions. We pre-generate the functions of all the gradients of our objective to improve efficiency. Since only our objective function is nonlinear, the problem is linearly constrained which tends to solve more easily than general nonlinear programs with nonlinear constraints. The solver uses a sparse sequential quadratic programming method using limited memory quasi-Newton approximations to the Hessian of the Lagrangian.

## 6.2 Transient Analysis

Even though our technique is based on using worst case DC current constraints, which we know from the monotonicity of the power grid will result in perhaps somewhat pessimistic estimates, it is possible to remove some of the pes-

simism by extending the analysis to the transient current domain, as follows. If current profiles are known for certain blocks or nodes, we may run our optimization on a clock cycle by cycle basis. Extracting the DC upper bound currents of all the current profiles within a clock period and running our analysis, will yield a delay time that would be better (*shorter*) than a delay time associated with general DC constraints but would still be a worst-case estimate for that clock cycle.

## 7. EXPERIMENTAL RESULTS

Our technique was implemented and tested on the IS-CAS85 and the combinational parts of the ISCAS89 benchmarks. Not having access to power grids from industrial designs and in order to test our approach under different conditions, we have opted to *generate* a number of grids ourselves. The grid generation process is automatic, and employs a random number generator, as well as user-specified technology and topology parameters. Starting with a square uniform grid of a given size, we proceed to randomly delete a user-specified percentage of nodes, thus rendering the grid structurally non-uniform. Typical geometric and physical grid characteristics (e.g. grid dimensions) as well as characteristics of the fabrication process (e.g. sheet resistance of a particular level of metallization) are given by the user, leading to an initial value of the conductance of every branch. When a node is deleted, the conductances of the remaining surrounding edges (branches) are increased by a random amount around a user-specified percentage of their initial values. The rationale behind this is to allow the non-uniform grid to be loaded with currents comparable to its uniform predecessor while exhibiting comparable *IR*-drops. The number of  $V_{dd}$  sites and current sources are supplied by the user; and are then distributed at random over the grid nodes. The supplies of the critical paths extracted from the ISCAS benchmarks were then randomly connected to our power grids. This random process of circuit to power grid connection was done in order to best emulate all the possible designs that could be encountered from critical paths within specific blocks to paths that may span the geometry of the entire chip.

Experiments were run on a 1 GHz Sun machine with 4 GB memory. Table 1 shows some of our results. A number of benchmark critical paths randomly connected to varying sized power grids, from 1000 nodes to 40,000 nodes, were simulated using our NLP approach. The worst case delay

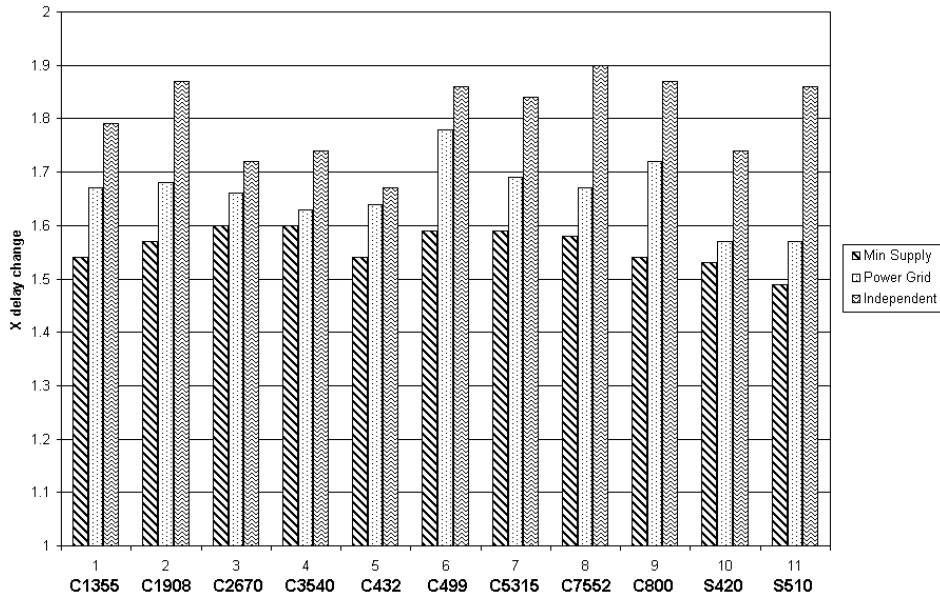


Figure 6: Comparison of Time Delay Calculation Techniques, Normalized to Delay of Nominal Supply

