

A Vector-based Approach for Power Supply Noise Analysis in Test Compaction

Jing Wang⁺, Ziding Yue^{*}, Xiang Lu^{*}, Wangqi Qiu⁺, 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 solution is described to prevent such overkill. Low-cost power supply noise models are developed and used in compaction. An error analysis of these models is given. This paper improves on prior work in terms of models and algorithm to increase accuracy and performance. Experimental results are given on ISCAS89 circuits.

1. Introduction

Due to reduced timing margins and increased clock rates, delay testing has become a critical concern. To detect small manufacturing defects that do not cause functional failure but reduce the speed of circuits, at-speed delay testing using the path delay fault model [1] has been used. However, as semiconductor technology is scaled, designs are becoming more and more sensitive to various noise sources [2], such as leakage noise, crosstalk and power supply noise. Excessive noise causes performance degradation and signal integrity problems. Moreover, it has a significant impact on the timing performance of deep sub-micron (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 in successive technology generations [4], more simultaneous switching activity per area is expected. In addition, DSM technology requires the use of reduced supply voltages. Experiments show that if the supplies are allowed to vary by up to 12.5%, one can observe by simulation up to a 2.4x increase in gate delay in 130 nm CMOS [5]. Industrial data for sub-90 nm CMOS gates also shows that delay sensitivity rises as supply voltage decreases [4]. These trends lead to a larger power noise impact on delay.

Several techniques [6][7] have been proposed for estimating power supply noise. Different supply network and circuit models were used to achieve accuracy. Jiang et al. [3] proposed a vector-independent approach using genetic algorithms. Liou et al. [8] proposed an estimation method based on a statistical timing analysis framework.

Traditionally, a combined package/on-chip power grid model consists of a number of circuit elements [9]:

- 1) RLC model of package leads, ball grid arrays, power planes
- 2) RC model of on-chip power interconnect
- 3) RC model of intrinsic decoupling capacitance of non-switching devices and n-well regions
- 4) RC model of explicitly designed decoupling capacitance
- 5) Model of AC currents of switching devices

Most prior work in testing while considering power supply noise adopts a vector-less strategy due to the high simulation cost of the power supply noise model on large circuits. Tirumurti et al. [4] proposed a fault modeling method that added power noise to a generalized fault model [10]. Pant et al. [11] also proposed a vector-less analysis approach for computing the maximum path delay under power supply fluctuations.

Kristic et al. [12], using vector-based approach, 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.

The research described here is based on our prior work on considering power supply noise during static test vector compaction [13]. That work used a path delay fault ATPG tool [1] to generate test sets and combined power supply noise and delay analysis with static vector compaction. Random fill of don't care bits can be applied to test vectors to increase fortuitous detection of non-target defects. However, this can produce overkill due to excessive supply noise [14]. Worse, it may be compaction alone that generates excessive activity [15]. In order to avoid such overkill, we generated compacted vectors with power supply noise up to the mission-mode level on targeted paths. This included compaction based on a worst-case voltage noise target and a worst-case delay target. A novel power model for fast vector-based power noise analysis was adopted, which avoided the complicated and costly power network analysis. A simple linear delay model was applied to calculate path delay under noise. Model verification compared with Cadence Spectre simulation was also presented. For all the vectors analyzed, the average worst-case voltage error was about 1%, and the average worst-case delay error was 2.8%.

However, in our previous work, off-chip inductance factor was effectively neglected given the assumption that the time constant of the RLC circuit is relatively large compared with longest nominal delay, so that the current provided from off-chip during path propagation is insignificant. In addition, the linear delay model was not sufficient to model the variation of device voltage levels and their impact on delay. A third problem was that our prior work only considered the current due to load capacitance charging, and did not consider the short-circuit current. An additional shortcoming of the previous work was that compaction for a delay target was much more costly than compaction for power supply noise alone.

This paper proposes improved power region and delay models for vector-based, layout-aware power noise estimation. We model both on-chip voltage drop and package lead effects while keeping the same computation complexity as our previous work, so the power noise analysis is faster than other available methods. A gate delay calculation method considering both temporal and spatial voltage level variation [16] is then implemented to estimate path delay. This approach is then integrated with the compaction procedure in order to control the power supply noise level. The static compaction algorithm adopts the same framework as in previous work. However, the program has been improved to significantly reduce CPU time. ISCAS89 benchmarks have been used in the experiment to 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 introduces the method implemented in our tool to calculate propagation delay considering voltage level variation. Section 4 describes the compaction algorithm. Section 5 includes data to estimate the error of our power noise estimation model. Experimental results on ISCAS89 benchmarks are presented in Section 6. Section 7 concludes with directions for future research.

