Dual-Transition Glitch Filtering in Probabilistic Waveform Power Estimation

Fei Hu
Dept. of Electrical and Computer Engineering
Auburn University, AL 36849
hufei01@auburn.edu

Vishwani D. Agrawal
Dept. of Electrical and Computer Engineering
Auburn University, AL 36849
vagrawal@eng.auburn.edu

ABSTRACT
Existing gate-level probabilistic approaches to power estimation fail to accurately model the glitch filtering by inertial delays. This effect has an impact on the power dissipation of a circuit and should not be neglected, especially for dynamic power estimation of circuits with dynamic power optimization. We propose a new glitch filtering analysis using the dual-transition probability that captures the states of a node at two different time instances. Experiments show that probabilistic simulation and the tagged probability simulation (TPS) techniques, when enhanced by the dual-transition analysis, provide more consistent power estimation. For circuits with a large component of glitch power, up to 29% improvement in the estimation accuracy is obtained.

Categories and Subject Descriptors
I.6.5 [**Simulation and Modeling**]: Model development – *Modeling methodologies*

General Terms
Algorithms, Design, Performance, Verification

Keywords
Dual-transition probability, glitch filtering, probabilistic waveform simulation, dynamic power estimation.

1. INTRODUCTION
Accurate power estimation methods are critical to IC designs. For gate level power estimation, the major power component is the switching power dissipated when signal transitions charge and discharge load capacitances [1]. Although leakage power is becoming more significant as device sizes shrink, an accurate estimation of switching power is always of great interest.

Most of the earlier probabilistic approaches for gate level switching power estimation [2],[4],[10],[8],[9] assume a zero delay model, thus neglecting all signal changes in the transient interval, which are known as glitches or hazards. The zero-delay techniques have been extended [11] for the real delay case to include the glitch power. The *transition density* approach [7] includes the glitch power in the estimation but makes the assumption that no two signals make simultaneous transitions. In probabilistic simulation (CREST) [5],[6] and the tagged probabilistic simulation (TPS) [12],[3], *probability waveforms* are used to statistically model the signal activity including glitches. A major shortcoming of these techniques is that the *glitch filtering effect* is seldom considered. Glitch filtering effect refers to the fact that glitches with pulse width less than the gate inertial delay will be "filtered" out by the gate. Glitch filtering effect can therefore dramatically change the switching activity of gates. An accurate modeling of this effect is essential in power estimation, especially for applications concerning dynamic power optimization.

In this paper, we propose an improved glitch filtering method in probabilistic simulation based on a new measure, the dual-transition probability, which captures the states of a node at two different time instances. Our technique can be applied to both probabilistic simulation (CREST) [5],[6] and TPS [12],[3]. Experimental results show that the enhanced TPS achieves a more accurate and consistent power estimation than the original one. For certain circuits with a large component of glitch power, up to 29% improvement on estimation accuracy of total power is obtained.

2. BACKGROUND AND DEFINATIONS

2.1 Probability Waveform
A probability waveform w is a sequence of *signal probabilities* and *transition probabilities* over the *signal transition interval* [6]. Signal probability $sp_n(t)$ is defined as the probability of node n having logic 1 at time t. Transition probability is the probability of a signal transition at the node. We define transition probability $P_n^s(t)$ as the probability that node n has a logic transition state $s \in \{00,01,10,11\}$ at time t, $sp_n(t-) = P_n^{10}(t) + P_n^{11}(t)$, and $sp_n(t+) = P_n^{01}(t) + P_n^{11}(t)$. The average switching power of a node is calculated as

$$P_{av}(n) = \frac{1}{2} f_{clk} V_{dd}^2 C_n \sum_i (P_n^{01}(t_i) + P_n^{10}(t_i)) \quad (1)$$

where f_{clk} is the clock frequency, V_{dd} is the supply voltage, C_n is the load capacitance of node n and $\sum_i P_n^{01}(t_i) + P_n^{10}(t_i)$ is the transition density [7] of the node.

