Estimation of Maximum Power Supply Noise for Deep Sub-Micron Designs

Yi-Min Jiang, Kwang-Ting Cheng,

Dept. of Electrical & Computer Engineering
University of California
Santa Barbara, CA 93106
e-mail: {jym, timcheng}@bigbend.ece.ucsb.edu

An-Chang Deng

Synopsys, Inc.
700 East Middlefield Road
Mountain View, CA 94043
e-mail: deng@synopsys.com

Abstract

We propose a new technique for generating a small set of patterns to estimate the maximum power supply noise of deep sub-micron designs. We first build the charge/discharge current and output voltage waveform libraries for each cell, taking power and ground pin characteristics, the power net RC and other input characteristics as parameters. Based on the cells' current and voltage libraries, the power supply noise of a 2-vector sequence can be estimated efficiently by a cell-level waveform simulator. We then apply the Genetic Algorithm based on the efficient waveform simulator to generate a small set of patterns producing high power supply noise. Finally, the results are validated by simulating the obtained patterns using a transistor level simulator. Our experimental results show that the patterns generated by our approach produce a tight lower bound on the maximum power supply noise.

1. Introduction

Power supply noise due to switching current is becoming an important factor for deep sub-micron designs. The power supply noise reduces the actual voltage level reaching a device, which leads to increasing additional signal delay that may result in performance degradation. Moreover, excessive noise may cause logic and/or timing errors.

This paper focuses on the estimation of the maximum power supply noise which includes the inductive ΔI noise and *IR* voltage drop caused by the signal switching in internal circuitry as well as input and output buffers. To be able to observe switching at the signals, a two-vector sequence, $V = (v_1, v_2)$, has to be applied at the inputs of the combinational portion of the circuit. For a circuit with n primary inputs, this would require simulation of 4^n patterns for simulating all possible patterns at these inputs. This is practical only for circuits with a very small number of primary inputs. Also, the power supply noise is highly dependent on the instantaneous currents through all segments of the power and ground lines as well as the *RC* values of these power/ground segments. Accurate current waveforms flowing through all power/ground segments need to be derived efficiently during the process of the maximum power supply noise estimation.

We try to simulate a large number of patterns and use the *Genetic Algorithm* (GA) [4] to select/derive a small set of patterns that would cause high power supply noise. Because all power/ground segments' *RC*'s need to be considered in simulation to derive accurate power/ground segments' currents and voltages, circuit-level simulation will be unacceptably slow for this application due to the large number of simulation runs required. We therefore first derive comprehensive current/voltage waveform libraries for each cell (which can be repeatedly used by all designs based on these libraries). We then perform the simulation at the cell-level. We use an efficient event-driven waveform/logic simulator extended from [3] for current waveform simulation. Based on the waveform simulation results and the current/voltage waveform library, the current flowing through each cell with respect to a given pattern can be efficiently estimated. Note that for a segment in a power/ground net tree, the current waveform is not a direct superposition of the current waveforms of the cells downstream of the segment. In Sec. 5.2, we will discuss the related issues and discuss how to derive an estimated current waveform of a power/ground net segment based on the current waveforms of the cells.

Based on this simulation framework, we use GA to derive a small set of patterns. Finally, we can use a lower-level simulator to validate these patterns and identify the worst one among the selected set. This framework can be used to identify the patterns that would cause high power supply noise at any specified block in the chip, or the *IR* voltage drop at any given power supply segment. The difference for different blocks or different segments will be in the fitness functions used in the GA. Our experimental results show that our approach produces, on the average, 23% and 17% tighter lower bounds for the benchmark set, than the bounds obtained by the weighted random approach (which

uses random patterns with very high primary input transition probabilities) and the GA approach directly based on a transition level simulator, respectively. Also, the estimation time of our method is significantly faster.