2. Estimation of Power Supply Noise

As discussed in Section 1, some work [6][7][8] has been proposed for power supply noise estimation. Despite their comprehensiveness and accuracy, these approaches are much too expensive to be applied to large circuits during vector compaction. Hence, we need a vector-based solution that can quickly and accurately 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 noise due to wire inductance. Both components can be observed on the package and on-chip power grid. Generally, the Ldi/dt noise is predominant on the package, since the package lead resistance is low; while IR drop is predominant on the chip due to high resistance.

Traditionally, only the on-chip resistive IR drop has been addressed, so most analysis tools model the on-chip power grid as a RC network. However, as we move into deep submicron design with higher frequency and circuit density, the Ldi/dt noise becomes a significant concern. To accurately model and analyze Ldi/dt , a RLC network is necessary. A comprehensive package/on-chip power grid model was introduced in [9].

2.2. Simplified Power Region Model

Much work [17][18][19] has been published on transient power grid analysis. However, RLC or RC network analysis is much 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 in Section 5.

Our second approximation is that the on-chip current in a region, denoted as $I_{on-chip}$, 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 noise is neglected for simplicity. On-chip wire resistance 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.

Third, we assume that the off-chip current in a region, denoted as $I_{off-chip}$, comes from a constant current source. This current source averages the previous K clock cycles of current consumption (based on the off-chip time constant). Thus, $I_{off-chip}$ must be taken into consideration if a lot of switching activities occur in the previous cycles. However, during scan test, the scan cycle is much longer than the mission mode cycle, so a chip is usually in the idle state prior to the launch of a delay test vector. If the off-chip time constant is comparable to the scan clock time, $I_{off-chip}$ becomes insignificant and can be ignored.

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 [19]. 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. $I_{on-chip}$ is the current from the on-chip capacitance, and $I_{off-chip}$ is the current from the pads.

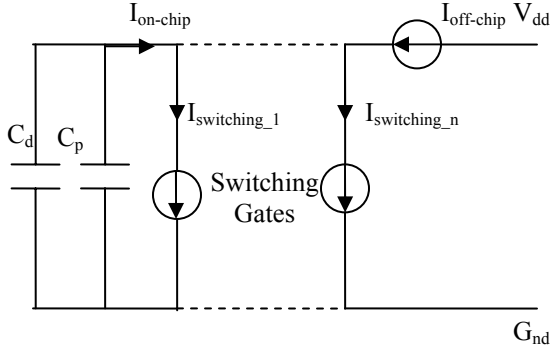


Fig. 1. Simplified power supply model within a region.

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

$$\Delta V_{max} = (I_{on-chip}) / (C_d + C_p), \quad (1)$$

$$\Delta V_{max} = (\sum I_{switching_i} - I_{off-chip}) / (C_d + C_p), \quad (2)$$

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

2.3. Circuit Switching Model

We must calculate $I_{switching_i}$ for each logic gate in order to compute ΔV_{max} . Switching current drawn from the supply network in CMOS circuits consists mainly of two parts, the short circuit current and the charging/discharging current on output capacitive load. The latter term is usually the dominant term, due to slew rate design constraints.

Charging/discharging current in CMOS circuits is well understood and easy to estimate. Tirumurti et al. [4] created a table of peak power and ground currents for different values of gate output load and input slope by simulation. This approach incorporates both short-circuit and charging current. 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 approximation approach was used by Chen [7].

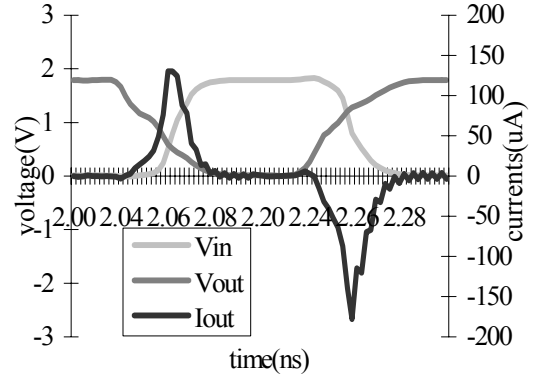


Fig. 2. Charging/discharging current waveform for an inverter.

During switching in a static CMOS gate, a direct path from the power supply to ground is established [20] that results in short circuit current. Short circuit current is dependent on the input rise/fall time, the load capacitance and gate design. When load capacitance is small enough, the short circuit current dominates the current waveform drawn from the supply network. As for charging/discharging current, we also create a table of peak current for different values of gate output load and input slope by simulation. Input slope for each gate is computed by static timing analysis assuming nominal delay. This approximation is necessary since we don't know the actual input slope for each gate before estimating voltage drop and apply our delay models. The current waveform is approximated as triangular. However, in low power designs, the shape of the short circuit current waveform is very close to triangular.

2.4. 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.

