Static Compaction of Delay Tests Considering Power Supply Noise

Jing Wang+, Xiang Lu*, Wangqi Qiu+, Ziding Yue*, Steve Fancler+, Weiping Shi*, D. M. H. Walker+

+Dept. of Computer Science
Texas A&M University
College Station TX 77843-3112
Tel: (979) 862-4387
Fax: (979) 847-8578
Email: walker@cs.tamu.edu

*Dept. of Electrical Engineering
Texas A&M University
College Station TX 77843-3124
Tel: (979) 458-0093
Fax: (979) 845-2630
Email: wshi@ee.tamu.edu

Abstract

Excessive power supply noise can lead to overkill during delay test. A static compaction algorithm is described in this paper that prevents such overkill. A power supply noise estimation tool has been built and integrated into the compaction process. Compaction results for KLPG delay tests for ISCAS89 circuits under different power grid environments are presented.

1. Introduction

Delay testing becomes increasingly critical as timing margins are reduced and clock rates increase. At-speed delay testing using the path delay fault model [1] has been used to detect small delay defects. However, rapid advances in deep submicron (DSM) technology make designs more sensitive to various noise sources [2], such as leakage noise, crosstalk and power supply noise. Excessive noise has a significant impact on the timing performance of DSM designs [3].

Power supply noise refers to the noise on the supply and ground network, which reduces device voltage levels and increases signal delay [2][3]. As frequency, gate density and current density increase [4], more simultaneous switching activity per area is expected. In addition, industrial data for sub-90 nm CMOS gates show that delay sensitivity rises as supply voltage decreases [4]. These trends lead to a larger power noise impact on delay.

Several techniques [5][6] have been proposed for estimating power supply noise. Different supply network and circuit models were used to achieve accuracy. Jiang [3] proposed a vector-independent approach using genetic algorithms. Liou [7] proposed an estimation method based on a statistical timing analysis framework.

Prior work in delay fault testing with power supply noise [8] has focused on generating the maximum power supply noise on one path at a time. However, the maximum noise may be considerably greater than the mission-mode worst-case noise. Moreover, the method may be in competition with other goals such as crosstalk generation that may have greater impact on path delay.

Tirumurti [4] proposed a fault modeling method that added power noise to a generalized fault model [9]. A vector-less strategy was used due to the high simulation cost of the power noise model on large circuits.

Our work, based on a path delay fault ATPG tool [1], combines power supply noise analysis with static vector compaction. Random fill of don't care bits is used to increase fortuitous detection of non-target defects. However, this can produce overkill due to excessive supply noise [10]. Worse, it may be compaction alone that generates excessive activity [11]. In order to avoid overkill due to power supply noise, our goal is to generate compacted vectors with power supply noise up to the mission-mode level on targeted paths. This approach is different from previous work that focused on maximizing noise, or only analyzed the location of the problems, or simulated their effect.

A novel approach is proposed in this paper for vector-based, layout-aware power noise estimation. It avoids complicated power network analysis, thus is much faster than existing power noise analysis tools. This approach is then integrated with the compaction procedure in order to control the power supply noise level. ISCAS89 benchmarks have been used in the experiment, in which we show the validity and efficiency of our method.

This paper is organized as follows. Section 2 provides background for power supply noise and our solution to noise estimation. Section 3 describes the compaction algorithm. Section 4 includes data to estimate the error of our power noise estimation model. Experimental results on ISCAS89 benchmarks are presented in Section 5. Section 6 concludes with directions for future research.

2. Estimation of Power Supply Noise Effect

As mentioned in Section 1, some work [5][6][7] has been proposed for power supply noise estimation. Despite their comprehensiveness and accuracy, these approaches are too expensive to be applied to large circuits during vector compaction. Hence, we need a new vector-based solution that can quickly estimate power supply noise.

2.1. Power Supply Noise

Power supply noise consists of two major components: the *IR* drop due to wire resistance, and the *Ldi/dt* due to wire inductance. Both components can be observed on the package and on-chip power grid. Generally, the *Ldi/dt* drop is predominant on the package, since the package lead resistance is low; while *IR* drop is predominant on the chip due to high interconnect resistance.