2. Related Works

Recently, several approaches have been proposed for power supply noise [10][1][2] and maximum instantaneous current [8][9][13][7][6] estimation. Senthinathan and Prince [10] derived several closed-form equations to calculate simultaneous switching noise (SSN). Chang *et al.* [1] proposed a scaling model to estimate the ground bounce caused by the switching in internal circuitry for deep sub-micron circuits. Chen and Ling [2] proposed a simulated switching circuit model to estimate the power supply noise including the *IR* voltage drop and inductive ΔI noise based on an integrated package-level and chip-level power bus model. In these approaches, closed form equations or efficient model are used to estimate the power supply noise at each cell based on the given input conditions. However, since the power supply noise is strongly input pattern dependent, the accurate and efficient techniques for finding the maximum power supply noise in the entire circuit are needed.

For the maximum instantaneous current estimation, Kriplani *et al*. [8] present a pattern-independent algorithm called iMax algorithm to find an upper bound on the maximum instantaneous current. Several ATPG based techniques are proposed [9][13] to generate patterns causing high instantaneous current. Krstic *et al*. [9] also propose a probability-based algorithm to find these patterns. Two genetic-algorithm-based approaches for finding the lower bound for the maximum instantaneous current have been proposed [7][6]. These two approaches applies the genetic algorithm to identify patterns causing high instantaneous current through iteratively generating new patterns for simulation. The new patterns are generated using genetic operations, based on "good" patterns derived in the previous iterations.

3. Overview

The overall process of our technique for maximum power supply noise estimation is shown in Figure 1. We assume that the netlist and physical design are given. Our technique needs current/voltage waveform libraries for each cell, which is used by a waveform simulator. The libraries can be obtained by HSPICE simulation. The details of building these libraries are shown in Section 4. In the beginning, we extract the effective power/ground net *RC*'s for each block consisting of a small number of cells. We then generate the initial pattern population, and use a waveform simulator extended from [3] based on current/voltage waveform library to simulate each pattern. The details of the waveform simulation and the computation of the power supply noise based on the simulation results are described in Section 5. Throughout this paper, we will use the term *waveform simulator* to refer to the event-driven waveform/

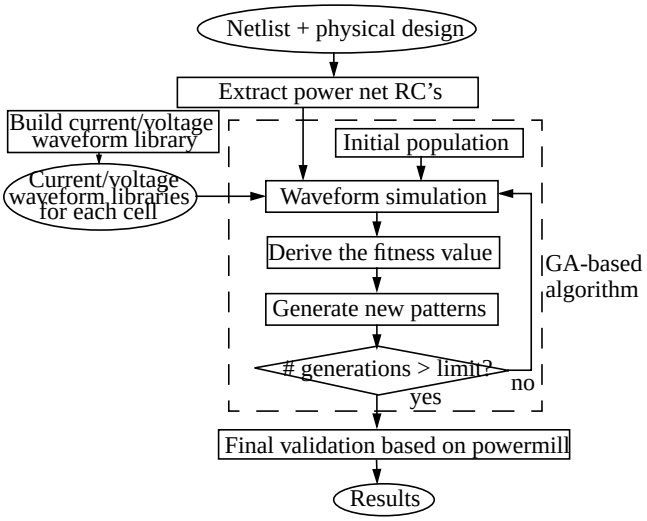

Figure 1: The flow of our technique for maximum power supply noise estimation.

logic simulator. We then generate the new patterns based on GA operations (selection, crossover and mutation). The fitness value of a pattern is simply the highest power supply noise at the target areas. Through iteratively generating new patterns for simulation, a small set of patterns is selected/ derived. Each pattern is then further validated using a transistor-level simulator. We use PowerMill [11] in our experiment.

4. Characterization: Building Current/Voltage Waveform Library

4.1 Circuit model for power supply noise

In our discussion, we assume that the topologies of the power and ground nets are single-pad-trees. However, our approach can be easily extended to handle multi-pad-tree and general graph topologies. For inductive ΔI noise we consider only the part caused by the change of instantaneous current on the package lead inductance and ignore the one from the wire/substrate inductance which is considerably smaller. Figure 2 shows a circuit model for each cell.

