Logic Level Power Estimation Considering Spatiotemporal Correlations

Radu Marculescu, Diana Marculescu
and Massoud Pedram

CENG Technical Report 94-05

Department of Electrical Engineering - Systems
University of Southern California
Los Angeles, California 90089-2562
(213)740-4458

April 1994

Logic Level Power Estimation Considering Spatiotemporal Correlations

Radu Marculescu, Diana Marculescu and Massoud Pedram

Department of Electrical Engineering - Systems

University of Southern California

Los Angeles, CA90089-2561

Phone: (213)-740-9079

e-mail: radum@poisson.usc.edu

April 1994

Logic Level Power Estimation Considering Spatiotemporal Correlations

Abstract

Switching activity estimation in combinational circuits is addressed from a probabilistic point of view. The zero-delay hypothesis is considered and under pseudorandom or biased input sequences, the activities at the primary outputs and all internal nodes are estimated. Work by previous researchers is extended to manage complex spatio-temporal correlations by using lag-one Markov Chains and conditional probabilities. Evaluations of the model and a comparative analysis presented for benchmark circuits demonstrates the accuracy and the practicality of the method. The results presented in this paper are useful in power estimation and low power design.

1. Introduction

In solving the complex problem of power estimation for digital circuits, knowledge about the average switching activity in a circuit plays a significant part. Indeed, to compute the power dissipation, most of the current models rely on the switching activity information about the circuit. The accuracy in making such estimations has become an important objective by itself. The key issue is to account for various dependencies, irrespective of the particular way in which the inputs and the target circuits are described.

Most of the existing work in pseudorandom testing and power estimation relies on probabilistic methods and signal probability calculations. [1] presents one of the earliest work in computing the signal probabilities in a combinational network. The authors associate variable names with each of the circuit inputs representing the the signal probabilities of these inputs and then, for each internal circuit line, they compute algebraic expressions involving these variables. These expressions represent the signal probabilities for these lines. While the algorithm is simple and general, its worse case time complexity is exponential.

For tree circuits which consists of simple gates, the exact signal probabilities can be computed during a single post-order traversal of the network [2]. Alternatively, one may use a graph-based algorithm to compute the exact values of signal probabilities using Shannon's expansion [3]. This algorithm relies on the notion of the supergate of a node and identifies the set of maximal supergates in order to calculate the signal probabilities. In the worst-case, this algorithm becomes equivalent to an exhaustive true-value simulation.

Common digital circuits exhibit a lot of dependencies; by far, the most known one is the dependency due to reconvergent fan-out among different signal lines, but even structurally independent lines may have dependencies (induced by the sequence of inputs applied to the circuit) which cannot be neglected. Accounting for all kinds of dependencies is impossible even for small circuits; consequently, for real-size circuits, only some of the dependencies have been considered and even then, only heuristics have been proposed.The main reason behind this situation is the difficulty in managing complex data dependencies at acceptable levels of computational work. [4] provides an extension to [2] called the weighted averaging algorithm; this approach attempts to take into account the first order effects of reconvergent fanout stems in the variable support of the node. It is linear in the product of the number of circuit inputs and the size of the circuit. [5] gives an algorithm, known as the cutting algorithm, which computes lower and upper bounds on the signal probability of reconvergent nodes by cutting the multiple-fanout reconvergent input lines and assigning an appropriate probability range to the cut lines and then propagating the bounds to all the other lines of the circuits by using propagation formulas for trees. The effectiveness of the cutting algorithm, however, depends on the non deterministic choice of the cuts; well-chosen cuts lead to better estimates of the signal probabilities while poorly chosen cuts result in poor estimates. The algorithm runs in polynomial time in terms of the size of the circuits. Ercolani et al. presents [6] a procedure for propagating the signal probabilities from the circuit inputs toward the circuit outputs using only pairwise correlations between circuit lines and ignoring higher order correlations. The signal probability of a product term is estimated by breaking down the implicant into a tree of 2-input AND gates, computing the correlation coefficients of the internal nodes and hence the signal probability at the output. Similarly, the signal probability of a sum term is estimated by breaking down the implicate into a tree of 2-input OR gates.

People working in power estimation area have also considered the issue of signal probability estimation. [7] gives an exact procedure based on Ordered Binary-Decision Diagrams (OBDDs) [8] which is linear in the size of the corresponding function graph (the size of the graph, of course, may be exponential in the number of circuit inputs). Using an event-driven simulation-like technique, [7] describes a mechanism for propagating a set of probability waveforms throughout the circuit. Unfortunately, this approach doesn't take into account the correlations that might appear due to reconvergent fan-out among the internal nodes of the circuit. [10] extends this approach to account for first-order spatial correlations among probabilistic waveforms. [9] uses symbolic simulation to produce the exact boolean conditions for switching at a particular node of the circuit. This approach is expensive in terms of computational cost (time and space requirements).

None of the methods summarized above adequately capture temporal correlations between signal probabilities for a given node in a circuit. Consequently, new techniques that partially account for these correlations are emerging (e.g. [11]).

The approach proposed in this paper improves the state-of-the-art by a new analytical model which accounts for spatio-temporal correlations. Its mathematical foundation is probabilistic in nature, and consists of using lag-one Markov Chains to capture different kinds of depedencies in combinational circuits under a zero-delay model. Temporal correlations for the values of some signal x in two successive clock cycles are considered through a Markov Chain with only two states; first-order spatial correlations for pairs of signals (x,y) are modelled by a four-state Markov Chain. For the first time to our knowledge, we have considered in a systematic way different kinds of dependencies in large combinational modules for both pseudorandom and biased input streams; in addition, we report here the results of a detailed analysis and our experiences on benchmark circuits.

The results presented in this paper are useful in power estimation and low power design; once the system, architectural and technological decisions for power minimization are made, it is the switching activity of the logic that determines the power consumption of a circuit. Our approach provides a sound framework for efficiently and accurately estimating the effects of different transformations/optimizations on the power consumption of the circuits under comlplex spatio-temoral correlations.

The paper is organized as follows. In section 2 we present in detail our model for switching activity estimation and we provide a measure of its complexity . In section 3 we give some practical considerations and our experiences on benchmark circuits. Finally, we summarize our main results and we indicate possible extentions.