Traditionally, only the on-chip resistive *IR* drop has been addressed, so most analysis tools model the power grid as an RC network. As we move into deep submicron design with higher frequency and circuit density, the *Ldi/dt* drop becomes a significant concern. To accurately model and analyze *Ldi/dt* drop, a RLC network is necessary. However, at frequencies below about 1 GHz, inductive drop can be neglected.

2.2. Power Region Model

Much work [12][13][14] has been published on transient power grid analysis. However, RC network analysis is too expensive for compaction. Therefore, we make several approximations to simplify the problem.

Power grid analysis [4] of bumped chips shows that the supply voltage impact of a switching transient is contained within a local area, since most current flows through nearby pads. Therefore we assume that the supply voltage within a region (e.g. between a set of power pads) is uniform, and the voltage of each region is independent of each other. Hence, voltage drop for any gate in the region is identical. In addition, all switching activities across the region are equivalent, and any switching events outside the region can be neglected. The error of this approximation, along with several other approximations introduced later, will be estimated by simulation in Section 4.

Our second approximation is that the off-chip current in a region comes from a current source that averages the previous K clock cycles of current consumption (based on the off-chip time constant), and the on-chip current in a region comes from the on-chip decoupling and parasitic supply capacitance within the region. The decoupling capacitors are modeled as a single lumped capacitor between power and ground. The on-chip *Ldi/dt* drop is neglected for simplicity. The resistance factor is also ignored in this model so that the analysis becomes much easier than a traditional RLC network. Our model approximates the supply grid voltage as stepwise constant across the chip.

Voltage drop occurs on both supply and ground nets. A complete voltage drop analysis should take both networks into account. However, most prior work focuses only on the power supply network, with the assumption that power and ground can be separated [14]. Considering the fact that ground bounce is a similar phenomenon, we further assume that the ground network is ideal, which means the ground bounce is not taken into account in this work.

Our simplified Power Region model is illustrated in Fig. 1. C_d is the distributed decoupling capacitance in a region, and C_p is the total parasitic capacitance of devices and interconnect within the region connected to the power supply network in the current clock cycle. All switching gates that draw current from the supply within this region during the clock cycle are modeled as time-varying current sources $I_{switching_i}$. The switching current model is discussed in Section 2.3. $I_{on\text{-}chip}$ is the current from the on-chip capacitance, and $I_{off\text{-}chip}$ is the current from the pads.

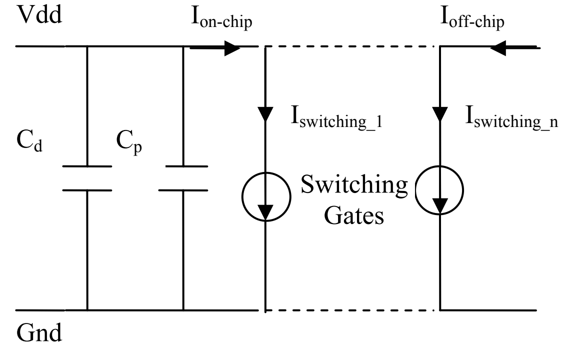

Fig. 1. Simplified power supply model within a region.

The maximum regional voltage drop during a clock cycle ΔV_{max} is:

$$\Delta V_{max} = (\int I_{on\text{-}chip}) / (C_d + C_p)$$
$$= (\sum\int I_{switching_i} - \int I_{off\text{-}chip})/(C_d + C_p) \qquad (1)$$

We assume that $\int I_{switching_i}$ occurs over the time of the nominally longest path delay during that clock cycle. After the switching transitions, V_{DD} recovers through $I_{off\text{-}chip}$ to V_{DDinit} at the start of the next cycle.