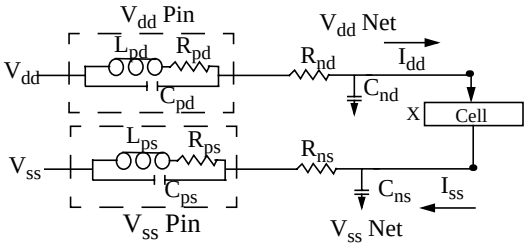

Figure 2: Circuit model for power supply noise caused by the switching in internal circuitry and I/O buffers.

The model is used to derive the current waveform flowing through each cell. We use V_{dd} and V_{ss} to denote power and ground, respectively. Each V_{dd} and V_{ss} pin is modeled by an RLC network (L_{pd}, R_{pd} and C_{pd} for V_{dd} pin, and L_{ps}, R_{ps} and C_{ps} for V_{ss} pin) as shown in Figure 2. The R_{nd} and C_{nd} (R_{ns}

and C_{ns}) correspond to the effective resistance and capacitance of V_{dd} (V_{ss}) line from the V_{dd} pin to the V_{dd} node of the cell (the V_{ss} node of the cell to the V_{ss} pin), respectively.

4.2 Building the current/voltage waveform library

Power supply noise at the V_{dd} and V_{ss} nodes of a cell can be computed by summing up the inductive ΔI noise and IR voltage drop along the series of power lines segments from the V_{dd} pin to the V_{dd} node of the cell (V_{ss} node of the cell to V_{ss} pin). We therefore need to derive accurate current waveforms for all segments of V_{dd} and V_{ss} lines and pins, which depend on the charge and discharge current waveforms of all cells. Based on the circuit model, for a given circuit with the netlist and its physical design, we first estimate the current waveform for each cell with respect to a given input pattern. The current waveforms for all the cells are then used to compute the current waveforms flowing through the V_{dd} and V_{ss} pins as well as all segments of V_{dd} and V_{ss} nets. The power supply noise on the V_{dd} and V_{ss} nodes of each cell can then be estimated based on these current waveforms.

The charge/discharge current and output voltage waveforms for a cell depend on various characteristics including the type of the cell, the starting/ending voltage and rising (falling) time of the input voltage waveform, loading capacitance of the cell, V_{dd} (V_{ss}) pin *RLC* values, and effective power/ground net *RC*'s (see Fig. 2). To characterize the current and output voltage waveforms, we build the current/voltage waveform libraries with indices including the above characteristics by using the circuit level simulator HSPICE. The ranges and intervals for all indices used in our libraries for a sample cell library are shown in Table 1. For a given

Table 1: The ranges and intervals for input indices in library.

	input voltage waveform			effective power net resistance	effective power net capacitance	loading capacitance
	starting voltage	ending voltage	rising (falling) time			
range	0 V - 3.3 V	0 V - 3.3 V	0.1 ns - 1.0 ns	1 Ω - 151 Ω	50 fF - 650 fF	100 fF - 1600 fF
interval	0.6 V	0.6 V	0.3 ns	30 Ω	200 fF	500 fF

package specification, the pin *RLC* values are fixed. To reduce the sizes of our libraries, we assume that only one input of a cell changes the value from low to high (or high to low), and the values of the other inputs are kept in their stable values such that the output of the cell switches and thus draws current. The input voltage waveform in our library is a ramp, which is characterized by three characteristics: starting voltage, ending voltage, and the slope (rising time or falling time). We sample the HSPICE results for output voltage and charge/discharge current waveforms with a fixed time step and store the discrete values of the waveforms in the library. These current/voltage waveform libraries are used to estimate the current waveform of each cell for a given input pattern applied at the primary input. To reduce the sizes of the waveform libraries, we build the comprehensive libraries only for cells with up to 4 inputs. For complex gates with more than 4 fanins, we apply various heuristics to reduce the number of entries. For example, we group all possible input patterns of the gate into several sets in such a way that each pattern in a set exhibits similar current. Then we perform HSPICE simulation for these sets to build the current/voltage waveform libraries. Note that all libraries are built only once, and can be repeatedly used for power supply noise estimation for all designs based on the same cell library.