2. An analytical model for dependencies

2.1. Temporal correlations

We treat the sequence that corresponds to different values of a signal line x as a discrete process where time units $1, 2, \ldots, n$ represent the time instances when the input vectors $V_1, V_2, \ldots, V_n$ are applied to the circuit under consideration. During the application of the input vectors, x may be 0 or 1, so that if we define its state at time n by random variable x_n, then the behavior of line x can be described as a lag-one Markov Chain $\{x_n\}_{n>1}$, over the state set S = {0,1}, through the transition matrix Q [12]:

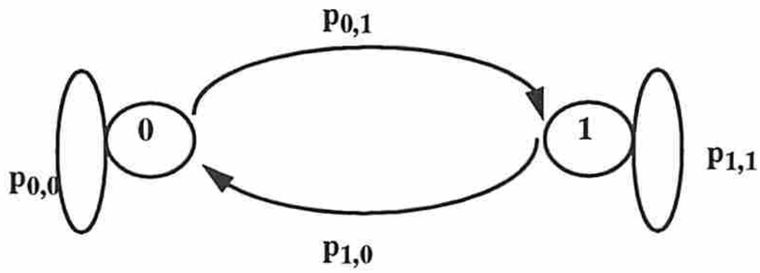

Fig.1

$$x_n = \begin{cases} 0 \text{ if } x = 0 \\ 1 \text{ if } x = 1; \end{cases} \qquad Q = \begin{bmatrix} p^x_{0,0} & p^x_{1,0} \\ p^x_{0,1} & p^x_{1,1} \end{bmatrix} \tag{1}$$

Every entry $p_{i,j}$ in the Q matrix represents a conditional probability and may be viewed as the one-step transition probability to state i at step n from state j at step n-1. The expressions for these conditional probabilities are:

$$\begin{aligned} p^x_{0,0} &= p((x(t) = 0) | (x(t-\delta) = 0)) = \frac{p((x(t) = 0) \wedge (x(t-\delta) = 0))}{p(x(t-\delta) = 0)} \\ p^x_{0,1} &= p((x(t) = 1) | (x(t-\delta) = 0)) = \frac{p((x(t) = 1) \wedge (x(t-\delta) = 0))}{p(x(t-\delta) = 0)} \\ p^x_{1,0} &= p((x(t) = 0) | (x(t-\delta) = 1)) = \frac{p((x(t) = 0) \wedge (x(t-\delta) = 1))}{p(x(t-\delta) = 1)} \\ p^x_{1,1} &= p((x(t) = 1) | (x(t-\delta) = 1)) = \frac{p((x(t) = 1) \wedge (x(t-\delta) = 1))}{p(x(t-\delta) = 1)} \end{aligned} \tag{2}$$

In the Q matrix, every column adds to unity, i.e:

$$\begin{aligned} p^x_{0,0} + p^x_{0,1} &= 1 \\ p^x_{1,0} + p^x_{1,1} &= 1 \end{aligned} \tag{3}$$

A lag-one Markov Chain has the property that one-step transition probabilities do not depend on the 'history', i.e they are the same irrespective of the number of previous steps. If the process $\{x_n\}_{n>1}$ is homogenous, then the probability distribution of the chain $\mathcal{P}$ may be expressed as:

$$\mathcal{P} = (Q)^n \mathcal{P}_0 \tag{4}$$

where $\mathcal{P}_0$ is the initial distribution vector.
If we assume the stationarity of the process $\{x_n\}_{n>1}$, then the relation (3) becomes [12]:

$$\mathcal{P} = Q\mathcal{P} \tag{5}$$

***Proposition 1**:* The signal probabilities may be expressed in terms of conditional probabilities as follows:

$$p(x=0) = \frac{p^x_{1,0}}{p^x_{1,0}+p^x_{0,1}} \qquad p(x=1) = \frac{p^x_{0,1}}{p^x_{1,0}+p^x_{0,1}} \tag{6}$$

Proof: Relation (5) may be written explicitly as:

$$\begin{bmatrix} p\,(x=0) \\ p\,(x=1) \end{bmatrix} = \begin{bmatrix} p^x_{0,0} & p^x_{1,0} \\ p^x_{0,1} & p^x_{1,1} \end{bmatrix} \begin{bmatrix} p\,(x=0) \\ p\,(x=1) \end{bmatrix}$$

or

$$p(x=0) = p^x_{0,0}p(x=0) + p^x_{1,0}p(x=1) \quad p(x=1) = p^x_{0,1}p(x=0) + p^x_{1,1}p(x=1)$$

where $p(x=1)$ represents the signal probability. But we have that $p(x=0) = 1 - p(x=1)$, respectively $p(x=1) = 1 - p(x=0)$ and then relations (6) follow immediately.

❒

***Definition 1**:* We define the transition probabilities as follows:

$$\begin{aligned} p(x_{0\to 0}) &= p\,((x\,(t) = 0) \wedge (x\,(t-\delta) = 0)) \\ p(x_{0\to 1}) &= p\,((x\,(t) = 1) \wedge (x\,(t-\delta) = 0)) \\ p(x_{1\to 0}) &= p\,((x\,(t) = 0) \wedge (x\,(t-\delta) = 1)) \\ p(x_{1\to 1}) &= p\,((x\,(t) = 1) \wedge (x\,(t-\delta) = 1)) \end{aligned} \tag{7}$$

***Proposition 2**:* Transition probabilities may be expressed in terms of conditional probabilities as:

$$p(x_{0\to 0}) = \frac{p^x_{1,0}p^x_{0,0}}{p^x_{1,0}+p^x_{0,1}} \qquad p(x_{0\to 1}) = \frac{p^x_{1,0}p^x_{0,1}}{p^x_{1,0}+p^x_{0,1}} \tag{8}$$

$$p(x_{1\to 0}) = \frac{p^x_{1,0}p^x_{0,1}}{p^x_{1,0}+p^x_{0,1}} \qquad p(x_{1\to 1}) = \frac{p^x_{1,1}p^x_{0,1}}{p^x_{1,0}+p^x_{0,1}}$$

Proof: Using the relation (2) and assuming the stationarity property for the process, we have:

$$p(x_{i\to j}) = p(x=i)p^x_{i,j}$$

for any values $i, j = 0, 1$. From relation (6), the above formulas are straightforward.

□

Proposition 3: Conditional probabilities may be expressed in terms of transition probabilities as:

$$p^x_{0,0} = \frac{p(x_{0\to 0})}{p(x_{0\to 0})+p(x_{0\to 1})} \quad p^x_{0,1} = \frac{p(x_{0\to 1})}{p(x_{0\to 0})+p(x_{0\to 1})}$$

$$p^x_{1,0} = \frac{p(x_{1\to 0})}{p(x_{1\to 0})+p(x_{1\to 1})} \quad p^x_{1,1} = \frac{p(x_{1\to 1})}{p(x_{1\to 0})+p(x_{1\to 1})} \tag{9}$$

Proof: It suffices to use the following two identities and the equations (8):

$$p(x_{0\to 0})+p(x_{0\to 1}) = \frac{p^x_{1,0}}{p^x_{1,0}+p^x_{0,1}}$$

$$p(x_{1\to 0})+p(x_{1\to 1}) = \frac{p^x_{0,1}}{p^x_{1,0}+p^x_{0,1}}$$

□

Relying on Propositions 1, 2, and 3, the relationship between signal, conditional and transition probabilities can be illustrated as below:

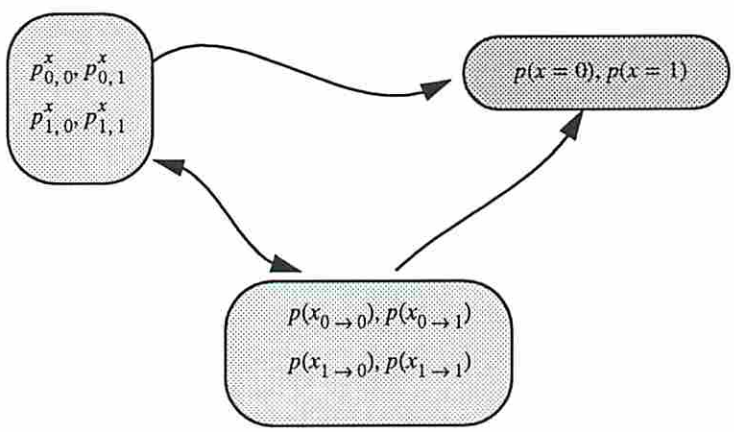

Fig.2

As we can see, to compute the signal probabilities we need less information, but the ability to derive anything else is severely limited; on the other side, once we get either conditional or transition probabilities we have all we need for that particular signal.