A tagged probability waveform [3] can be viewed as a partitioning of a probability waveform according to the initial and final steady state signal values. Four tagged probability waveform are defined for node n: $w_n^{00}, w_n^{01}, w_n^{10}$, and w_n^{11}, where 00, 01, 10, 11 are called the tags. *Tagged waveform probability,* denoted as $P(w_n^{xy})$, is the sum of occurrence probabilities of all logic waveforms represented by w_n^{xy}. Let $sp_n^{xy}(t)$, where $x, y \in \{0,1\}$), represent the signal probability in tagged probability waveform w_n^{xy}, and $P_{n,xy}^s(t)$ be the probability that node n has a logic transition state

GLSVLSI'05, April 17–19, 2005, Chicago, Illinois, USA.

$s \in \{00,01,10,11\}$ at time t in tagged probability waveform w_n^{xy}, then the switching power dissipation of the node is calculated as

$$P_{av}(n) = \frac{1}{2} f_{clk} V_{dd}^2 C_n \sum_i \sum_x \sum_y (P_{n,xy}^{01}(t_i) + P_{n,xy}^{10}(t_i)) \quad (2)$$

where $\sum_i \sum_x \sum_y (P_{n,xy}^{01}(t_i) + P_{n,xy}^{10}(t_i))$ is the transition density of the node.

2.2 Propagation of Waveform in TPS

In TPS, the *macroscopic spatial correlations* between steady state signal values (tags) are used to approximate the exact spatial correlations. The macroscopic correlation coefficient for tagged waveforms, w_a^{xy}, w_b^{wz}, is defined as

$$\omega_{a,b}^{xy,wz} = \frac{P(w_a^{xy} \wedge w_b^{wz})}{P(w_a^{xy}) P(w_b^{wz})} \quad (3)$$

where $x, y, w, z \in \{0,1\}$. The correlation coefficient is the ratio of the joint probability of two tagged waveforms and the product of the two individual tagged waveform probabilities.

Under the assumption that all correlations between any two states of different inputs at the same time instance are the same as $\omega_{a,b}^{xy,wz}$, the transition probabilities for a two input AND gate are calculated as

$$P_{c,(xy,wz)}^{01}(t+d) = (P_{a,xy}^{01}(t) P_{b,wz}^{11}(t) + P_{a,xy}^{01}(t) P_{b,wz}^{01}(t) + P_{a,xy}^{11}(t) P_{b,wz}^{01}(t)) \omega_{a,b}^{xy,wz} \quad (4)$$

$$P_{c,(xy,wz)}^{10}(t+d) = (P_{a,xy}^{10}(t) P_{b,wz}^{11}(t) + P_{a,xy}^{10}(t) P_{b,wz}^{10}(t) + P_{a,xy}^{11}(t) P_{b,wz}^{10}(t)) \omega_{a,b}^{xy,wz} \quad (5)$$

The left hand side of each equation is the *joint-tagged* (by *xy,wz*) transition probability. All joint-tagged waveforms with the same output tag are combined together to form four tagged waveforms.

3. NEW GLITCH FILTERING METHOD

3.1 Original Glitch Filtering in TPS

The glitch filtering effect refers to subtraction of transition probabilities from probability waveforms. In TPS [3], glitch filtering on joint-tagged waveform $w_c^{xy,wz}$ of a two input AND gate, *c = ab,* with inertial delay *d* can be described as follows. For a rising transition event $tu_a^{xy}(t_1)$, all falling transition events $tu_b^{wz}(t_2)$ that come from the other gate input with $t_1 < t_2 < t_1 + d$ are subject to glitch filter. It means that $\omega_{a,b}^{xy,wz} P_{a,xy}^{01}(t_1) P_{b,wz}^{10}(t_2)$ is subtracted from both $P_{c,(xy,wz)}^{01}(t_1 + d)$ and $P_{c,(xy,wz)}^{10}(t_2 + d)$. In essence, authors assume that $P_{a,xy}^{01}(t_1)$ and $P_{b,wz}^{10}(t_2)$ are also correlated by macroscopic correlations described by $\omega_{a,b}^{xy,wz}$, and that $\omega_{a,b}^{xy,wz} P_{a,xy}^{01}(t_1) P_{b,wz}^{10}(t_2)$ represents the probability that both rising and falling transitions occur at the output *c* at times t_1 and t_2. This process is applied to each joint-tagged probability waveform before it is merged into four output tagged probability waveform.