5. Waveform Simulation

5.1 Deriving current waveforms flowing through blocks

Given a physical design, we first group the cells which are physically close to each other into small blocks and compute the effective power/ground *RC*'s for each block by using an *RC* reduction tool ULTIMA-PR [12]. Then we apply a waveform simulator extended from [3], which is based on the event-driven logic simulation algorithm, to simulate a given input pattern. The waveform simulator can handle the input voltage waveforms of the cells containing glitches as well as partial voltage swing, and produce the output voltage waveforms for all cells. The resulting voltage waveforms at all internal nodes along with the current/voltage waveform libraries we built can therefore be used to efficiently estimate the charge/discharge current for each internal cell and thus each block.

Note that the charge/discharge current for each block is derived with respect to the corresponding effective power/ground *RC* which are used to model the conducted path in the power/ground nets. In the following, we propose a technique to efficiently obtain the currents flowing through the V_{dd} and V_{ss} pins as well as each segment of power and ground lines based on the obtained current waveforms for all blocks.

5.2 Deriving current waveforms flowing through power/ground net segments

The current waveforms flowing through power and ground net segments depend on the charge/discharge currents of all blocks. However, even for a power supply net with a one-pad-tree topology, the current waveform of a segment is not a direct superposition of the current waveforms of the blocks downstream of the segment. If the duration of the current pulse is much smaller than the *RC* time constant from V_{dd} (V_{ss}) pin to the V_{dd} (V_{ss}) node of the block, not all charge (discharge) current is instantly coming from (to) V_{dd} (V_{ss}) pin. Part of the current is coming from (to) the neighboring capacitances along power and ground lines, and these capacitances will be charged up slightly later by the current from its neighboring capacitances again

and eventually by the external V_{dd} source. Therefore, the current waveforms in different segments along the path from V_{dd} pin to V_{dd} node (V_{ss} node to V_{ss} pin) of the block are different.

We conduct a simple experiment to illustrate this point. Consider the V_{dd} net of a circuit with a tree topology shown in Figure 3(a), where each node except the terminal nodes has 3 branches, and the total number of terminal nodes is 5,000. Suppose each terminal node corresponds to the V_{dd} node of a cell. The resistance and capacitance values in Figure 3(a) are extracted from the power supply net of a corresponding physical design. We assume just one block has switching current and all cells in other blocks are in stable values and do not draw any current. In this figure the charge current flowing through the switching block is modeled as a current source *i* at one of the terminal nodes. Figure 3(b) shows the waveforms for current *i* and the current in all segments (r_0 - r_6) along the path, which are derived by HSPICE. The vertical axis is the current (in μA) of each segment in the path, and the horizontal axis is the time (in *ns*). Note that the peak current through the root segment r_0 is much smaller than that the current source *i*, and the current waveforms in different segments are different. However, the segments closer to the switching block have waveforms similar to that of the current source.

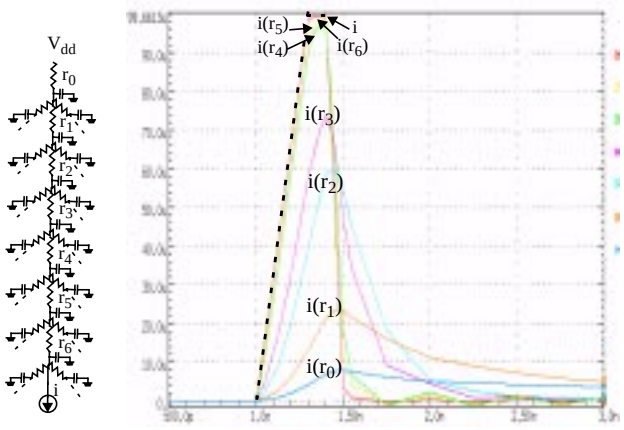

Figure 3: The current distribution for the charge current of a block with a small time period.