2.3. Circuit Switching Model

We must calculate $\int I_{switching_i}$ for each logic gate in order to compute ΔV_{max}. Tirumurti [4] created a table of peak power and ground currents for different values of gate output load and input slope by simulation. We adopt a similar strategy. Fig. 2 shows a typical waveform for an inverter. This waveform is approximated as triangular if the load is small, otherwise as a trapezoid, in order to compute the total charge of each transition. A similar approach was used by Chen [6].

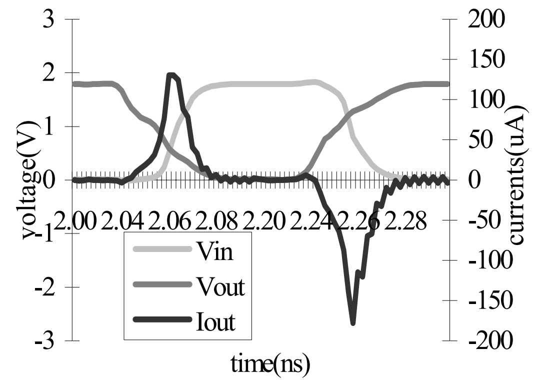

Fig. 2. A current waveform for an inverter.

2.4. Delay vs. Supply Voltage

The supply voltage is not constant during a clock cycle. As a result, the V_{DD} that a gate sees depends on its location on a path, with later gates seeing lower voltages than earlier gates. In order to avoid analysis of time-varying supply voltage, the effective supply voltage seen by gates

on a path is the average of V_{DDinit} and $V_{DDinit} - \Delta V_{max}$, the initial and worst-case voltages during a clock cycle. During the clock cycle transitions consume charge from the local supply grid capacitance and the voltage falls. Making the realistic assumption that $I_{switching}$ is higher than $I_{off\text{-}chip}$, the worst-case voltage occurs when the last path within a region stops propagating. If paths are of similar length and gates along the path have similar delay sensitivities, then the average voltage will be a reasonable approximation.

The delay of a gate can be modeled as a function of its supply voltage, input slope and output load capacitance. Several models been proposed for delay functions. Bai proposed the following delay equation [15].

$$t_d = A + BV_{DD} + CV_{DD}^{\ 2} \tag{2}$$

The coefficients can be obtained by simulation data analysis. Bai also suggested linear functions of supply voltage with appropriate coefficients [15] if the voltage drop is not too large. The error of this linear model was estimated to be less than 5%. Hence, our model of rising delay increase on gates is expressed as follows:

$$\Delta delay\ /\ delay = \delta\Delta V\ /\ V_{DD} \tag{3}$$

where delay is the standard delay under ideal supply voltage, ΔV is the estimated voltage drop, and V_{DD} is the ideal supply voltage. A table of coefficients δ under different output loads and input slopes is obtained by simulation for each standard gate type.

2.5. Power Supply Noise Estimation Procedure

Fig. 3 is the flow chart of the noise estimation procedure. To estimate the power noise effect of a vector (a vector pair for delay faults), we first use logic simulation to find transitions on all nets in the circuit. Layout information is then needed to estimate voltage drop for each region. In practice, only those regions traversed by the targeted path need to be considered. We then calculate path delay with our delay model.

The time complexity for this procedure is $O(G)$, where G is the total number of gates of the circuit. This means that our estimation approach has the same time complexity as logic simulation.

3. Compaction

A simple greedy static compaction strategy is used. Vectors are considered one by one in order and combined with the first compatible vector found in the compacted vector list. We also implemented a static compaction tool using simulated annealing in order to find a close-to-optimal solution for compaction. Our experiments show that the results of greedy algorithms are close to optimal.

Experiments have been performed on several ISCAS89 benchmarks and an industrial circuit, “Controller 1”. The results are shown in Table 1. The vectors, generated by CodGen [16], are launch-on-shift robust path delay tests targeting the longest rising and falling transition path through every line in the circuit (termed KLPG-1). Two greedy algorithms are implemented. Greedy I loads vectors for compaction in a forward order, while Greedy II uses a backward order. The smaller test set is then selected from the two. The simulated annealing algorithm chooses two vectors at random for compaction and uses a weighted heuristic to determine whether the move is accepted. Table 1 includes the compacted test size and running time for the two greedy algorithms and simulated annealing. It is shown that the greedy approach generates 1-2% more tests (in the last column of the table) than simulated annealing.