3.2 Dual-Transition Probability

The above method for glitch filtering is imprecise. First, the glitches coming from the single input can not be filtered by this method. Second, even if we maintain the same assumption for the use of macroscopic correlation, the probability that both rising and falling transitions occur at t_1 and t_2 is not simply $\omega_{a,b}^{xy,wz} P_{a,xy}^{01}(t_1) P_{b,wz}^{10}(t_2)$. A signal could have four possible transition states at any given time instance, that is a rising transition (01), a falling transition (10), holding on one (11), or holding on zero (00). For a two-input AND gate, to have a rising transition at the output at time instance $t_1 + d$, inputs *(a, b)* at t_1, should be (01, 01), (01, 11) or (11, 01). Similarly, to have a falling transition at the output at $t_2 + d$, inputs *(a, b)* at t_2 should be (10, 10), (10, 11) or (11, 10). To derive the probability that both transitions occur at t_1 and t_2 all nine possible cases of input states (at t_1 and t_2) must be considered.

For our new method of glitch filtering, we define *dual-transition probability* $P_{n,xy}^{sn1,sn2}(t_1, t_2)$ as the probability of a joint event that node *n* has the state *sn1* at time t_1 and the state *sn2* at t_2 on the *xy* tagged probability waveform, where $sn1, sn2 \in \{00,01,10,11\}$ and $x, y \in \{0,1\}$. Then, the probability that both rising and falling transitions occur at t_1 and t_2 at the output of a two input AND gate can be calculated as

$$P_{c,(xy,wz)}^{01,10}(t_1 + d, t_2 + d) = \sum_{i=1}^{3} \sum_{j=1}^{3} P_{a,xy}^{sa1_i, sa2_j}(t_1, t_2) P_{b,wz}^{sb1_i, sb2_j}(t_1, t_2) \omega_{a,b}^{xy,wz} \quad (6)$$

where *c* is the output, *a* and *b* are the inputs, *d* is the gate inertial delay, $x, y, w, z \in \{0,1\}$, $(sa1_i, sb1_i) \in \{(01,11), (11,01), (01,01)\}$, and $(sa2_j, sb2_j) \in \{(10,11), (11,10), (10,10)\}$. The macroscopic correlation coefficient $\omega_{a,b}^{xy,wz}$ is still adopted to approximate the spatial correlations between two inputs. The output dual-transition probability is joint-tagged with (*xy,wz*). Similar to the merging of joint-tagged transition probabilities, joint-tagged dual-transition probabilities are added together according to their output tags to form a (single tagged) dual-transition probability.

3.3 New Glitch Filtering Method

With the dual-transition probability, the glitch filtering on joint-tagged waveform $w_c^{xy,wz}$ of a two input AND gate with inertial delay *d* can be described as follows. For a rising transition event at t_1, all falling transition events at t_2, $t_1 < t_2 < t_1 + d$, are subject to glitch filter. It means that the dual-transition probability $P_{c,(xy,wz)}^{01,10}(t_1, t_2)$ is subtracted from both $P_{c,(xy,wz)}^{01}(t_1 + d)$ and $P_{c,(xy,wz)}^{10}(t_2 + d)$. Furthermore, for a falling transition event at t_1, all rising transition events at t_2 for which $t_1 < t_2 < t_1 + d$ are subject to glitch filter. The dual-transition probability $P_{c,(xy,wz)}^{10,01}(t_1, t_2)$ is subtracted from $P_{c,(xy,wz)}^{10}(t_1 + d)$ and $P_{c,(xy,wz)}^{01}(t_2 + d)$.