The current waveform in each segment along the power net (r_0 - r_6) is a function of the current waveforms flowing through the leaf cells and the *RC's* of all segments in the power net tree. It will be unacceptably slow to explicitly perform circuit level simulation on the circuit consisting of the power lines *RC* tree and current source of the block (like Fig. 3(a)) for each derived block current waveform and for each pattern. We therefore develop a "library" to speed up this process. We attempt to build a power segment waveform library for each block using the block current waveforms as parameter. In this phase, for each block, we first reduce the *RC*'s in the power net tree using the *RC* reduction tool [12], except the ones in the target path (In other words, we construct the effective power net tree like Fig. 3(a) for each block). We then assume a triangular current source, which is characterized by rising time, falling time and the peak value as parameters to approximate the charge/discharge current waveform of the block. Sample ranges and intervals for these indices used in this library constructing phase are shown in Table 2. Then we perform HSPICE simulation for each instance of the parameterized current sources and derive the current waveforms flowing through all segments in the path. We then sample these obtained current waveforms of all segments with a fixed time step, and store the sampled discrete waveforms in a power segment waveform library. The libraries for ground net are built in the same way.

After the power segment waveform libraries for all blocks are built, given the charge/discharge current waveform of a block with respect to an input pattern derived based on the procedure in Sec. 6, we approximate the waveform as a triangular form and then derive the corresponding current waveforms at the power net segments from the power segment waveform library. Note that the power segment waveform library are built only once for a given design, and can be repeatedly used to estimate the power supply noise for simulation of a large number of input patterns.

Table 2: The ranges and intervals for the indices in power segment waveform library.

	triangular current waveform		
	rising time (ns)	falling time (ns)	peak value (μA)
range	0.1 - 0.7	0.1 - 0.7	10 - 510
interval	0.2	0.2	100

For given input patterns, the estimation process for power supply noise is summarized as follows. First, we build the power segment waveform libraries, compute the effective power and ground net *RC's* for each small block (which consists of a set of adjacent cells), and apply the waveform simulator to simulate the input pattern to obtain the charge and discharge current waveforms for all blocks. Then, for each block, based on the block current waveform and the power segment waveform libraries, we derive the *effective* current waveforms flowing through all power net segments along the target path, and the effective current waveform flowing through power pins. The overall current waveform flowing through each power net segment can then be computed by summing up all the corresponding *effective* current waveforms for all blocks downstream of the target segment. The *IR* voltage drop for each block can be computed by summing up the *IR* voltage drops of all power net segments along the path. The inductive ΔI noise on V_{dd} (V_{ss})

pin can be obtained by evaluating the equation $L * di/dt$ where i is the total current waveform passing through the V_{dd} (V_{ss}) pin.

6. Genetic Algorithm for Pattern Generation

Based on this efficient framework for deriving the power supply noise for any given 2-vector sequence, we apply the Genetic Algorithm (GA) to generate a small set of patterns that would cause high power supply noise at a specified area. We use the tool described in [7] to generate such patterns. The vector generation process is based on the Genetic Algorithm (GA) [4] and is an iterative process. The iterative process stops after the number of simulated patterns reaches a limit and the tool reports a small number of patterns causing the highest power supply noise at the specified block(s).