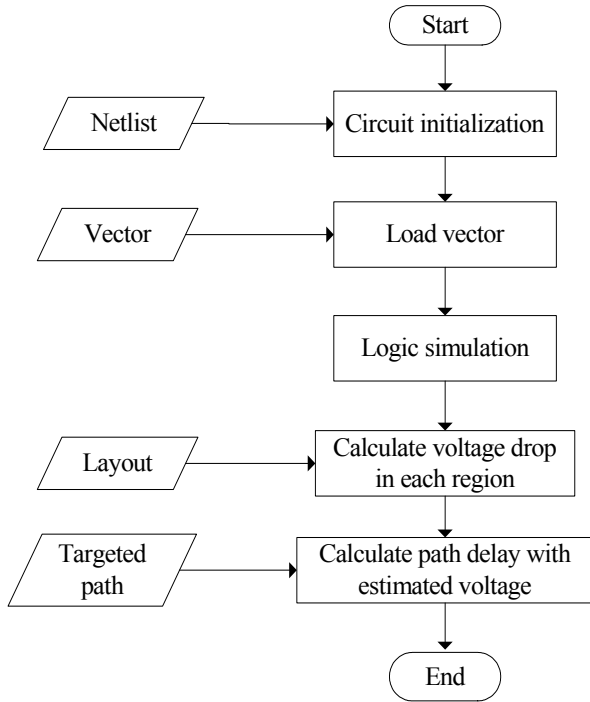


Fig. 3 Power noise estimation procedure.

3. Gate and Path Delay

Power-noise-aware timing analysis can be classified into two issues; how to compute the actual on-chip voltage levels, and how to compute propagation delay. We discussed the first problem in Section 2. In this section, we focus on propagation delay computation.

3.1. Temporal Voltage Variation vs. Delay

The supply voltage is not constant during a clock cycle due to supply noise. When the time constant of the noise shape is much larger than the transition time of a logic gate, the voltage level during the transition can be regarded as constant [16]. However, it is hard to know the actual power supply noise on a gate when its transition occurs unless we know the real noise waveform and the real time point of the transition.

In order to avoid analysis of time-varying supply voltage, the effective supply voltage seen by gates on a path is approximated as the average of V_{DDinit} and $V_{DDinit} - \Delta V_{max}$, the initial and worst-case voltages during a clock cycle. The V_{DD} that a gate sees depends on its location on a path, with later gates seeing lower voltages than earlier gates. 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 $I_{off-chip}$, 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 error due to this approximation will be evaluated in section 5.

3.2. Spatial Voltage Variation vs. Delay

The supply voltage varies both temporally and spatially. In real designs, gates in a path are not necessarily placed in the same neighborhood. If two gates, one is a driver and the other is a receiver, are placed far from each other, their supply voltage levels are very likely to be different because power supply noise varies spatially [16]. Fig. 4 shows an example of input and output waveforms for a rising transition on an inverter. The input voltage level, which is also the voltage level of the driver gate, is different from the receiver gate and the output. The charging/discharging current depends significantly on the input voltage level, thus changes gate delay. Since there are multiple inputs in many gates, the characterization cost by simulation is too expensive for implementation. We need a delay model independent of the input voltage levels.

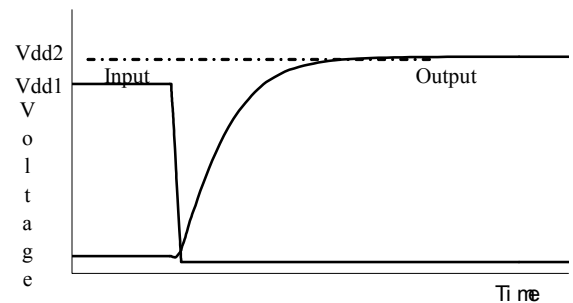


Fig. 4. An inverter with different input voltage and device voltage level

3.3. Voltage Level Equalization for Delay Calculation

An equalization method to model different driver and receiver supply voltages was proposed by Hashimoto et al [16]. Since gate delay is the time to charge/discharge the gate output load and voltage level variation causes gate delay variation by changing charging/discharging current, gate delay can be kept unchanged by increasing/decreasing the output load in the same ratio. DC analysis was executed varying all input voltage levels and a Response Surface [21] was built for charging/discharging current before and after voltage level equalization. The current ratio is then used to compute the replaced output load value. Since voltage levels of all inputs have already been equalized, only the device voltage level, output load and input slope will be taken as variables for gate delay calculation.

However, in our experiments, we found that when output load and input slope fall into different regions other

than in Hashimoto’s paper [16], the results are poor. Fig. 5 illustrates one of the cases where equalization increased the delay error instead of decreasing it. The parameters for Fig. 5 are as follows: 1) Input voltage level of 1.8 V; 2) Device voltage level of 1.7 V; 3) Output load of 0.43pf; 4) Input falling transition time of 27.75ps; 5) The inverter is built with 180 nm, 1.8 V technology. Fig. 5 shows the input and output waveforms with and without the change in output load. It is observed that load change introduces extra delay error in this case.