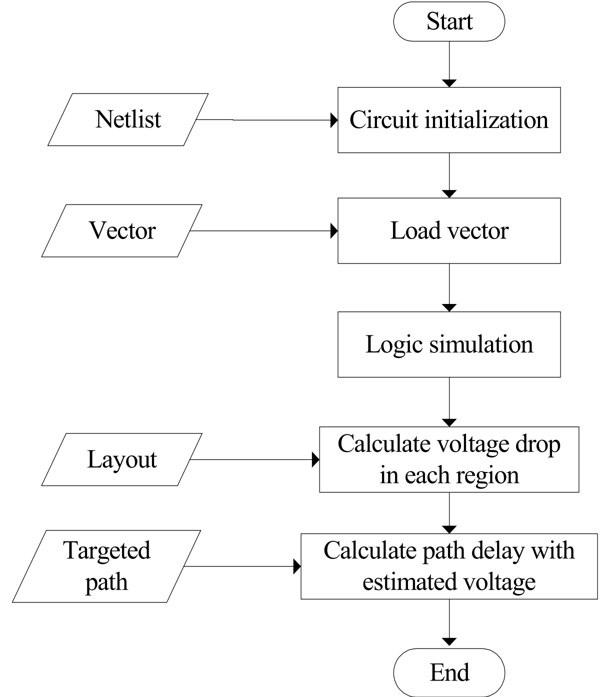

Fig. 3. Power noise estimation procedure.

Table 1. Compaction results for greedy algorithms and simulated annealing.

Circuit	Initial Test Size	Greedy I forward		Greedy II backward		Greedy Test Size	Simulated Annealing		Greedy vs. SA (%)
		Test Size	Time (s)	Test Size	Time (s)		Test Size	Time (s)	
s1423	395	216	3	215	4	215	212	300	1.4
s1488	192	88	1	86	1	86	85	1457	1.2
s1494	193	86	1	84	1	84	83	1503	1.2
s13207	3220	916	46	899	82	899	901	6 d	-0.2
Controller 1	12274	2325	405	2232	892	2232	2203	30 d	1.3

The key goal of our compaction tool is that the power noise effect for all compacted vectors is within the mission-mode level, with compaction rate only the second concern. There are various ways to define the mission mode noise level. The simplest approach is to use the maximum voltage drop specified by the power grid designer. If silicon is available, an empirical approach is to apply functional vectors to the circuit using ATE and

measure the overall supply noise. The worst-case voltage drop can be selected as an upper bound for all regions for all vectors during compaction. We can indirectly specify a noise constraint upon the maximum noise-induced delay increase on all targeted paths of a vector. This approach is favored since it directly targets the cause of supply noise overkill – slow paths.

The comprehensive compaction procedure is illustrated in Figure 4. Un-compacted vectors are loaded in order and a quick pre-check is performed. The pre-check step is a rough prediction of whether the vector has a chance to exceed the power noise limit. Basically, the transition count in the input vector pair can be taken as an estimator [17]. A transition count threshold can be set by experience, so that any vectors with fewer input transitions can be assumed "safe". This pre-check step is extremely fast as it only scans the vector without simulation. However, this step should not be performed if the power noise level must be guaranteed, since there are rare cases where a few transitions on circuit inputs generate a large amount of switching activity. Next, if the un-compacted vector exceeds the power noise limit, it is saved in a separate vector list. The high power noise level of vectors in this list is due to ATPG instead of compaction. Such vectors should be rare given the low care bit density in path delay test vectors [16]. If the power noise level for that vector is within limits, compaction is performed. Whenever a compatible vector in the compacted vector list is found, power noise estimation is performed on the new compacted vector. If the power noise level is within limits, the new compacted vector is kept. Otherwise, the compaction is invalid and the next compatible vector is considered.