7. Experimental Results

We use a 0.55 μm and 3.3 V CMOS library and a physical design system GARDS [5] to layout each benchmark circuit we used. After the physical design, we further use GARDS to extract the power/ground net RC-trees and compute all segments RC 's. For the V_{dd} and V_{ss} pins characteristics, we apply the values used in [1]. For generating all current/voltage waveform libraries and power segment waveform libraries, we perform the HSPICE simulation with level 3 model parameters for 0.55 μm feature size used in GARDS. We use 0.02 ns as a fixed time step to store the current/voltage waveform. Empirically, this value gives accurate estimation results for maximum power supply noise and reasonable sizes of current/voltage waveform libraries.

Estimation errors of power supply noise for a given input pattern

In order to evaluate the accuracy of the waveform simulator given in Section 5, we generate 100 random patterns (2-vector sequence) and perform simulation for each circuit by applying three different simulators: (1) the waveform simulator, (2) PowerMill and (3) HSPICE for the 7 largest ISCAS85 combinational benchmark circuits and 12 largest ISCAS89 sequential benchmark circuits. Table 3 show the results for the peak power supply noise for based on the three different simulators. Columns 2-4 show (1) the minimum, (2) the average and (3) the maximum power supply noise for the 100 patterns based on the waveform simulator. Columns 5-6, 7-8 show the average absolute errors ($|\Delta V|$) and the corresponding error percentage of our results and PowerMill results as compared with the noise derived by HSPICE simulation. For circuits C6288, C7552, s9234, s13207, s15850, s38417, s38584 and s35932, since HSPICE cannot give the results in a reasonable time, we do not compare the simulation values for both waveform simulator and PowerMill. The average power supply noise based on HSPICE is shown in Column 9. The average CPU times per simulation run for the three simulators are also reported.

Table 3: Power supply noise for 100 random patterns for benchmark circuits.

	Power supply noise estimation								Average CPU time (sec.)		
	Waveform simulator					PowrMill		HSPICE	Wave form simula-tor	Power Mill	HSPICE
Circuits	Min	Ave	Max	Ave		Ave		Ave			
	(V)	(V)	(V)	$\|\Delta V\|$	error (%)	ΔV	error (%)	(V)			
C1355	0.31	0.44	0.50	0.03	6.4	0	0	0.47	0.1	12.4	116
C1908	0.11	0.28	0.43	0.04	12.5	0	0	0.32	0.1	42.4	709
C2670	0.31	0.47	0.64	0.04	7.8	0.02	3.9	0.51	0.1	18.5	1042
C3540	0.20	0.31	0.44	0.02	6.9	0.02	6.8	0.29	1.6	31.9	1665
C5315	0.48	0.62	0.78	0.04	6.9	0.04	6.8	0.58	2.0	63.5	3559
s1196	0.28	0.34	0.42	0.04	13.3	0	0	0.30	0.1	3.8	2162
s1238	0.31	0.34	0.44	0.04	10.5	0.02	5.3	0.38	0.1	5.1	3065
s1423	0.54	0.62	0.71	0.05	8.7	0.02	3.5	0.57	0.1	6.5	461
s1488	0.34	0.39	0.47	0.04	11.4	0.02	5.7	0.35	0.1	8.8	2155
s1494	0.32	0.38	0.46	0.04	9.5	0.01	2.4	0.42	0.1	9.3	1666
s5378	0.41	0.63	0.77	0.05	8.6	0	0	0.58	0.2	23.6	5230
average	-	-	-	-	9.3	-	3.1	-	0.4	20.5	1985

As can be seen, on the average, the average estimation error of our method compared to HSPICE is 9.3%. The average estimation error for PowerMill is 3.1%. On the other hand, HSPICE simulation cannot estimate the power supply noise for large benchmark circuits in a reasonable time. Also, for the largest benchmark circuits we used s35932, PowerMill needs on average 686 seconds to simulate one pattern, and the waveform simulator needs only 5.6 seconds. The reasonable accuracy and the high efficiency of the waveform simulator make it possible to serve as the core of the GA-based test generation to explore the huge solution space for this application.