Definition 2: For any given line x, the switching activity is:

$$sw(x) = p(x_{0 \to 1}) + p(x_{1 \to 0}) = 2\frac{p^x_{1,0}p^x_{0,1}}{p^x_{1,0} + p^x_{0,1}} \tag{10}$$

2.2. Spatial correlations

This type of correlations has two important sources:
- Structural dependencies due to reconvergent fan-out (RFO);
- Pattern dependencies, that is, normally independent signal lines which become correlated due to a particular sequence of inputs.

To take into account the exact correlations is practically impossible even for small circuits. To make this problem more tractable, we allowed only pairwise correlated signals, which is undoubtedly an approximation, but provides good results in practice. Consequently, we considered the correlations for all 16 possible transitions of a pair of signals (x,y) and modelled it as a lag-one Markov Chain with 4 states (states 0, 1, 2, 3 which stand for encodings 00, 01, 10, 11 of (x,y)):

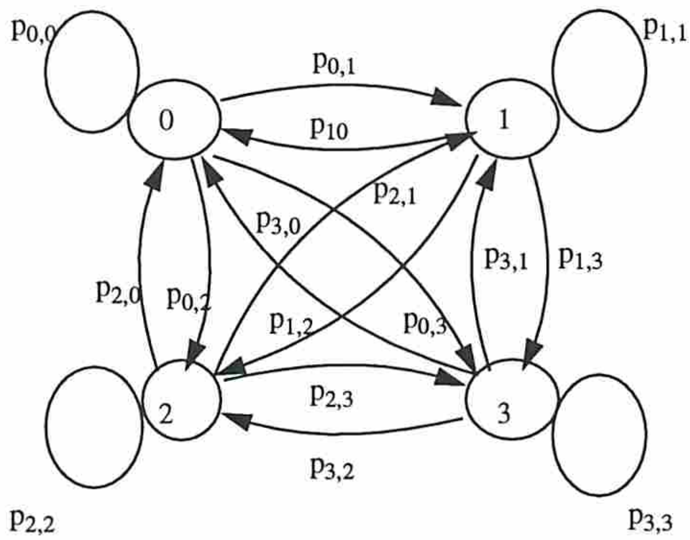

Fig.3

Definition 3: We define the conditional probability $p_{a,b}$ as:

$$p_{a,b} = p(x(t) = k \wedge y(t) = l | x(t-\delta) = i \wedge y(t-\delta) = j) \tag{11}$$

where a, b = 0, 1, 2, 3, a being encoded as ij and b as kl.

Ercolani et al. consider in [6] structural dependencies between any two signals in a circuit, through the signal correlation coefficients (*SC*); these coefficients can be expressed as:

$$SC_{ij}^{xy} = \frac{p\,(x = i \wedge y = j)}{p(x = i)p(y = j)} \tag{12}$$

where i, $k = 0,1$. Assuming that higher order correlations of two signals to a third one can be neglected, they use the following approximation:

$$p\,(x = i \wedge y = j \wedge z = k) = \frac{p\,(x = i \wedge y = j)\,p\,(x = i \wedge z = k)\,p\,(y = j \wedge z = k)}{p(x = i)p(y = j)p(z = k)}$$

Differently stated, the correlation coefficient among three signals was defined as:

$$SC_{ijk}^{xyz} = \frac{p(x = i \wedge y = j \wedge z = k)}{p(x = i)p(y = j)p(z = k)}$$

which is then equal to:

$$SC_{ijk}^{xyz} = SC_{ij}^{xy} SC_{ik}^{xz} SC_{jk}^{yz}$$

Our approach is more general; in order to capture the spatial correlations between signals, for each pair of signals (x,y) and for all possible transitions for both of them, we consider the transition correlation coefficients (*TC*).

***Definition 4**:* We define the *TC* for two signals x, y as:

$$TC_{ij,\,kl}^{xy} = \frac{p(x\,(t-\delta) = i \wedge x\,(t) = k \wedge y\,(t-\delta) = j \wedge y\,(t) = l)}{p(x\,(t-\delta) = i \wedge x\,(t) = k)p(y\,(t-\delta) = j \wedge y\,(t) = l)} \tag{13}$$

where $i, j, k, l = 0, 1$.

***Proposition 1**:* For every pair of signals (x,y) and all possible values $i, j, k, l = 0, 1$, the following holds:

$$SC_{ij}^{xy} = \sum_{k,\,l = 0,\,1} TC_{ij,\,kl}^{xy} \frac{p(x_{i \to k})p(y_{j \to l})}{p(x = i)p(y = j)} \tag{14}$$

Proof: For the four-state Markov Chain in fig.1 and relation (11) we have that $\sum_{b=0}^{3} p_{a,b} = 1$ for every value of a; that means

$$\sum_{k,\,l = 0,\,1} p(x\,(t) = k \wedge y\,(t) = l \,|\, x\,(t-\delta) = i \wedge y\,(t-\delta) = j) = 1$$

But, according to the definition of conditional probabilities

$$p(x\,(t) = k \wedge y\,(t) = l \,|\, x\,(t-\delta) = i \wedge y\,(t-\delta) = j) =$$

$$= \frac{p(x_{i \to k} \wedge y_{j \to l})}{p(x(t-\delta) = i \wedge y(t-\delta) = j)}$$

and then

$$\sum_{k,l=0,1} \frac{p(x_{i \to k} \wedge y_{j \to l})}{p(x(t-\delta) = i \wedge y(t-\delta) = j)} = 1$$

Hence, from the above relation , applying (12) and (13) we get

$$\sum_{k,l=0,1} \frac{p(x_{i \to k}) p(y_{j \to l}) TC^{xy}_{ij,kl}}{p(x(t-\delta) = i) p(y(t-\delta) = j) SC^{xy}_{ij}} = 1$$

Equivalently, we get:

$$\sum_{k,l=0,1} \frac{TC^{xy}_{ij,kl}}{SC^{xy}_{ij}} \frac{p(x_{i \to k}) p(y_{j \to l})}{p(x = i) p(y = j)} = 1$$

and hence the required relation is satisfied:

$$SC^{xy}_{ij} = \sum_{k,l=0,1} TC^{xy}_{ij,kl} \frac{p(x_{i \to k}) p(y_{j \to l})}{p(x = i) p(y = j)}$$

❐

Proposition 2: For every pair of signals (x,y) and all possible values $i, j = 0, 1$, the following equations hold:

$$\begin{aligned} \sum_{j=0,1} SC^{xy}_{ij} p(y = j) &= 1 \qquad \forall i = 0, 1; \\ \sum_{i=0,1} SC^{xy}_{ij} p(x = i) &= 1 \qquad \forall j = 0, 1. \end{aligned} \tag{15}$$

Proof: From the definition of SC, we get

$$\sum_{j=0,1} SC^{xy}_{ij} p(y = j) = \frac{1}{p(x = i)} p\left((x = i) \sum_{j=0,1} (y = j)\right) = 1$$

The second equation follows in a similar manner.