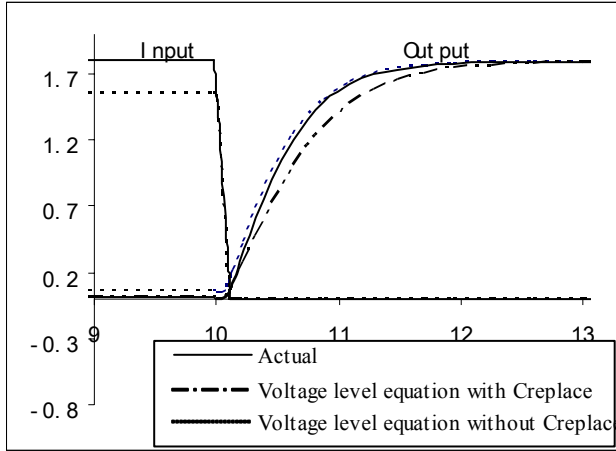


Fig. 5. Input and output waveforms of an inverter with and without output load replaced

The waveform in Fig. 5 is quite typical, since a gate with larger output load consequently has a longer gate delay, thus is more significant than gates with small load when we calculate path propagation delay. In addition, in our power region model, we assume that all gates across the region have the same voltage level. In most cases, the input voltage level and device voltage levels are similar. Thus, we choose to equalize input voltage level and device voltage level without changing output load. The final delay model is presented in section 3.4. The error of this approximation will be estimated in section 5.

3.4. Delay Model

In our work, gate delay is specified as the time interval between the input crossing timing of $V_{dd1}/2$ and the output crossing timing of $V_{dd2}/2$. The transition time is calculated in the 30% to 70% interval [22].

Several models been proposed for delay functions. Bai proposed the following delay equation [23]:

$$t_d = A + BV_{DD} + CV_{DD}^2 \quad (3)$$

where coefficients can be obtained by simulation data analysis. However, the coefficients here strongly depend on the input transition time and output capacitance. Thus our models are generalized as follows.

$$t_d = f(t_{in}, C_{out}, V_{dd}) \quad (4)$$

$$t_{out} = g(t_{in}, C_{out}, V_{dd}) \quad (5)$$

where t_d is the gate delay, t_{in} is the input transition time and t_{out} is the output transition time [16]. V_{dd} equals V_{dd2} , since the driver voltage level has been equalized to the receiver voltage level. C_{out} is the actual output load. A table method based on equation 4 and 5 has been used to calculate t_d and t_{out} .

4. Test Vector Compaction

Static compaction is a technique to reduce the size of test set following test generation. Many static compaction algorithms have been proposed for sequential circuits [24][25][26]. In our work, 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.

In our previous work [13], we implemented a static compaction tool using greedy algorithms and simulated annealing. We believe that we can find a close-to-optimal solution for compaction using simulated annealing. Experiments were performed on several ISCAS89 benchmarks and an industrial circuit with launch-on-shift robust path delay vectors generated by CodGen [27]. Our experiments showed that the greedy approach generates 1-2% more tests than simulated annealing, thus close to optimal.

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 Fig. 6. It is similar to our prior work with one change. Un-compacted vectors are loaded in order and a quick pre-check is performed. This pre-check step will be discussed below. 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 [27]. 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, a pre-check is performed to see if we can skip power noise estimation for the compacted vector. Power noise estimation is performed if the pre-check fails. 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.

The pre-check step is a rough prediction of whether the vector has a chance to exceed the power noise limit, using the transition count in the input vector pair as a noise estimator [28]. A transition count threshold must 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. In our work, the threshold is set based on our prior compaction experience. The pre-check step should not be performed if the power noise level must be guaranteed considering those rare cases where a few transitions on circuit inputs generate a large amount of switching activity.

Analysis of prior performance results showed that a large amount of time during compaction was spent indexing gates on the target path for delay evaluation. The array data structure was replaced by a hash table to significantly improve performance.

5. Error Evaluation

We need to estimate the error introduced by our approximations. Cadence Spectre was used to simulate ISCAS89 benchmark s1488 implemented in 180 nm technology with a realistic RLC supply network. We set V_{DDinit} to the nominal V_{DD} of 1.8 V, and we recorded relevant simulation data and then compared it to our estimate.

Pattern sets for s1488 were generated by using the CodGen path delay test generator [29]. Robust launch-on-capture path delay tests targeting the longest rising and falling transition path through every line in the circuit (termed KLPG-1) were generated. One path was targeted per pattern. Since we have not implemented functions in our current timing model to find the exact path that causes ultimate delay, only static sensitized path delay tests are used, for which all side inputs of the targeted path are restricted to be static non-controlling. These static sensitized path delay tests are also free from glitches which may cause unexpected delay error. The “don’t care” bits of the patterns were then filled for minimum transitions.