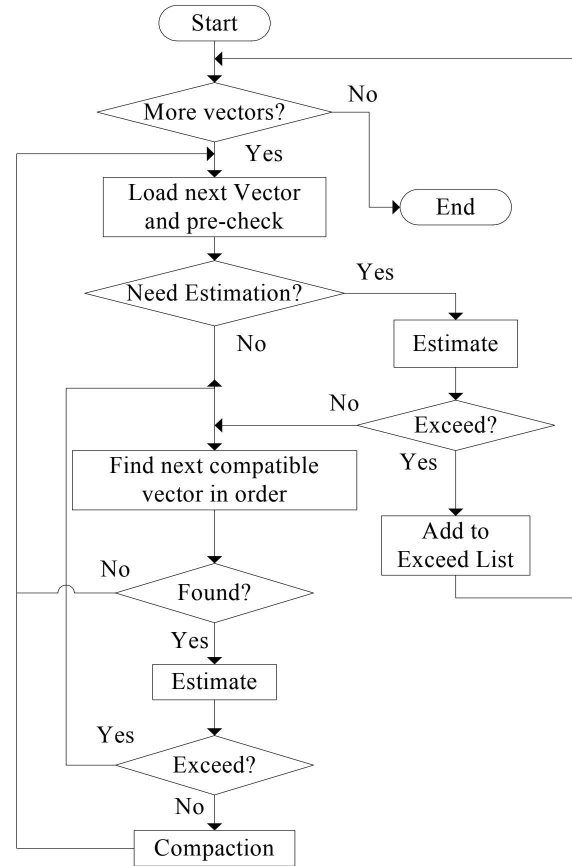

Fig. 4. Compaction flow chart

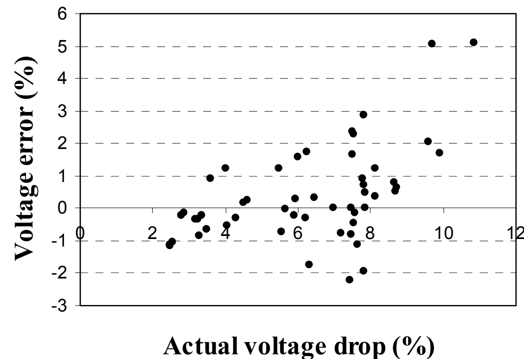

Fig. 5. Voltage error versus actual voltage drop for vectors of s1488

4. Error Evaluation

We need to estimate the error introduced by our approximations. Cadence Spectre was used to simulate ISCAS89 circuit s1488, implemented in 180 nm technology with a realistic RLC supply. We assume V_{DDinit} is the nominal V_{DD} of 1.8 V and $I_{off\text{-}chip}$=0. We recorded the worst-case voltage drop for all gates on all targeted paths of each vector, and then compared it with our estimate.

Fig. 5 illustrates the voltage error distribution for 55 vectors randomly selected from the KLPG-1 test set versus their actual voltage drop from Spectre. The voltage error is the difference between the worst-case voltage level estimated by our tool and simulated by Spectre divided by the Spectre result. The voltage error for each vector is defined as the maximum error among all the gates on its targeted path. The average voltage error is 1%, with 96% of the vectors having error within ±3%. Outliers appear when the actual voltage drop exceeds 10% of the nominal supply voltage. The average of voltage error is 1%. A suitable guardband can be used to avoid overkill due to model error while minimizing test escapes.

5. Experimental Results

A combined static compaction tool has been developed in C++ and run on a 2.3 GHz Pentium 4 system. The experiments have been performed on ISCAS89 benchmarks implemented using 180 nm, 1.8 V technology. Path delay test sets used for compaction have been generated by CodGen [16]. They are KLPG-1 tests using launch-on-capture.