Estimation for maximum power supply noise for benchmark circuits

For each benchmark circuit, we apply the GA-based test generation process based on the waveform simulator to generate the input patterns and then select 10 patterns producing the highest power supply noise. The size of population of GA in the experiment is 30, and the number of generations is 50. That means the number of patterns simulated by the waveform simulator is 1,500. Then we simulate the obtained 10 patterns using PowerMill [11]. The reported maximum power supply noise is referred to as a tight lower bound of the maximum power supply noise. To evaluate our technique, we compare the results with those produced by two different test generation techniques. (1) Apply the same GA-based procedure but use PowerMill as the underlying simulator instead of the waveform simulator. Because of the higher simulation time per pattern, we reduce the number of total simulation runs to 300 for combinational circuits (the size of population is 10 and the number of generations is 30), and 150 simulation runs for sequential circuits (with the same size population and 15 generations). (2) Simply apply the same number of the weighted random patterns with primary input switching probability of 0.9, and use PowerMill to identify the one producing the highest power supply

noise.

The estimated maximum power supply noise for the benchmark circuits are shown in Table 4. The maximum

Table 4: Estimation for maximum power supply noise for benchmark circuits.

	Maximum Power Supply Noise						CPU time (min.)		
	Weighted Random		GA with PowerMill Only		Ours		Weighted Random	GA with Power Mill Only	Ours
	(V)	normal.	(V)	normal.	(V)	normal.			
C1355	0.42	1	0.53	1.26	0.68	1.62	62	63	23
C1908	0.64	1	0.67	1.05	0.68	1.06	212	213	34
C2670	0.70	1	0.70	1.00	0.79	1.13	93	95	31
C3540	0.69	1	0.72	1.04	0.79	1.15	160	164	65
C5315	0.72	1	0.81	1.13	0.92	1.28	318	321	71
C6288	0.91	1	1.02	1.12	1.15	1.26	760	767	155
C7552	0.91	1	0.88	0.97	1.13	1.24	785	789	132
s1196	0.66	1	0.66	1.00	0.66	1.00	10	10	9
s1238	0.63	1	0.63	1.00	0.75	1.19	13	13	13
s1423	0.51	1	0.53	1.04	0.62	1.22	16	16	16
s1488	0.57	1	0.55	0.96	0.61	1.07	22	22	21
s1494	0.58	1	0.61	1.05	0.62	1.69	23	23	21
s5378	0.69	1	0.78	1.13	0.84	1.22	59	60	31
s9234	0.62	1	0.67	1.08	0.82	1.32	48	50	23
s13207	0.81	1	0.77	0.95	0.86	1.06	203	207	60
s15850	0.72	1	0.66	0.92	0.83	1.15	737	742	131
s38417	0.72	1	0.75	1.04	0.86	1.19	1317	1321	225
s38584	0.85	1	0.95	1.12	1.10	1.29	1530	1532	247
s35932	1.06	1	1.16	1.09	1.25	1.18	1715	1721	276
average	-	1	-	1.05	-	1.23	425	427	83

supply noise and normalized values estimated by (1) weighted random approach, (2) GA based on PowerMill only, and (3) our approach are shown in Columns 2-3, 4-5 and 6-7, respectively. All the normalized values are with respect to the values derived by the weighted random approach. Note that all the values are validated by PowerMill. The CPU times for the three approaches are reported in Columns 8, 9 and 10, respectively. The CPU times shown in Column 10 include the CPU time for computing the effective power/ground nets *RC*'s for each block, the waveform simulation time for 1,500 patterns, the runtimes for *RC*-reduction and HSPICE simulation for building the power segment waveform libraries, and the PowerMill simulation time for the final 10 patterns.