❐

Proposition 3: For every pair of signals (x,y) and all possible values $i, j, k, l = 0, 1$ the following equations hold:

$$\begin{aligned} \sum_{j,l=0,1} TC^{xy}_{ij,kl} p(y_{j \to l}) &= 1 \quad \forall i, k = 0, 1; \\ \sum_{i,k=0,1} TC^{xy}_{ij,kl} p(x_{i \to k}) &= 1 \quad \forall j, l = 0, 1; \end{aligned} \tag{16}$$

Proof: Similar to the proof for Proposition 2, but using the definition of *TC*.

❐

We provide in the following two useful results:

Proposition 4: The set of 4 equations and 4 unknowns SC_{ij}^{xy}, $i, j = 0, 1$ in Proposition 2 is indeterminate. Moreover, the matrix of the system has the rank ≤ 3.

Proposition 5: The set of 8 equations and 16 unknowns $TC_{ij,kl}^{xy}$, $i, j, k, l = 0, 1$ is indeterminate; the matrix of the system has the rank ≤ 7.

The last two propositions are very important from a practical point of view. The set of equations involving *SC*'s may be solved knowing only SC_{11}^{xy} for example, and that was the approach taken by Ercolani et al. in [6] (although, no similar analysis appeared in the original paper). In the more complex case involving *TC*'s, we need to know at least 9 out of 16 coefficients in order to deduce all values.

2.3. Propagation mechanisms

In what follows we ignore higher order correlations, that is, the correlation between any number of signals is expressed only in terms of pairwise correlation coefficients; the same assumption was used in [6], but only for signal correlation coefficients.

Definition 5: We define the *TC* among three signals as:

$$TC_{ijk,\, lmn}^{xyz} = \frac{p(x_{i \to l} y_{j \to m} z_{k \to n})}{p(x_{i \to l}) p(y_{j \to m}) p(z_{k \to n})}$$

Neglecting higher order correlations, we therefore assume that the following holds for any signals x, y, z and any values $i, j, k, l, m, n = 0, 1$:

$$TC_{ijk,\, lmn}^{xyz} = TC_{ij,\, lm}^{xy} TC_{jk,\, mn}^{yz} TC_{ik,\, ln}^{xz} \tag{17}$$

Definition 5 and relation (17) may be easily extended to any number of signals. Based on the above assumption, we use an OBDD-based procedure for computing the transition probabilities and for propagating the *TC*'s through the network. The main reason for using the OBDD representation for a signal is that it is a canonical representation of a Boolean function and that it offers a disjoint cover which is essential for our purposes. Depending on the set of signals with respect to which we represent a node of the boolean network, two approaches may be used:

- The global approach - for each node, we build the OBDD in terms of the primary inputs of the circuit;
- The incremental approach - for each node, we build the OBDD in terms of its immediate fanin and propagate the transition probabilities and the *TC*'s through the boolean network .

The first approach is more accurate, but requires much more memory and running time; indeed, for many large circuits, it is nearly impractical. The second one, offers accurate enough results whilst being more efficient as far as memory requirement and running time are concerned.

a) Computation of the transition probabilities

Let f be a node in the boolean network represented in terms of n (immediate or primary input) variables x_1, x_2, ..., x_n; it may be defined through the following two sets of OBDD paths:

- Π_1 - the set of all paths in the ON-set of f
- Π_0 - the set of all paths in the OFF-set of f

Some of the approaches reported in the literature (e.g. [9]), use the XOR-OBDD of f at two consecutive time steps to compute the transition probabilities. We consider instead only the OBDD of f and through a dynammic programming approach, we compute the transition probabilities more efficiently.
Based on the above representation, the event 'f switching from value i to value j' ($i, j = 0, 1$), may be written as:

$$f_{i \to j} = \sum_{\pi \in \Pi_i} \sum_{\pi' \in \Pi_j} \prod_{k=1}^{n} x_{k_{i_k \to j_k}} \tag{18}$$

where i_k, j_k are the values of variable x_k on the path π and π' respectively ($i_k, j_k = 0, 1, 2$, where 2 stands for don't care values) for each $k = 1, 2, ..., n$. Thus, the probability that f switches from i to j may be expressed as:

$$p(f_{i \to j}) = p(\sum_{\pi \in \Pi_i} \sum_{\pi' \in \Pi_j} \prod_{k=1}^{n} x_{k_{i_k \to j_k}}) \tag{19}$$

Applying the property of disjoint events (which is satisfied by the collection of paths in the OBDD), the above formula becomes:

$$p(f_{i \to j}) = \sum_{\pi \in \Pi_i} \sum_{\pi' \in \Pi_j} p(\prod_{k=1}^{n} x_{k_{i_k \to j_k}}) \tag{20}$$

However, since the variables x_k may *not* be spatially independent of one another, the probability of a path to 'switch' from $(i_1, i_2, ..., i_n)$ to $(j_1, j_2, ..., j_n)$ may *not* be expressed as the product of transition probabilities for individual variables. Instead, we will use the following result which holds if we neglect higher order correlations.

Proposition 6: If relation (17) is true for any three signals from the set $\{x_1, x_2, ..., x_n\}$, then:

$$p(\prod_{k=1}^{n} x_{k_{i_k \to j_k}}) = \prod_{k=1}^{n} (p(x_{k_{i_k \to j_k}}) \prod_{k < l \leq n} TC^{x_k x_l}_{i_k i_l, j_k j_l}) \tag{21}$$

Proof: Follows directly from relation (17) by induction on the number of variables.

❐

According to this result, the transition probability of the signal f for any values i, $j = 0, 1$ satisfies the following:

Proposition 7 The transition probability of a signal f from state i to state j ($i, j = 0, 1$) is:

$$p(f_{i \to j}) = \sum_{\pi \in \Pi_i} \sum_{\pi' \in \Pi_j} \prod_{k=1}^{n} (p(x_{k_{i_k \to j_k}}) \prod_{k < l \le n} TC^{x_k x_l}_{i_k i_l, j_k j_l}) \tag{22}$$

Proof: Follows immediately applying Proposition 6.

❐

Though this expression seems to be very complicated, its complexity is within reasonable bounds. We will show that it is not necessary to enumerate all *pairs* of paths in the OBDD (which would provide a quadratic complexity in the number of paths in the OBDD), but for a fixed path in Π_i the computation may be done in linear time in terms of the OBDD-nodes.