Experimental results for s38417 are given in Table 2 and Table 3. We assume that the there are 4 supply pads for this benchmark, and each region is a square centered around a pad. Thus, a region contains 6K gates in this case. We believe that if we perform a static forward-order compaction without noise analysis, the resulting test set (denoted as *s*) can serve as an approximate lower bound for any compaction method that considers power supply noise. The un-compacted test set (denoted as *ucs*) for this benchmark contains 13941 vectors (*ucs* = 13941) and has a fill-rate of 2.5%. For this test set, *s* is 940 with a fill-rate of 25%, and the static forward-order compaction without noise for s38417 takes 95 seconds. The V_{DDinit} is set to the nominal V_{DD} and $I_{off\text{-}chip}$=0, simulating the typical *Ldi/dt* problem of scan delay test, in which off-chip inductor currents fall to their leakage level between the last shift clock and the launch clock. The pre-check procedure is skipped by setting the transition count threshold to be the number of input pins.

Two kinds of constraints on power supply noise have been implemented. One is maximum voltage drop in any region, while the other is maximum path delay increase caused by power supply noise. Table 2 shows how the compaction results vary with the power supply noise constraint. The first column in Table 2 shows the specified maximum voltage drop. Column 2 is the compacted test set size, and column 3 lists α, the percentage increase in compacted test set size due to the noise constraint. Column 4 lists the number of calls to the power noise estimation procedure during compaction, and column 5 lists the failure ratio β, the fraction of the time that a potential vector compaction exceeds the noise constraint. Column 6 is the number of vectors that exceeds the noise constraint prior to compaction. The last column shows CPU time.

We can also use our delay model described in Section 2. The maximum delay increase on all the targeted paths of a vector is constrained. Since delay is the major concern of the path delay test, it is the eventual estimate of the power supply noise effect on delay testing. Table 3 has similar content as Table 2. The only difference is that the constraint is the maximum percentage increase of path delay listed in column 1.

Table 2 and Table 3 show that a tighter constraint on maximum voltage drop or path delay generally results in a larger compacted test set. A tighter constraint also costs more CPU time since more compaction choices are rejected. Delay constraints further increase running time due to the need to estimate the delay of every path of the vector. We also find that the increase of compacted test set size is relatively small compared with the increase in estimation calls. The main reason is that the fill rate of un-compacted vectors is quite low, so that most vectors will finally get compacted after a number of trials. In other words, our compaction tool tends to compact vectors in a way that power supply noise is more evenly distributed among all vectors. Note that in Table 2, when 9% maximum voltage drop is allowed, the compacted test size is smaller than *s*, due to order-dependence of the greedy algorithm.

Table 2. Compaction results for s38417 with a regional voltage drop constrained (ucs = 13941, s = 940).

Max Voltage Drop (%)	Test Size	α (%)	Estimation Calls	β (%)	Originally failed vectors	CPU Time (Hr)
5	1 212	28.9	254 612	91.0	508	6.1
6	1 104	17.4	106 159	77.5	105	2.7
7	956	1.7	38 119	36.7	3	1.0
8	941	0.1	25 411	5.0	0	40 min
9	938	-0.2	24 209	0.2	0	38 min
10	940	0	24 145	0.0	0	38 min

Table 3. Compaction results for s38417 with path delay increase constrained on targeted paths (ucs = 13941, s = 940).

Max. Delay Increase (%)	Test Size	α (%)	Estima-tion Calls	β (%)	Originally failed vectors	CPU Time (Hr)
10	1 116	18.7	62 448	61.9	147	7.0
11	991	5.4	47 188	49.0	36	6.0
12	952	1.3	32 271	25.2	6	4.8
13	942	0.2	26 737	9.7	0	4.2
14	941	0.1	24 652	2.1	0	4.1

Fig. 6 plots the overhead α against the constraint on maximum voltage drop for several ISCAS89 benchmarks including s38417. The power grid design is similar to s38417. Fig. 7 plots the overhead against the constraint on maximum delay. All the circuits show the same trend, though sensitivity varies.