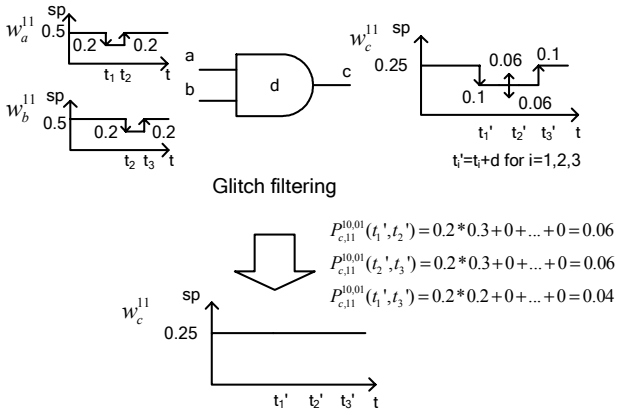

Figure 1. Propagation of probability waveform (*d* is the gate delay, assuming $t_1 < t_2 < t_3 < t_1 + d$).

This process is applied to each joint-tagged probability waveform before they are merged into four output tagged probability waveforms. Figure 1 shows an example of the process.

Note that our new glitch filtering method is not an exact method in the sense that we ignored the higher order joint probabilities, i.e., probabilities of three or more transitions occurring simultaneously. To avoid the exponential increase of computation cost by propagation of such higher order joint probabilities, we only consider pairwise dual-transition probabilities. Experiments show that this method provides a good approximation for glitch filtering.

3.4 Dual-Transition Probability Propagation

3.4.1 Calculating Dual-Transition Probability

Dual-transition probability is propagated from the primary inputs toward the outputs of the circuit. The general equation for propagation of dual-transition probability is as follows,

$$P^{sc1,sc2}_{c,(xy,wz)}(t_1+d,t_2+d)=\sum_{i=1}^{k}\sum_{j=1}^{l}P^{sa1_i,sa2_j}_{a,xy}(t_1,t_2)P^{sb1_i,sb2_j}_{b,wz}(t_1,t_2)\omega^{xy,wz}_{a,b} \quad (7)$$

where $x,y,w,z\in\{0,1\}$, $(sa1_i,sb1_i)\in Q1$, and $(sa2_j,sb2_j)\in Q2$. *Q1*(or *Q2*) represents the group of input combinations on *a* and *b* that give output state *sc1* (or *sc2*). The size of *Q1* (or *Q2*) is *k* (or *l*). For primary inputs, where transitions can only occur at time zero, $P^{sn1,sn2}_{n,xy}(0,t)=P^{sn1}_{n,xy}(0)$ if *sn1*, *sn2* is a valid sequence of states on *n*. Otherwise, $P^{sn1,sn2}_{n,xy}(0,t)=0$. For example, $P^{01,11}_{n,xy}(0,t)=P^{01}_{n,xy}(0)$, while $P^{01,01}_{n,xy}(0,t)=0$ because it is impossible to have two rising transitions at times *0* and *t* at PI.

3.4.2 Updating Probabilities after Glitch Filtering

Clearly, no two transitions can occur within the gate delay after the glitch filtering. Therefore, dual-transition probabilities need to be updated accordingly after the glitch filtering described in Section 3.3 is done. Defining $sn[1], sn[2]$ as the first (left) and second (right) bit of *sn*, the dual-transition probability is updated as follows. For t_1, t_2 with $t_1<t_2<t_1+d$, $P^{sn1,sn2}_{n,xy}(t_1,t_2)=P^{sn1}_{n,xy}(t_1)$ (or $=P^{sn2}_{n,xy}(t_2)$) if $sn2[1]=sn2[2]=sn1[2]$ (or $sn1[1]=sn1[2]=sn2[1]$). Otherwise, $P^{sn1,sn2}_{n,xy}(t_1,t_2)=0$.