First, we wish to estimate the voltage drop error. We recorded the simulated worst-case voltage drop for all gates on all targeted paths of each vector, and then compared it with the result from our calculation. Fig. 7 compares the voltage drop calculated by our estimation tool with the actual voltage drop by Spectre for the 46 vectors. The data are roughly fit with $R^2 = 0.57$.

Second, we need to know how much error is caused by our delay model. Similarly, Fig. 8 plots the nominal delay calculated by our tool versus the actual nominal delay simulated by Spectre. With $R^2 = 0.95$, we found that the error of the nominal delay model is relatively small compared with the voltage drop error.

Finally, we want to estimate the delay error under power supply noise. We can see from Fig. 9 that the

correlation $R^2 = 0.20$ is poor. Since the error of the nominal delay model is much lower than this, the error mainly comes from: 1) the estimation of worst-case voltage drop; 2) the assumption of uniform voltage level across the region, neglecting the spatial variation of the supply; and 3) averaging the voltage level for path delay calculation, neglecting the voltage variation during the clock cycle. We will further evaluate these error sources in future work in order to minimize them.

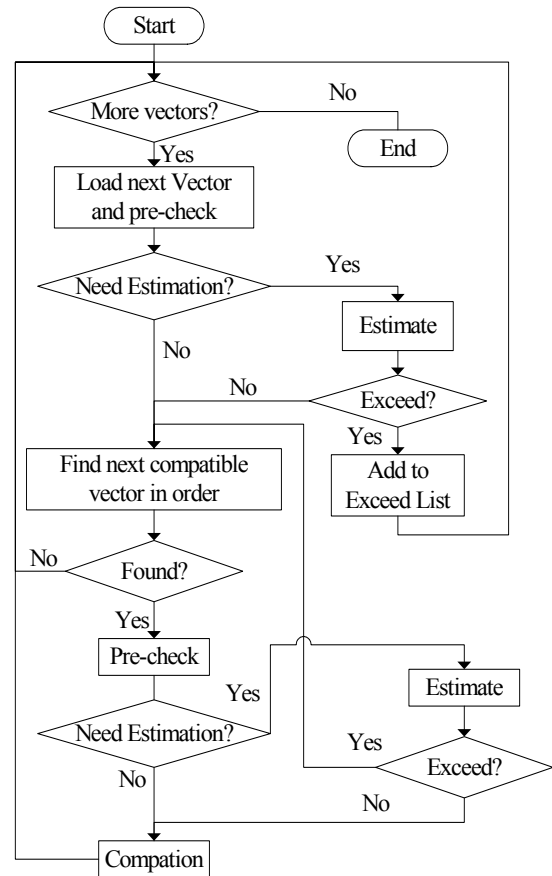


Fig. 6. Compaction flow chart.

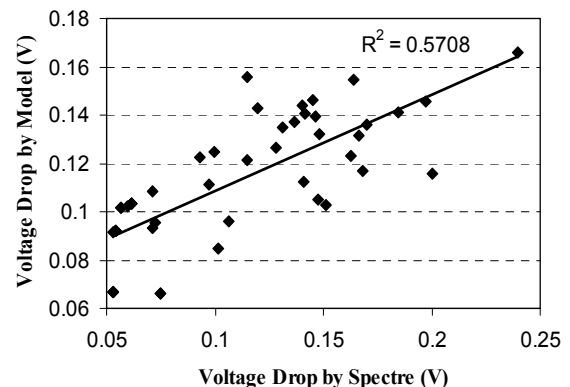


Fig. 7. Voltage drop by our estimation vs. actual voltage drop by Spectre for vectors of s1488.

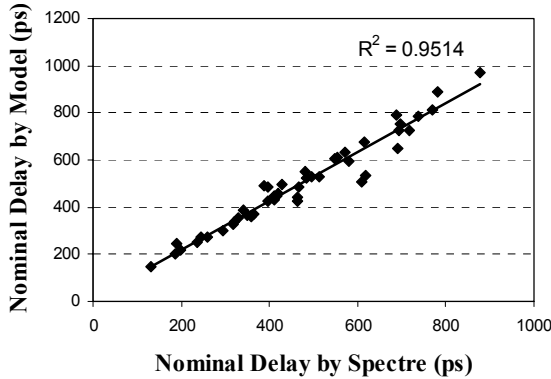


Fig. 8. Nominal Delay by our estimation vs. actual nominal delay by Spectre for vectors of s1488.