As discussed in Section 2, the on-chip decoupling capacitance will affect voltage drop. Ratio γ is defined as the on-chip power grid capacitance divided by the total signal net capacitance of the circuit. In the previous experiments, γ=1. In real chips, γ is typically larger. Further experiments have been performed to show the relationship of α and γ, Larger values of γ are obtained by increasing on-chip decoupling capacitance. As shown in Fig. 8, the compacted test set size increases as γ decreases when the voltage drop constraint is set to 10%. Note that the larger the circuit, the less overhead for a given γ. This is due to the averaging effect of switching activity in a large circuit.

6. Conclusions and Future Work

This paper presented a static compaction solution that reduces overkill induced by excessive power supply noise. A fast power supply noise estimation solution was presented and the error of the estimation evaluated. This

power supply noise estimation approach has been integrated with static compaction, and demonstrated on ISCAS89 benchmarks. It was shown that supply noise constraint, delay constraint and power grid capacitance have a strong impact on compaction.

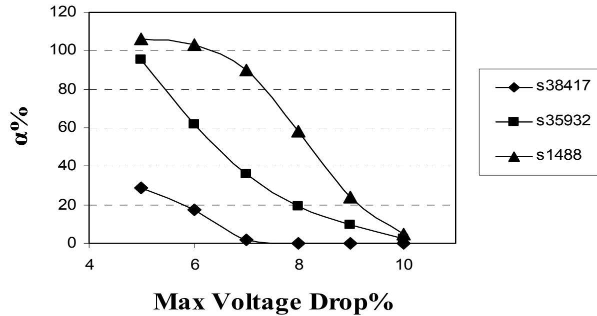

Fig. 6. Overhead α% at different levels of maximum voltage drop.

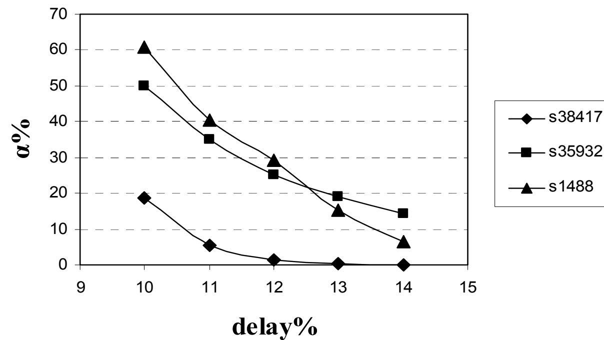

Fig. 7. Overhead α% at different levels of maximum path delay increase.

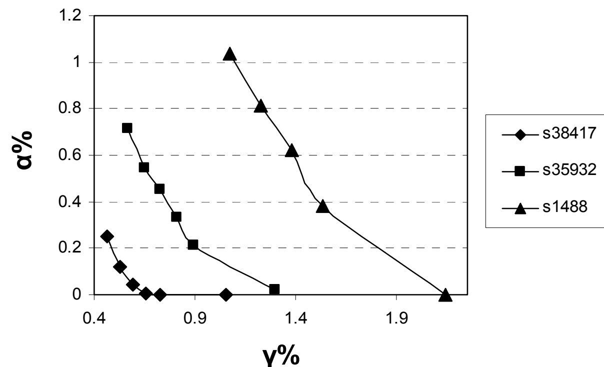

Fig. 8. Overhead α% vs. γ%.

In the future, we will extend this work to dynamic compaction within the CodGen KLPG test generator [16]. This requires improving algorithm performance, by reducing the cost of power noise and delay estimation, through the use of incremental analysis.

Acknowledgements

This research was funded in part by Texas Advanced Research Program grant 512-186-2001 and by National Science Foundation contract CCR-1109413.

References

[1] W. Qiu and D. M. H. Walker, "An Efficient Algorithm for Finding the K Longest Testable Paths Through Each Gate in a Combinational Circuit," IEEE Int'l Test Conf., Charlotte, NC, Sept. 2003, Vol. 1, pp. 592-601.

[2] K. L. Shepard and V. Narayanan, "Noise in Deep Submicron Digital Design," IEEE/ACM Int'l Conf. on Computer Aided Design, San Jose, CA, Nov. 1996, pp. 524-531.