For the incremental approach, we need a mechanism not only for computing the transition probabilities, but also for propagating the TC's through the boolean network. For a given node in the circuit, it is only necessary to propagate the TC of the output with respect to the signals on which the inputs depend. The dependency between an input and another signal may have as a cause either a RFO or a propagated primary input dependency.

b) Propagation of the transition correlation coefficients

Let f be a node with immediate inputs $x_1, x_2, ..., x_n$ and x a signal on which at least one of the inputs $x_1, x_2, ..., x_n$ depends. According to the definition of the TC, for every $i, j, p, q = 0, 1$ possible values of f and x respectively, we have:

$$TC^{fx}_{ip, jq} = \frac{p(f_{i \to j} x_{p \to q})}{p(f_{i \to j}) p(x_{p \to q})} \tag{23}$$

Since the transition probabilities for f and x are already computed at this point, the only problem is to compute the probability of both f and x switching from i to j and from p to q respectively. We get the following important result:

Proposition 8 The TC between signals f and x, for any values $i, j, p, q = 0, 1$ may be expressed as:

$$TC^{fx}_{ip, jq} = \frac{\sum_{\pi \in \Pi_i} \sum_{\pi' \in \Pi_j} \prod_{k=1}^{n} (TC^{x_k x}_{i_k p, j_k q} p(x_{k_{i_k \to j_k}}) \prod_{1 \le k < l \le n} TC^{x_k x_l}_{i_k i_l, j_k j_l})}{p(f_{i \to j})} \tag{24}$$

Proof: Using the representation of the event 'f switches from i to j' given in (18), we obtain the following for the event 'f switches from i to j and x switches from p to q *simultaneously*':

$$f_{i \to j} x_{p \to q} = \left(\sum_{\pi \in \Pi_i} \sum_{\pi' \in \Pi_j} \prod_{k=1}^{n} x_{k_{i_k \to j_k}} \right) x_{p \to q}$$

and:

$$p(f_{i \to j} x_{p \to q}) = p(\sum_{\pi \in \Pi_i} \sum_{\pi' \in \Pi_j} \{ x_{p \to q} \prod_{k=1}^{n} x_{k_{i_k \to j_k}} \})$$

Applying the disjointness property of the paths, we get:

$$p(f_{i \to j} x_{p \to q}) = \sum_{\pi \in \Pi_i} \sum_{\pi' \in \Pi_j} p(x_{p \to q} \prod_{k=1}^{n} x_{k_{i_k \to j_k}})$$

Since the variables x_i may not be independent and, furthermore, at least one of them depends on x, we need to apply the result provided by proposition 1 for the set of n+1 variables $\{x_1, x_2, \ldots, x_n, x\}$:

$$p(f_{i \to j} x_{p \to q}) = \sum_{\pi \in \Pi_i} \sum_{\pi' \in \Pi_j} p(x_{p \to q}) \prod_{k=1}^{n} (TC^{x_k x}_{i_k p, j_k q} p(x_{k_{i_k \to j_k}}) \prod_{k < l \le n} TC^{x_k x_l}_{i_k i_l, j_k j_l})$$

Thus, the TC between f and x follows immediately.

❐

c) Complexity issues

In order to assess the complexity claimed above, let us define the following notation:

$$f_{\pi \to j} = \sum_{\pi' \in \Pi_j} \prod_{k=1}^{n} x_{k_{i_k \to j_k}} \tag{25}$$

where π is a fixed path in Π_i. Thus, using the disjointness property, the corresponding probability is:

$$p(f_{\pi \to j}) = \sum_{\pi' \in \Pi_j} p(\prod_{k=1}^{n} x_{k_{i_k \to j_k}})$$

Since the path π is fixed, the above probability may be computed on the OBDD in the same way as a signal probability. The idea is that, using Shannon decomposition, the signal probability (and hence the above probability) may be computed in linear time in the number of the OBDD-nodes [8]. Thus, may be decomposed as follows:

$$f_{\pi \to j} = x_{k_{i_k \to 0}} f^{\overline{x_k}}_{\pi \to j} + x_{k_{i_k \to 1}} f^{x_k}_{\pi \to j} \tag{26}$$

where $f^{\overline{x_k}}_{\pi \to j}, f^{x_k}_{\pi \to j}$ are the cofactors with respect to $\overline{x}_k$ and x_k, respectively. Based on this recursive decomposition, we may also write a similar relation for the corresponding probabilities, taking also into account the possible existing correlations:

$$p(f_{\pi \to j}) = p(x_{k_{i_k \to 0}}) p(f^{\overline{x_k}}_{\pi \to j}) \prod_{1 \le k < l \le n} TC^{x_k x_l}_{i_k i_l, 0 j_l} + p(x_{k_{i_k \to 1}}) p(f^{x_k}_{\pi \to j}) \prod_{k < l \le n} TC^{x_k x_l}_{i_k i_l, 1 j_l} \tag{27}$$

Having computed this probability for each path π, we immediately get the corresponding transition probabilities and hence the switching activity.Thus, for a fixed path π, the complexity is $O(n^2 N)$ where n is the number of variables and N is the number of nodes in the OBDD. The n^2 factor comes from the necessity of taking into account the correlations: besides the transition probabilities, we also have to keep track of the TC's involved on each path. There is a number of $\binom{n}{2}$ factors in the product, thus the complexity is quadratic in the number of variables.

Hence, overall, for all the paths in Π_i, the time complexity is $O(n^2 NP)$ where P is the number of paths in the OBDD. In the incremental approach, this is within reasonable limits since n does not exceed 3 or 4 variables in the immediate fanin of the node.

Example: Let's consider the following function: $f = x_1 \oplus x_2 \oplus x_3$ and its OBDD representation from fig.4. Suppose $i = 0, j = 1$ and $\pi = (0\ 1\ 1)$ (a fixed path in the OFF-set of f, Π_0). We compute the probability given in (27) by using a bottom-up parsing of the OBDD from the leaf labelled 1 to the root. We adopt a dynamic programming approach in which at each level we use the results computed at lower levels. For each node, the partial results are shown in fig. 4. The same operations are performed for any other path in Π_0, thus allowing us to compute in the same manner all the transition probabilities and hence the switching activity, based on relation (10). A similar approach is used to propagate the TC between f and some other signal x

$$\begin{aligned} p(f_{\pi \to j}) = {} & p(x_{1_{0\to 0}})p(x_{2_{1\to 1}})p(x_{3_{1\to 0}})TC^{x_2x_3}_{11,10}TC^{x_1x_3}_{01,00}TC^{x_1x_2}_{01,01} + \\ & p(x_{1_{0\to 0}})p(x_{2_{1\to 0}})p(x_{3_{1\to 1}})TC^{x_2x_3}_{11,01}TC^{x_1x_3}_{01,01}TC^{x_1x_2}_{01,00} + \\ & p(x_{1_{0\to 1}})p(x_{2_{1\to 0}})p(x_{3_{1\to 0}})TC^{x_2x_3}_{11,00}TC^{x_1x_3}_{01,10}TC^{x_1x_2}_{01,10} + \\ & p(x_{1_{0\to 1}})p(x_{2_{1\to 1}})p(x_{3_{1\to 1}})TC^{x_2x_3}_{11,11}TC^{x_1x_3}_{01,11}TC^{x_1x_2}_{01,11} + \end{aligned}$$

$$p(f^{\overline{x_1}}_{\pi \to j}) = p(x_{2_{1\to 1}})p(x_{3_{1\to 0}})TC^{x_2x_3}_{11,10} + p(x_{2_{1\to 0}})p(x_{3_{1\to 1}})TC^{x_2x_3}_{11,01}$$

$$p(f^{x_1}_{\pi \to j}) = p(x_{2_{1\to 0}})p(x_{3_{1\to 0}})TC^{x_2x_3}_{11,00} + p(x_{2_{1\to 1}})p(x_{3_{1\to 1}})TC^{x_2x_3}_{11,11}$$

$$p(f^{x_1\overline{x_2}}_{\pi \to j}) = p(f^{\overline{x_1}x_2}_{\pi \to j}) = p(x_{3_{1\to 0}})$$

$$p(f^{x_1x_2}_{\pi \to j}) = p(f^{\overline{x_1}\overline{x_2}}_{\pi \to j}) = p(x_{3_{1\to 1}})$$