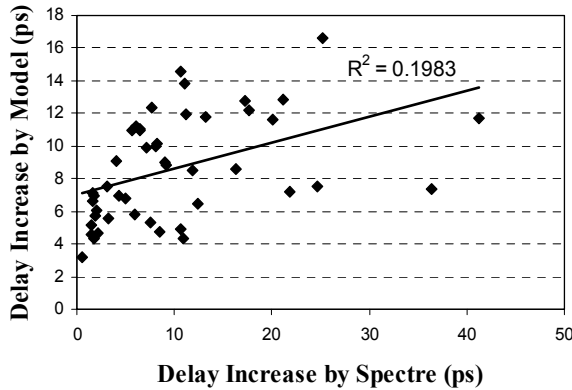


Fig. 9. Delay increase calculated by our estimation vs. delay increase by Spectre for circuit s1488.

6. Experimental Results

6.1. Compaction

A combined static compaction tool was developed in our prior work. The algorithm was revised as discussed above, and our improved power region and delay models incorporated. The tool was written 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 were generated by CodGen [27]. They are KLPG-1 robust tests using launch-on-capture targeting the longest rising and falling transition path through every line in the circuit (termed KLPG-1). All “don’t care” bits of the patterns are set to minimum transition values so as not to introduce extra noise.

Experimental results for s38417 are given in Table 1 and Table 2. We assume that there are 4 supply pads for this benchmark, and each region is a square centered around a pad. Thus, each region contains approximately

6000 gates. 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 our previous work, γ was set to 1. However, since our improved circuit switching model has integrated short-circuit current, we set γ to 3.8 so as to keep voltage drop typical. The V_{DDinit} of each cycle is set to the nominal V_{DD} , assuming the cycle time is set such that voltage waveform returns approximately to the nominal voltage level at the start of the cycle. Since $I_{off-chip}$ is the average current consumed in the previous K cycles, we arbitrarily set $I_{off-chip}=0$, simulating the typical Ldi/dt problem of scan delay test, in which circuit activities are idle during the long scan clock cycle.

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. Instead of skipping the pre-check procedure as in prior work, the procedure is implemented here by setting the transition count threshold to 0.1%, based on experience. For s38417, a transition count less than 0.1% means that there is only one transition on all input pins and scan chains. Experiments show that none of those vectors exceed even the tightest voltage drop constraint we have ever applied.

As in prior work, two kinds of constraints on power supply noise have been implemented. One is maximum worst-case voltage drop in any region, while the other is maximum percentage increase of path delay caused by power supply noise. 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 1 shows how the compaction results vary with constraint. The first column in Table 1 shows which type of constraint is applied. Column 2 is the percentage constraint. Column 3 is the number of vectors that exceeds the noise constraint prior to compaction and column 4 is the size of compacted test set excluding the original failed vectors. Column 5 lists α , the percentage increase in compacted test set size due to the noise constraint. Column 6 is the times that noise estimation procedure is skipped through pre-check, and column 7 is the total number of calls to the power noise estimation procedure during compaction. Column 8 lists the failure ratio β , the fraction of the times that a potential vector compaction exceeds the noise constraint. Column 9 is the fill-rate after compaction. The last 4 columns show running time. Column 10 is the total time spent on logic simulation, column 11 is the total time spent on noise estimation and column 12 shows total CPU time.

The last column shows the average CPU time per estimation call.

Generally, voltage drop or path delay constraints result in a larger compacted test set. A tighter constraint requires more estimation calls and more CPU time. Delay constraints further increase running time due to the need to estimate the delay of every path of the vector. We find that all experimental data are consistent with the conclusions we drew from our prior experiments. We also find that the tools have been speed up in both constraint cases. When constrained by maximum worst-case voltage drop, the CPU time is slightly shorter than before due to the effect of

pre-check procedure, which reduces the total number of estimation calls. Note that the number of skipped estimation calls by pre-check procedure hardly changes, mainly because the transition count threshold set in the experiment is quite tight, and most compaction will generate more than one transition and exceed the threshold. When path delay increase is constrained, the CPU time is greatly improved compared to prior work, since the time per estimation call is much shorter with optimized data structure. We can see from the table that the running time per estimation call is only slightly increased when path delay is calculated.

Table 1. Compaction results for s38417 with two types of restraint. The fill-rate of the un-compacted test set is 2.5%, with ucs = 13941 and s = 940.