[3] Y.-M. Jiang and K.-T. Cheng, "Analysis of Performance Impact Caused by Power Supply Noise in Deep Submicron Devices," ACM/IEEE Design Automation Conf., New Orleans, LA, June 1999, pp. 760-765.

[4] C. Tirumurti, S. Kundu, S. Sur-Kolay, Y.-S. Chang, "A Modeling Approach for Addressing Power Supply Switching Noise Related Failures of Integrated Circuits,", Design, Automation and Test in Europe Conf. and Exhibition, Paris, France, Feb. 2004, pp. 1078-1083.

[5] Y.-S. Chang, S. K. Gupta and M. A. Breuer, "Analysis of Ground Bounce in Deep Sub-Micron Circuits," IEEE VLSI Test Symp., Monterey, CA, Apr. 1997, pp. 110-116.

[6] H. H. Chen and D. D. Ling, "Power Supply Noise Analysis Methodology for Deep Submicron VLSI Chip Design," ACM/IEEE Design Automation Conf., Anaheim, CA, June 1997, pp. 638-643.

[7] J.-J. Liou, A. Kristic, Y.-M. Jiang and K.-T. Cheng, "Modeling, Testing, and Analysis for Delay Defects and Noise Effects in Deep Submicron Devices," IEEE Trans. on Computer-Aided Design, vol. 22, no. 6, June 2003, pp. 756-769.

[8] A. Kristic, Y.-M. Jiang and K. T. Cheng, "Pattern Generation for Delay Testing and Dynamic Timing Analysis Considering Power-Supply Noise Effects," IEEE Trans. on Computer-Aided Design, vol. 20, no. 3, Mar. 2003, pp. 416-425.

[9] S. T. Zachariah, Y.-S. Chang, S. Kundu and C. Tirumurti, "On Modeling Cross-talk Faults," Design, Automation and Test in Europe Conf. and Exhibition, Munich, Germany, Mar. 2003, pp. 490-495.

[10] J. Saxena, K. M. Butler, V. B. Jayaram, S. Kundu, N. V. Arvind, P. Sreeprakash, M. Hachinger, "A Case Study of IR-Drop in Structured At-Speed Testing," IEEE Int'l Test Conf., Charlotte, NC, Sept. 2003, pp. 1098-1104.

[11] R. Sankaralingam, R. R. Oruganti and N. A. Touba, "Static Compaction Techniques to Control Scan Vector Power Dissipation," IEEE VLSI Test Symp., Montréal, Québec, Canada, Apr. 2000, pp. 34-40

[12] S. R. Nassif and J. N. Kozhaya, "Fast Power Grid Simulation," ACM/IEEE Design Automation Conf., Los Angeles, CA, June 2000, pp. 156-161.

[13] H. Qian, S. R. Nassif and S. S. Sapatnekar, "Random Walk in a Supply Network," ACM/IEEE Design Automation Conf., Anaheim, CA, June 2003, pp. 93-98.

[14] Z. Zhu, B. Yao and C.-K. Cheng, "Power Network Analysis Using an Adaptive Algebraic Multigrid Approach," ACM/IEEE Design Automation Conf., Anaheim, CA, June 2003, pp. 105-108.

[15] G. Bai, S. Bodda and I. N. Hajj, "Static Timing Analysis Including Power Supply Noise Effect on Propagation Delay in VLSI Circuits," ACM/IEEE Design Automation Conf., Las Vegas, NV, June 2001, pp. 295-300.

[16] W. Qiu, J. Wang, D. M. H. Walker, D. Reddy, X. Lu, Z. Li, W. Shi and H. Balachandran, "K Longest Paths Per Gate (KLPG) Test Generation for Scan-Based Sequential Circuits," IEEE Int'l Test Conf., Charlotte, NC, Oct. 2004, pp. 223-231.

[17] A. Kokrady and C. P. Ravikumar, "Fast, Layout-Aware Validation of Test-Vectors for Nanometer-Related Timing Failures," Int'l Conf. on VLSI Design, Bombay, India, Jan. 2004, pp. 597-602.