To update dual-transition probability for t_1, t_2 with $t_1+d<t_2$, we define *dual-transition correlation coefficient* $\omega^{sn1,sn2}_{n,xy}(t_1,t_2)$ as,

$$\omega^{sn1,sn2}_{n,xy}(t_1,t_2)=\frac{P^{sn1,sn2}_{n,xy}(t_1,t_2)}{P^{sn1}_{n,xy}(t_1)P^{sn2}_{n,xy}(t_2)} \quad (8)$$

where $sn1,sn2\in\{00,01,10,11\}$, $x,y\in\{0,1\}$. $P^{sn1,sn2}_{n,xy}(t_1,t_2)$ is the dual-transition probability and $P^{sn1}_{n,xy}(t_1)$, $P^{sn2}_{n,xy}(t_2)$ is the transition probability on tagged waveform. $\omega^{sn1,sn2}_{n,xy}(t_1,t_2)$'s are calculated before the glitch filtering is applied. After the glitch filtering, dual-transition probability is updated as follows,

$$P^{sn1,sn2}_{n,xy}(t_1,t_2)=P^{sn1}_{n,xy}(t_1)P^{sn2}_{n,xy}(t_2)\omega^{sn1,sn2}_{n,xy}(t_1,t_2) \quad (9)$$

Note that $P^{sn1}_{n,xy}(t_1), P^{sn2}_{n,xy}(t_2)$ in the above equation are transition probabilities after the glitch filtering (subtractions). Essentially, we assume that correlations of transitions at t_1, t_2 with $t_1+d<t_2$ do not change due to the glitch filtering. Experiments show our method performs very well under this assumption. To minimize the computation, only correlation coefficients (and dual-transition probabilities) with t_2-t_1<d_{max} are propagated, where d_{max} is the maximum gate delay in the circuit.

4. EXPERIMENTAL RESULTS

Our glitch filtering method can be applied to both probabilistic simulation and TPS. The proposed algorithm has been implemented in C++ as stand-alone software with both of above approaches, which are referred to as "ProSim+DT" and "TPS+DT", respectively. The software takes the description of a circuit (netlist and input signal probabilities) and gives an estimation of the switching activity (power) at each node. In our experiments, spatial and temporal independences are assumed for primary inputs. Signal probabilities for PIs are assumed as 0.5 and logic simulations are done with 40,000 random input vectors (same signal probability). As in other techniques [3], the steady state transition probabilities and macroscopic correlations are obtained from a zero delay logic simulator.

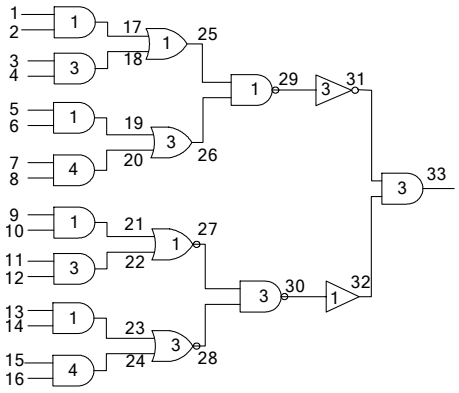

Figure 2. A simple tree structure circuit. Numbers on each gate indicate the arbitrarily selected gate delays.

4.1 Estimation Accuracy

Our first experiment is done for simple tree structure circuit as shown in Figure 2. The power estimates (in terms of transition density D_x) for nodes with maximum error are compared against those obtained from logic simulation in Table 1. The percentage error is the ratio between the absolute error and the node power given by logic simulator. The original probabilistic simulation (ProSim) gives larger error because it neglects glitch filtering at nodes 26, 28, 30 and 31. Original TPS gives a larger error at nodes 30 and 31 because of inaccurate glitch filtering. For a tree structure circuit where no spatial correlation exists, our application to both ProSim and TPS has the similar accuracy.