The experimental results show that, on the average, our approach gives 23% and 17% tighter lower bounds for the benchmark set, than the bounds obtained with the weighted random approach and the PowerMill-based approach, respectively. For the CPU time of the largest benchmark circuits *s35932*, the weighted random and GA with PowerMill only approaches need 28.5 hours to estimate the maximum power supply noise, and our approach needs only 4.6 hours.

8. Discussion and Conclusions

As discussed in Section 5.2, if the duration of the charge (discharge) current pulse caused by a cell is much smaller than the *RC* time constant from V_{dd} (V_{ss}) pin to the V_{dd} (V_{ss}) node of the cell, not all charge (discharge) current is from (to) V_{dd} (V_{ss}) pin at the same time instance. The experiment shown in Figure 3 validates the observation. It implies that, to analyze the *IR* voltage drop caused by such a cell, we need to consider the currents only in the power and ground segments near the cell rather than the entire power and ground lines. In other words, it is possible that we may need to consider the "*local effect*" of the power and ground lines instead of "*global effect*". We are currently investigating the issues to reduce the CPU time for building the power segment waveform libraries.

We have proposed an efficient technique for generating patterns that would produce high power supply noise to achieve a tighter lower bound for maximum power supply noise estimation. The experimental results show that in comparison with the results obtained by other test generation schemes, the patterns generated using our approach result in much tighter lower bound on the maximum power supply noise. This method can be applied to generate patterns that would cause high power supply noise at any interested block.

References

[1] Y.-S. Chang, S. K. Gupta, and M. A. Breuer, "Analysis of Ground Bounce in Deep Sub-Micron Circuits," *Proceedings of 15th IEEE VLSI Test Symposium*, pp. 110-116, April, 1997.

[2] H. H. Chen and D. D. Ling, "Power Supply Noise Analysis Methodology for Deep-Submicron VLSI Chip Design," *Proceedings of Design Automation Conference*, pp. 638-643, June 1997.

[3] A.-C. Deng, Y.-C. Shiau, and K.-H. Loh, "Time Domain Current Waveform Simulation of CMOS Circuits," *Proceedings of ICCAD*, pp. 208-211, November 1988.

[4] D. E. Goldberg, R. Burch, Genetic Algorithms in Search, Optimization, and Machine Learning, Reading, MA: Addison-Wesley, 1989.

[5] GARDS, "Command Reference Manual," Volume 1-4, Silicon Valley Research, September 1996.

[6] M. S. Hsiao, E. M. Rudnick, and J. H. Patel, "K2: An Estimator for Peak Sustainable Power of VLSI Circuits," *International Symposium on Low-Power Electronics and Design (ISLPED)*, pp. 178-183, August, 1997.

[7] Y.-M. Jiang, K.-T. Cheng, and A. Krstic, "Estimation of Maximum Power and Instantaneous Current Using a Genetic Algorithm," *Proc. of IEEE Custom Integrated Circuits Conference*, pp. 135-138, May 1997.

[8] H. Kriplani, F. N. Najm, and I. N. Hajj, "Pattern Independent Maximum Current Estimation in Power and Ground Buses of CMOS VLSI Circuits: Algorithms, Signal Correlations, and Their Resolution," *IEEE Transactions on CAD*, pp. 998-1012, August 1995.

[9] A. Krstic and K.-T. Cheng, "Vector Generation for Maximum Instantaneous Current Through Supply Lines for CMOS Circuits," *Proc. of Design Automation Conference*, pp. 383-388, June 1997.

[10] R. Senthinathan and J. L. Prince, *Simultaneous Switching Noise of CMOS Devices and Systems*, Kluwer Academic Publishers, 1994.

[11] SYNOPSYS, "PowerMill Reference Manual", August 1997.

[12] Ultima Interconnect Technology, Inc., "Ultima-PR User's Guide 2.2.6", 1997.

[13] C.-Y. Wang and K. Roy, "Estimation of Maximum Power for Sequential Circuits Considering Spurious Transitions," *Proc. of ICCD*, pp. 746-751, 1997.