Constraint Type	Constraint (%)	Originally Failed Vectors	Compacted Test Size	α (%)	Skipped Calls by Pre-check	Estimation Calls	β (%)	Compacted Fill-rate	Logic Simulation Time	Noise Estimation Time	CPU Time	Per Estimation Time (ms)
Max Worst-case Voltage Drop	3	1 265	1 168	158.8	2 797	544 141	95.8	0.107	7hr 16min	12hr 34min	12hr 34min	83.1
	4	610	1 020	73.4	2 798	187 620	87.5	0.153	2hr 39min	4hr 37min	4hr 37min	88.6
	5	139	947	15.5	2 797	48 294	50.3	0.221	39min	1hr 8min	1hr 8min	84.5
	7.5	0	940	0	2 798	24 148	0.02	0.250	19min	34min	34min	84.4
	10	0	940	0	2 798	24 144	0	0.250	19min	35min	35min	89.6
Max Delay Increase	3	916	958	99.4	2 920	276 906	91.7	0.132	3hr 50min	6hr 33min	6hr 34min	85.1
	5	265	947	28.9	2 841	129 810	81.6	0.198	1hr 48min	3hr 6min	3hr 7min	86.1
	7.5	86	937	8.8	2 803	49 763	51.7	0.231	41min	1hr 12min	1hr 12min	87.0
	10	17	938	1.6	2 800	38 109	36.7	0.247	32min	57min	57min	88.9
	15	0	940	0	2 798	24 145	0	0.250	20min	36min	36min	89.4

Table 2. Compaction results for s38417 when decoupling capacitance varies with maximum voltage drop constrained at 10%. The fill-rate of the un-compacted test set is 2.5%, with ucs = 13941 and s = 940.

γ	Originally Failed Vectors	Compacted Test Size	α (%)	Skipped Calls by Pre-check	Estimation Calls	β (%)	Compacted Fill-rate	Logic Simulation Time	Noise Estimation Time	CPU Time	Per Estimation Time (ms)
1.2	677	1 024	81.0	2 797	207 651	88.74	0.147	2hr 52min	4hr 56min	4hr 57min	85.7
1.5	198	956	22.8	2 798	63593	62.37	0.209	53min	1hr 32min	1hr 33min	86.7
2.3	0	940	0	2 798	24717	23.18	0.250	21min	37min	38min	91.0
3.1	0	940	0	2 798	24148	0.02	0.250	20min	36min	37min	89.8
3.9	0	940	0	2 798	24144	0	0.250	20min	36min	37min	89.8

Table 3. Delay of strictly robust test sets with either minimum-transition filling and random filling on ISCAS89 benchmarks.

Circuit	Test Set Size	Fill-rate (%)	Path Delay (ns)								
			Min			Max			Mean		
			Nominal Voltage	Random Fill	Min Transition	Nominal Voltage	Random Fill	Min Transition	Nominal Voltage	Random Fill	Min Transition
s1488	167	66.0	0.121	0.125	0.125	0.887	0.990	0.935	0.500	0.504	0.500
s38417	13941	2.53	0.044	0.045	0.044	1.41	1.45	1.42	0.682	0.704	0.687
s35932	8732	0.39	0.025	0.026	0.025	0.584	0.596	0.585	0.264	0.269	0.264

Table 2 shows the compaction result when decoupling capacitance varies. The ratio γ is defined in the second paragraph in this section. , Larger values of γ are obtained by increasing on-chip decoupling capacitance. It again proves that decoupling capacitor, which, in our model, is the main provider of charge at the early stage of the cycle, has a dominating impact on voltage drop and path delay.

6.2. Minimum Transition Filling and Random Fill

Pattern sets used in experiments have been introduced in the second paragraph of section 6.1. The patterns can be either randomly filled or minimum transition filled to generate high or low noise levels. Note that the care bit density of each vector is at most a few percent for most circuits, so there is a large difference in circuit activity between these two types of patterns. Thus, experiments are performed on the un-compacted pattern set to compare delay with two filling approaches. We do not use a compacted pattern set here, since the problem can be magnified with a low fill-rate.

Table 3 shows the comparison of experimental data with different filling methods. Column 1 lists the size of the test set on which power noise estimation is performed. The fill-rate of the test set is shown in column 3. Benchmark s1488 has a large fill rate since it is a small circuit with only 8 input pins and a short scan chain. Minimum transition fill is expected to achieve the lowest power supply noise, while random fill should achieve above-normal noise. The delay distributions under these two filling methods are analyzed and the sample statistics are presented in columns 4 to 12.

To compare the delay with high and low noise in a visual way, we plot the two distributions of delay for s38417 in Fig. 10. The figure, along with Table 3, shows that random filling generally produces significantly higher delay than minimum transition fill, due to the higher supply noise level.

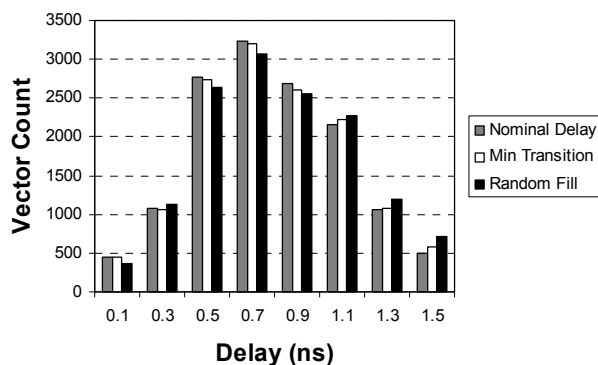


Fig. 10. Delay histogram with minimum transition filling and random filling on s38417.