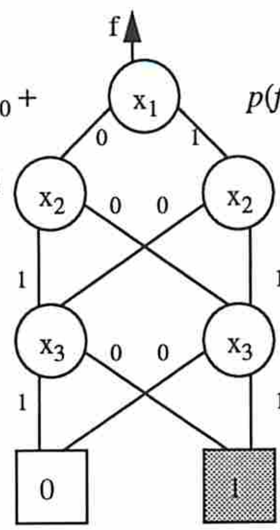

Fig.4

3. Practical considerations and experimental results

All experiments were performed in the SIS environment [14] on a SPARC II workstation with 64Mbytes of memory; the working procedure is shown below:

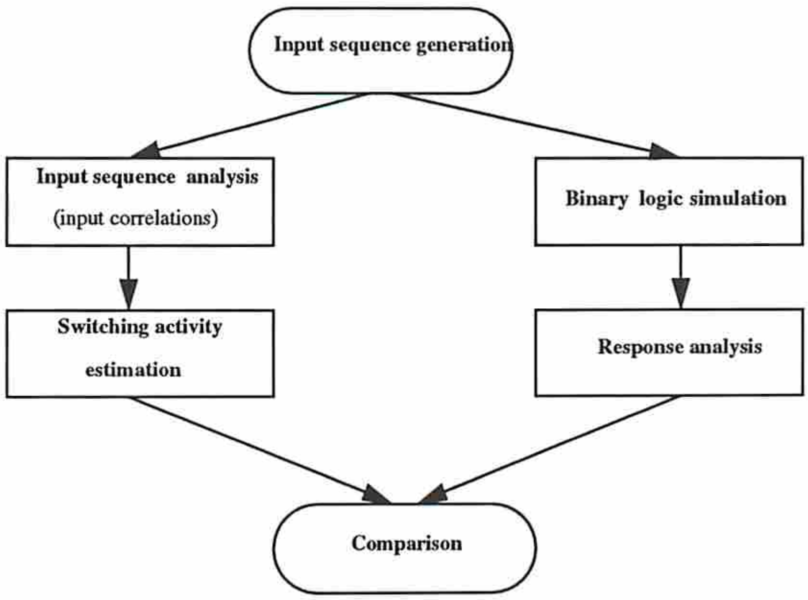

Fig.5

To generate pseudorandom (PR) inputs we have used as input generator a maximal-length linear feed-back shift register (LFSR) modified to include the all-zero pattern [13]; these registers are based on primitive polynomials that is, they randomly generate all distinct patterns that correspond to a given degree before repeating the sequence. Purely random generators do not exist, therefore the primitive polynomials used, give us multiple correlations among primary inputs.The length of the input register was set equal to the number of inputs of the circuit under analysis, thereby creating a pseudorandom source; when the length of this register became huge, we tried to keep the time/space requirements at a reasonable level and hence, for these cases we generated only a significant part of the exhaustive sequence (up to 2^{18} input patterns).

As the standard measure for power estimation, we have used the average switching activity at each node of the circuit calculated as in (11)[1]. In our experiments, we were mainly interested, to measure the accuracy of the model in estimating the switching activity locally (at each internal node of interest) and globally (for the entire circuit), given a set of inputs with spatiotemporal correlations. The analysis part of the experiment may be skipped if the user specifies directly the characteristics of the input stream (transition probabilities and correlation coefficienys).

To illustrate the main concepts of our approach, we consider in fig.6 the ISCAS circuit C17, fed by the sequence generated with the primitive polynomial $p(x) = 1 \oplus x \oplus x^3$. Due to the deterministic way in which

1. To calculate average power consumption of a gate in a synchronous CMOS circuit, one can use the well-known formula $P_{avg} = 0.5 (V_{dd}^2 / T_{cycle}) C_{load} sw(x)$ where V_{dd} is the supply voltage, T_{cycle} is the clock cycle period, C_{load} is the load capacitance and x is the output of the gate.

we generate the input sequence, independent lines become correlated as is the case with inputs 1 & 2, 2 & 3, 3 & 6, 6 & 7; moreover, the fan-out points on the input lines add in turn additional correlations. For an accurate analysis of the switching activity, we have to account for all these dependencies. In Table 1 we list the transition probability coefficients for this particular input sequence; we mention that in this case, all signal correlation coefficients are equal to 1. In Table 2 we present the estimated and exact values of the switching activity per clock cycle. In fig.6, the color code is used to reflect the switching activity at the output of the gates, i.e. darker gates are more active.

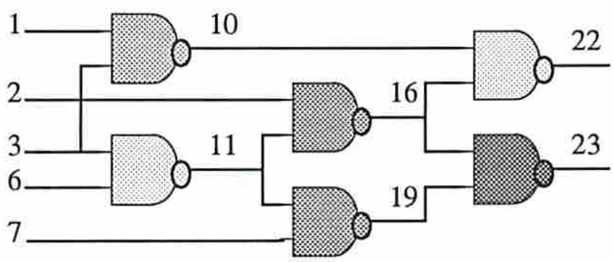

Fig.6

Table 1: C17 - TC for PR inputs

ij,kl	1&2	2&3	3&6	6&7
00,00	2.0000	2.0000	2.0000	1.7143
00,01	0.0000	0.0000	0.0000	0.4444
00,10	2.0000	2.0000	2.0000	1.7778
00,11	0.0000	0.0000	0.0000	0.0000
01,00	2.0000	2.0000	2.0000	2.2857
01,01	0.0000	0.0000	0.0000	0.0000
01,10	2.0000	2.0000	2.0000	1.7778
01,11	0.0000	0.0000	0.0000	0.0000
10,00	0.0000	0.0000	0.0000	0.0000
10,01	2.0000	2.0000	2.0000	1.7778
10,10	0.0000	0.0000	0.0000	0.4444
10,11	2.0000	2.0000	2.0000	1.7143
11,00	0.0000	0.0000	0.0000	0.0000
11,01	2.0000	2.0000	2.0000	1.7778
11,10	0.0000	0.0000	0.0000	0.0000
11,11	2.0000	2.0000	2.0000	2.2857

Table 2: C17 - sw_act for PR inputs

Node	Estimated sw_act	Exact sw_act
1	0.5000	0.5000
2	0.5000	0.5000
3	0.5000	0.5000
6	0.5000	0.5000
7	0.5625	0.5625
10	0.3750	0.3907
11	0.2500	0.2649
16	0.5000	0.5236
19	0.5687	0.5236
22	0.2978	0.3125
23	0.6006	0.5625

To bound the error during the propagation procedure, we used two mechanisms:

- One is based on the paradigm in fig.2 (We calculate the signal probabilities independently and use these values as a more reliable measure for correcting the values of transition probabilities that fall out of range

[0, 1]; more precisely, we normalize conditional probabilities such that relations (6) hold at each step);

- The other is based on limiting the *TC* values (We normalize the values of coefficients using the set of equations (16)).