More experiments have been conducted on ISCAS'85 benchmark circuits. Delay of each gate is assigned proportional to the number of fanouts. For each circuit, average node error (E_{avg}), standard deviation (σ) of the node error, and the error in total power (E_{tot}) are measured. Node errors are percentage errors with respect to the average node power obtained from logic simulation. Results are shown in Table 2, where those of original probabilistic simulation are not included due to the space limit. The errors for those would be much larger than that of ProSim+DT because of the lack of glitch filtering.

For benchmark circuits, where reconvergent fanouts exist, ProSim+DT gives a larger error than TPS in most cases because it neglects spatial correlations. For TPS+DT, the improvement of estimation accuracy over TPS is significant for certain circuits, e.g., c432, c1355 and c6288. Up to 29% improvement on E_{tot} is obtained. These circuits typically have a large component of glitch

Table 1. Node by node switching power comparisons for the simple tree structure circuit.

Nd.	Logic Sim.	ProSim		ProSim+DT		TPS		TPS+DT	
	P_{switch} (D_x)	P_{switch} (D_x)	*Err.* (%)	P_{switch} (D_x)	*Err.* (%)	P_{switch} (D_x)	*Err.* (%)	P_{switch} (D_x)	*Err.* (%)
26	0.491	0.563	14.6	0.492	0.3	0.492	0.3	0.492	0.3
27	0.564	0.563	0.2	0.563	0.2	0.564	0.1	0.564	0.1
28	0.489	0.563	15.0	0.492	0.6	0.493	0.7	0.493	0.7
29	0.434	0.453	4.4	0.432	0.4	0.434	0.0	0.434	0.0
30	0.450	0.593	31.8	0.455	1.0	0.483	7.2	0.453	0.6
31	0.351	0.453	29.1	0.339	3.2	0.434	23.8	0.346	1.2
32	0.450	0.593	31.8	0.455	1.0	0.483	7.2	0.453	0.6
33	0.263	0.375	42.4	0.256	2.8	0.306	16.4	0.263	0.2

power. The estimation accuracy is significantly improved due to our new glitch filtering method.

From Table 2, we can also find that TPS+DT gives a more consistent estimation over different circuits in terms of average and maximum errors.

Table 2. Estimation error comparison for benchmark circuits.

Circuits	ProSim+DT			TPS			TPS+DT		
	E_{avg} (%)	σ(%)	E_{tot} (%)	E_{avg} (%)	σ(%)	E_{tot} (%)	E_{avg} (%)	σ(%)	E_{tot} (%)
c17	5.8	7.8	0.7	2.3	2.6	0.1	2.3	2.6	0.1
c432	14.7	17.3	8.5	29.9	38.8	35.8	9.5	11.8	6.5
c499	6.2	11.6	6.6	6.8	14.0	7.0	3.6	8.2	0.6
c880	11.2	18.3	7.3	8.3	15.3	1.6	8.0	15.7	5.2
c1355	16.8	21.5	18.3	24.2	31.6	32.9	5.8	11.2	5.4
c1908	21.9	33.8	19.7	15.0	23.1	4.1	17.7	27.9	11.2
c2670	20.6	29.7	15.0	16.6	29.8	7.2	16.7	28.3	9.9
c3540	16.6	36.3	10.0	13.8	26.3	9.8	10.3	25.6	2.4
c5315	20.2	40.1	17.2	11.8	24.4	2.3	13.4	31.5	10.1
c6288	29.6	29.9	26.4	27.4	27.5	32.1	15.7	18.8	4.1
c7552	21.6	39.9	16.4	14.5	27.5	3.2	14.8	31.4	7.8
Avg.	16.8	26.0	13.3	15.5	23.7	12.4	10.7	19.4	5.7
Max.	29.6	40.1	26.4	29.9	38.8	35.8	17.7	31.5	11.2

We observe that TPS+DT gives the same or larger error than original TPS for certain circuits. This is due to the fact that estimation accuracy is jointly decided by the TPS and our glitch filtering method. The effectiveness of our new glitch filtering method is limited by the inherent errors in the TPS technique.