7. Conclusions and Future Work

This paper presented a static compaction solution that reduces overkill induced by excessive power supply noise. It is based on our prior work with improvements in both models and algorithm performance. The experimental results demonstrate the promise of the algorithm, There is still much potential for speedup since the logic simulator used is relatively slow. The primary challenge is improving the accuracy of the power supply noise model while maintaining its low cost.

In the future, we plan to extend this work to dynamic compaction within the CodGen KLPG test generator [27]. We will also model noise during multi-cycle tests.

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 and 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] R. Ahmadi and F. N. Najm, "Timing Analysis in Presence of Power Supply and Ground Voltage Variations," IEEE/ACM Int'l Conf. on Computer Aided Design, San Jose, CA, Nov. 2003, pp. 176-183.
- [6] 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.
- [7] 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.
- [8] 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, Jun. 2003, pp. 756-769.
- [9] R. Panda, D. Blaauw, R. Chaudhry, V. Zolotov, B. Young and R. Ramaraju, "Model and Analysis of Combined Package and on-Chip Power Grid Simulation," Int'l Symp. On Low Power Electronics and Design, Rapallo, Italy, Jul. 2000, pp. 179-184.
- [10] 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.
- [11] S. Pant, D. Blaauw, V. Zolotov, S. Sundareswaran and R. Panda, "Vectorless Analysis of Supply Noise Induced Delay Variation,"

- IEEE/ACM Int'l Conf. on Computer Aided Design, San Jose, CA, Nov. 2003, pp. 184-191.
- [12] 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.
- [13] J. Wang, X. Lu, W. Qiu, Z. Yue, S. Fancler, W. Shi and D. M. H. Walker, "Static Compaction of Delay Tests Considering Power Supply Noise." IEEE VLSI Test Symposium, 2005. To appear.
- [14] J. Saxena, K. M. Butler, V. B. Jayaram, S. Kundu, N. V. Arvind, P. Sreepakash and 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.
- [15] 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.
- [16] M. Hashimoto, J. Yamaguchi and H. Onodera, "Timing Analysis Considering Spatial Power/Ground Level Variation," IEEE/ACM Int'l Conf. on Computer Aided Design, San Jose, CA, Nov. 2004, pp. 814-820.
- [17] S. R. Nassif and J. N. Kozhaya, "Fast Power Grid Simulation," ACM/IEEE Design Automation Conf., Los Angeles, CA, Jun. 2000, pp. 156-161.
- [18] H. Qian, S. R. Nassif and S. S. Sapatnekar, "Random Walk in a Supply Network," ACM/IEEE Design Automation Conf., Anaheim, CA, Jun. 2003, pp. 93-98.
- [19] Z. Zhu, B. Yao and C.-K. Cheng, "Power Network Analysis Using an Adaptive Algebraic Multigrid Approach," ACM/IEEE Design Automation Conf., Anaheim, CA, Jun. 2003, pp. 105-108.
- [20] L. Bisdounis and O. Koufopavlou, "Short-Circuit Energy Dissipation Modeling for Submicrometer CMOS Gates," IEEE Trans. Circuits and Systems I: Fundamental Theory and Applications, vol. 47, Issue 9, Sept. 2000, pp. 1350-1361.
- [21] W. G. Cochran and G. M. Cox, "Experimental Designs," John Wiley & Sons, Inc., 2nd edition, 1957.
- [22] S. R. Nassif and E. Acar, "Advanced Waveform Models for the Nanometer Regime," ACM/IEEE TAU Workshop, Austin, TX, Feb. 2004.
- [23] 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, Jun. 2001, pp. 295-300.
- [24] T. M. Niermann, R. K. Roy, J. H. Patel and J. A. Abraham, "Test Compaction for Sequential Circuits," IEEE Trans. Computer-Aided Design, vol. 11, no. 2, Feb. 1992, pp. 260-267.
- [25] I. Pomeranz and S. M. Reddy, "On Static Compaction of Test Sequences for Synchronous Sequential Circuits," ACM/IEEE Design Automation Conf., Jun. 1996, pp. 215-220.
- [26] I. Pomeranz and S. M. Reddy, "Improving the Efficiency of Static Compaction Based on Chronological Order Enumeration of Test Sequences," Test Symposium, Proceeding of the 11th Asian, Guam, USA, Nov. 2002, pp. 61-66.
- [27] 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.
- [28] 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.
- [29] 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.