It should be pointed out that the actual values for all coefficients in Table 1 represent the characteristics of the input stream, so for example, if we select another primitive polynomial to generate the inputs, we may obtain a completely different set of transition correlation coefficients. This dependency is even more salient if we consider 'biased inputs' (i.e. the switching activity is not 0.5). To generate such sequences, we used a simple functional generator based on the 'random' function in C language. We set up a specific threshold $t \in [0,1]$ and generated a set of random numbers in [0,1]. If these numbers exceeded t, then the output of the generator was set to 1; otherwise the output was 0. We give in Tables 3 and 4, the values obtained for two such sequences, and in fig.7 the new distribution of switching activity among the internal nodes of the circuit. We note that in these cases we have spatial dependencies practically among all primary inputs.

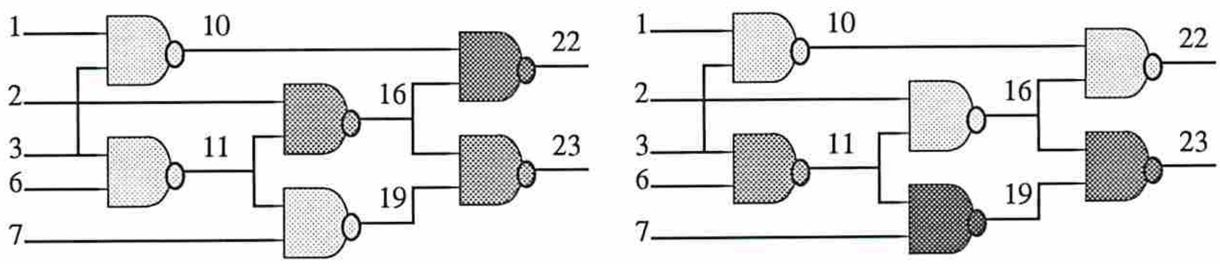

Fig.7

Table 3: C17 - sw_act for biased inputs

Node	Estimated sw_act	Exact sw_act
1	0.5625	0.5625
2	0.3750	0.3750
3	0.3125	0.3125
6	0.5625	0.5625
7	0.5000	0.5000
10	0.3125	0.3125
11	0.3125	0.3125
16	0.4569	0.5000
19	0.3367	0.3125
22	0.4960	0.5000
23	0.4527	0.4375

Table 4: C17 - sw_act for biased inputs

Node	Estimated sw_act	Exact sw_act
1	0.0625	0.0625
2	0.1250	0.1250
3	0.2500	0.2500
6	0.5000	0.5000
7	1.0000	1.0000
10	0.1250	0.1250
11	0.2500	0.2500
16	0.1295	0.1250
19	0.7262	0.7500
22	0.2040	0.1250
23	0.4597	0.5000

To assess the impact of spatio-temporal correlations on switching activity estimations, we considered three different benchmark circuits, namely **C17**, **f51m** , **5xp1** and performed the following set of experiments:

- a PR experiment where the inputs were generated with the polynomials $p(x) = 1 \oplus x^2 \oplus x^5$ for **C17**, $p(x)$

$= 1 \oplus x^4 \oplus x^5 \oplus x^6 \oplus x^7$ for **5xp1** and $p(x) = 1 \oplus x \oplus x^2 \oplus x^7 \oplus x^8$ for **f51m**;

- a biased experiment where the switching activities of the inputs were $sw(i) = 0.25$, i = 1, 2..., n-1 respectively $sw(n) = 0.375$, where $n = 5$ for **C17**, $n = 7$ for **5xp1** and $n = 8$ for **f51m** (these values were obtained by AND-ing pairwise the normal outputs of LFSRs which correspond to degrees 5, 7 and 8).

To compare our model with different other approaches reported in the literature, we analyzed exhaustively these circuits for the switching activity at primary outputs and all internal nodes. Comparing our estimations with the exact binary simulation results, we reported in Tables 5 ÷ 10, the usual measures for accuracy: maximum error (MAX), mean error (MEAN), root-mean square (RMS) and standard deviation (STD).

Table 5: C17 - PR inputs

	Global approach				Incremental approach			
	With spatial correlations		Without spatial correlations		With spatial correlations		Without spatial correlations	
Error	With temporal correlations	Without temporal correlations	With temporal correlations	Without temporal correlations	With temporal correlations	Without temporal correlations	With temporal correlations	Without temporal correlations
MAX	0.0391	0.1797	0.1797	0.1797	0.0565	0.1817	0.1855	0.1855
MEAN	0.0156	0.0627	0.0616	0.0627	0.0275	0.0625	0.0664	0.0664
RMS	0.0207	0.0893	0.0892	0.0893	0.0316	0.0899	0.0917	0.0917
STD	0.0149	0.0696	0.0707	0.0696	0.0170	0.0708	0.0693	0.0693
TIME	1.8 s	1.8 s	1.8 s	1.8 s	1.4 s	1.4 s	0.3 s	0.3 s

Table 6: C17 - biased inputs

	Global approach				Incremental approach			
	With spatial correlations		Without spatial correlations		With spatial correlations		Without spatial correlations	
Error	With temporal correlations	Without temporal correlations	With temporal correlations	Without temporal correlations	With temporal correlations	Without temporal correlations	With temporal correlations	Without temporal correlations
MAX	0.0431	0.1144	0.0435	0.1577	0.0248	0.0869	0.0431	0.1551
MEAN	0.0180	0.0641	0.0331	0.0775	0.0088	0.0479	0.0331	0.0773
RMS	0.0251	0.0730	0.0345	0.0916	0.0118	0.0547	0.0344	0.0911
STD	0.0191	0.0335	0.0103	0.0534	0.0086	0.0289	0.0103	0.0528
TIME	1.9 s	1.9 s	1.8 s	1.8 s	1.4 s	1.4 s	0.3 s	0.3 s

Table 7: 5xp1 - PR inputs

	Global approach				Incremental approach			
	With spatial correlations		Without spatial correlations		With spatial correlations		Without spatial correlations	
Error	With temporal correlations	Without temporal correlations	With temporal correlations	Without temporal correlations	With temporal correlations	Without temporal correlations	With temporal correlations	Without temporal correlations
MAX	0.0234	0.1289	0.1527	0.1527	0.1323	0.1289	0.1363	0.1355
MEAN	0.0055	0.0372	0.0465	0.0498	0.0217	0.0483	0.0434	0.0433
RMS	0.0086	0.0636	0.0733	0.0758	0.0376	0.0696	0.0656	0.0654
STD	0.0069	0.0531	0.0583	0.0587	0.0315	0.0515	0.0506	0.0505
TIME	693 s	693 s	678 s	678 s	200.5 s	200.5 s	3.3 s	3.3 s

Table 8: 5xp1 - biased inputs

	Global approach				Incremental approach			
	With spatial correlations		Without spatial correlations		With spatial correlations		Without spatial correlations	
Error	With temporal correlations	Without temporal correlations	With temporal correlations	Without temporal correlations	With temporal correlations	Without temporal correlations	With temporal correlations	Without temporal correlations
MAX	0.1080	0.2188	0.1250	0.2422	0.0879	0.2188	0.1143	0.2422
MEAN	0.0271	0.0549	0.0421	0.0677	0.0270	0.0659	0.0399	0.0824
RMS	0.0439	0.0773	0.0574	0.0911	0.0366	0.0856	0.0545	0.1012
STD	0.0355	0.0560	0.0402	0.0627	0.0255	0.0563	0.0383	0.0605
TIME	712.2 s	712.2 s	683.8 s	683.8 s	205.4 s	205.4 s	3.5 s	3.5 s