4.2 Computation Cost

For our implementation, the computation speed of TPS+DT is about 2-3 times faster than that of the logic simulation of entire vector sequence, while ProSim+DT is about 20-30 times faster and the original TPS is about two orders of magnitude faster than logic simulation. The computation complexity of TPS+DT is much higher because of the propagation of dual-transition probabilities. Since there is only one probability waveform for ProSim+DT, the computation cost of ProSim+DT is relatively lower and could be the best choice for tree structured circuits, where no spatial correlation exists among the signals.

5. CONCLUSIONS

Most existing gate-level probabilistic approaches for power estimation fail to accurately model the glitch filtering effect. However, this effect has a non-negligible impact on the power consumption of a circuit. In this paper, we propose an improved glitch filtering method using a new concept of dual-transition probability. Our method can be applied to both probabilistic simulation and TPS techniques. The enhanced TPS achieves a more accurate and consistent estimation than the original TPS. For certain circuits with a large glitch power, up to 29% improvement in estimation accuracy is obtained.

6. REFERENCES

[1] Chandrakasan, A. P. and Brodersen, R. W. *Low power digital CMOS design*. Kluwer academic publishers, Boston, 1995.

[2] Cirit, M. A. Estimating dynamic power consumption of CMOS circuits. In *Proc. ICCAD*, 1987, 534–537.

[3] Ding, C.-S., Tsui,C.-Y., and Pedram, M. Gate-level power estimation using tagged probabilistic simulation. *IEEE Trans. on CAD, 17*, 11 (Nov. 1998), 1099–1107.

[4] Ercolani, S., Favalli, M., Damiani, M., Olivo, P., and Ricco, B. Estimate of signal probability in combinational logic networks. In *Proc. of First European Test Conference*, 1989, 132–138.

[5] Najm, F. N., Burch, R., Yang, P., and Hajj, I. N. CREST - a current estimator for CMOS circuits. In *Proc. ICCAD*, Nov. 1988, 204–207.

[6] Najm, F. N., Burch, R., Yang, P., and Hajj, I. N. Probabilistic simulation for reliability analysis of CMOS VLSI circuits. *IEEE Trans. on CAD, 9*, 4 (Apr. 1990), 439–450.

[7] Najm, F. N. Transition density: a new measure of activity in digital circuits. *IEEE Trans. on CAD*, *12*, 2 (Feb. 1993), 310–323.

[8] Marculescu, R., Marculescu, D., and Pedram, M. Logic level power estimation considering spatiotemporal correlations. In *Proc. ICCAD*, Nov. 1994, 294–299.

[9] Marculescu, R., Marculescu, D., and Pedram, M. Efficient power estimation for highly correlated input streams. In *Proc. 32nd DAC*, June 1995, 628–634.

[10] Schneider, P. and Schlichtmann, U. Decomposition of Boolean functions for low power based on a new power estimation technique. In *Proc. International Workshop on Low Power Design*, Apr. 1994, 123–128.

[11] Theodoridis, G., Theoharis, S., Soudris, D., Stouraitis, T., and Goutis, C. An efficient probabilistic method for logic circuits using real delay gate model. In *Proc. ISCAS, vol. 1*, Jun. 1999, 286–289.

[12] Tsui, C.-Y., Pedram, M., and Despain, A. M. Efficient estimation of dynamic power consumption under a real delay model. In *Proc. ICCAD*, Nov. 1993, 224–228.