found under the influence of power grid is smaller than that found using STA analysis with independent supplies. The difference is seen to vary between 2% to 20%. The computation time for the solve of each delay time is seen to be a minimal 1 to 4 seconds. This reported time is only the time required to solve for the optimal solution of each critical path. It does not include the time required to perform a *preconditioning* on the linear component of the problem which may run in the order of 10 to 15 minutes for the larger sized grids. This computational time overhead, however, would only be required once for any power grid. Once it is computed it may be reused over and over for numerous critical paths within the chip or even with the same path should that path undergo design changes. Further, it was observed that our technique used about 100Mb of memory for the large grids, thus, that it may be easily applied to even larger grids. As for the computational cost times of up-front verification, that the node voltages do not violate the feasibility bounds, we borrowed the method in [5] but also improved it significantly by implementing an Interior Point Method with sparse matrix techniques. As a result, the time required for one check of a node voltage is in the order of half a minute for a 40,000 node grid. This check may be easily extended to larger grids as well.

Fig. 6 shows a comparison of calculated delay times of each of our test circuits using the minimum supply values, independent supplies and our power grid approach, normalized to the time delay of the nominal supply voltage. It is significant to observe that our approach finds a delay time that falls between the delay times of the minimum supply and the worst case independent method.

## 8. CONCLUSION

In today's integrated circuit designs, timing and its sensitivity to supply voltage fluctuations is a key concern. Analysis of voltage variations by simulation is a complicated task due to the requirement of stimulus (vectors, patterns, wave-

forms) in order to complete the simulation. It is hard in practice to obtain such stimulus and further, even if it were made available, the simulation would be required to run for prolonged periods of time with high computational cost overhead. We have proposed a method whereby we abstract circuit behavior in the form of user-supplied current constraints. By using a delay model that is expressed in the form of supply voltage variations of the path and running a nonlinear program, we may solve for the worst-case time delay.

## 9. REFERENCES

- [1] R. Ahmadi and F. N. Najm. Timing analysis in presence of power supply and ground voltage variations. In *International Conference on Computer Aided Design*, San Jose, 2003.
- [2] A. Berman and R. J. Plemmons. *Nonnegative Matrices in the Mathematical Science*. Academic Press, New York, 1996.
- [3] P. Gill, W. Murray, and M. Saunders. SNOPT: An SQP algorithm for large-scale constrained optimization. *SIAM J. Opt.*, 12:979–1006, 2002.
- [4] G. H. Golub and C. F. V. Loan. *Matrix Computations*. The Johns Hopkins University Press, 1996.
- [5] D. Kouroussis and F. N. Najm. A static pattern-independent approach for power grid voltage integrity verification. In *40<sup>th</sup> Design Automation Conference*, Anaheim, 2003.
- [6] H. Kriplani, F. N. Najm, and I. Hajj. Pattern independent maximum current estimation in power and ground buses of CMOS VLSI circuits: algorithms, signal correlations, and their resolution. *IEEE Transactions on Computer-Aided Design*, 14(8):998–1012, August 1995.
- [7] L. T. Pillage, R. A. Rohrer, and C. Visweswaraiiah. *Electronic Circuit and System Simulation Methods*. McGraw-Hill, Inc., New York, NY, 1995.
- [8] J. Rubenstein, P. Penfield, and M. A. Horowitz. Signal delay in RC tree networks. *IEEE Trans. on Computer-Aided Design*, 2(3):202–211, July 1983.
- [9] B. Widrow. *Adaptive signal processing*. Prentice-Hall, 1st edition, 1985.