Table 9: f51m - PR inputs

	Global approach				Incremental approach			
	With spatial correlations		Without spatial correlations		With spatial correlations		Without spatial correlations	
Error	With temporal correlations	Without temporal correlations	With temporal correlations	Without temporal correlations	With temporal correlations	Without temporal correlations	With temporal correlations	Without temporal correlations
MAX	0.0039	0.2720	0.2771	0.2770	0.2574	0.2767	0.2589	0.2589
MEAN	0.0008	0.0321	0.0314	0.0321	0.0224	0.0354	0.0367	0.0376
RMS	0.0015	0.0797	0.0796	0.0797	0.0659	0.0807	0.0772	0.0774
STD	0.0013	0.0754	0.0755	0.0754	0.0640	0.0749	0.0701	0.0699
TIME	286.5 s	286.5 s	273.2 s	273.2 s	35.6 s	35.6 s	1.8 s	1.8 s

Table 10: f51m - biased inputs

	Global approach				Incremental approach			
	With spatial correlations		Without spatial correlations		With spatial correlations		Without spatial correlations	
Error	With temporal correlations	Without temporal correlations	With temporal correlations	Without temporal correlations	With temporal correlations	Without temporal correlations	With temporal correlations	Without temporal correlations
MAX	0.0463	0.2020	0.2421	0.2178	0.0696	0.1927	0.3289	0.4174
MEAN	0.0115	0.0591	0.0658	0.0969	0.0280	0.0781	0.0607	0.1041
RMS	0.0185	0.0767	0.0722	0.1103	0.0393	0.0915	0.1032	0.1401
STD	0.0149	0.0505	0.0960	0.0544	0.0285	0.0492	0.0862	0.0968
TIME	290.1 s	290.1 s	276 s	276 s	65.6 s	65.6 s	1.8 s	1.8 s

The global approach refers to doing the switching activity calculation on the global OBDD representing the node function in terms of the circuit inputs, while incremental approach refers to the propagation mechanism using the network structure and the local OBDD representation (in terms of immediate inputs of the node).

As we can see, for PR inputs, global approaches with spatio-temporal correlations are overall 5 to 50 times more accurate than any other global approach which doesn't account for these dependencies. Incremental approaches which consider both types of correlations, are on average 1.5 to 3 times more accurate than the ones which neglect any of these; the price we have to pay in terms of accuracy is justified by a significant computational speed-up of incremental method vs. the global one. It is worth to note that, taking into account any of these correlations by itself, improves the accuracy of all estimations made. Thus, if our main interest is the accuracy, it's worth to include spatio-temporal correlations in a global style, if possible; otherwise, the incremental approach should be used.

For biased inputs, the global approach using both spatial and temporal correlations is 2 to 4 times more accurate; on the other hand, the incremental approach provides a gain in accuracy of 4 to 6 times. Whilst, in terms of accuracy, incremental approaches with spatio-temporal correlations provide roughly the same gain in accuracy as the global ones, the running time is clearly shorter.

These observations were proved to be consistent in all our experiments on benchmark circuits; in the following, we give the error values only for PR inputs, using the incremental approach. In reporting the error, we compared our switching activity estimates with the results of binary logic simulation at every internal node or primary output. The running time ranged from 1.3 s (for C17) to 50 min. (for C6288). These values include the time needed by the analysis module to process the input data stream in order to derive input statistics (transition probabilities and transition correlation coefficients).

Circuit	MAXIMUM ERROR	MEAN ERROR	ROOT_MEAN_SQUARE	STANDARD DEVIATION
C17	0.0565	0.0275	0.0316	0.0170
C432	0.0678	0.0131	0.0221	0.0179
C499	0.0668	0.0039	0.0084	0.0075
C880	0.1143	0.0175	0.0331	0.0283
C1355	0.0512	0.0021	0.0057	0.0053
C1908	0.0669	0.0061	0.0115	0.0098
C3540	0.1153	0.0155	0.0280	0.0233
C6288	0.1595	0.0187	0.0359	0.0310
alu4	0.1754	0.0271	0.0469	0.0386
z4ml	0.0750	0.0125	0.0211	0.0172
duke2	0.3199	0.0272	0.0657	0.0609

To conclude, two important observations should be made. First, Markov Chains are useful in modelling input correlations (this is proved by the accuracy of global approaches). Second, the degree in which any type of correlation affects the overall quality of estimations, depends on the internal structure of the circuit and the correlations among the primary inputs. The best way to use this framework in practice would be to consider both approaches in a hierarchical manner: large combinational modules may be partitioned until they become manageable in a global fashion. If there is room to improve the technique which account for spatio-temporal correlations, the assumption of neglecting higher order correlations between signals should be the first to start with.

4. Conclusions

We have proposed an original approach for estimation of the switching activity in combinational logic modules under pseudorandom or biased inputs. Using the zero-delay hypothesis, we have derived a probabilistic model based on lag-one Markov Chains and conditional probabilities. The main feature of our approach is the systematic way in which we can deal with complex dependencies that may appear in practice; more precisely, our model supports spatio-temporal correlations among the primary inputs or internal lines of the circuit under consideration. A comparative analysis and benchmark evaluations emphasize the superiority of our approach over the current existing techniques and show its practicality on large combinational modules. Our future work will concentrate on general delay models and on extensions of this approach beyond the logic level.

5. References

[1] K. Parker, and E. J. McCluskey, 'Probabilistic Treatment of General Combinational Networks', in *IEEE Trans.on Computers*, vol. C-24, June 1975

[2] L H.. Goldstein, 'Controlability/Observability Analysis of Digital Circuits', in *IEEE Trans.Circuits and Systems*, vol.CAS-26, Sept.1979

[3] S. C. Seth, L. Pan, and V.D.Agrawal 'PREDICT - Probabilistic Estimation of Digital Circuits Testability', in *Proc. 1985 Fault-Tolerant Computing Symposium*

[4] B.Krishnamurthy, and I.G.Tollis, 'Improved Techniques for Estimating Signal Probabilities', in *Proc.1986 Intl. Test Conference*

[5] J. Savir, G. S. Ditlow, and P. H. Bardell, 'Random Pattern Testability', in *IEEE Trans.on Computers*, vol. C-33, Jan.1984

[6] S. Ercolani, M. Favalli, M. Damiani, P. Olivo, and B. Ricco, 'Testability Measures in Pseudorandom Testing', in *IEEE Trans. on CAD*, vol. 11, June 1992

[7] F. N. Najm, R. Burch, P. Yang, and I. Hajj, 'Probabilistic Simulation for Reliability Analysis of CMOS VLSI Circuits', in *IEEE Trans. on CAD*, vol. CAD-9, April 1990

[8] R. E. Bryant, 'Symbolic Boolean Manipulation with Ordered Binary-Decision Diagrams', in *ACM Computing Surveys*, Vol. 24, No. 3, Sept. 1992

[9] A. Ghosh, S. Devadas, K. Keutzer, and J. White, 'Estimation of Average Switching Activity in Combinational and Sequential Circuits', in *Proc. 1992 Design Automation Conference*

[10] C.-Y. Tsui, M. Pedram, and A. M. Despain, 'Efficient Estimationof Dynamic Power Dissipation with an Application', in *Proc. 1993 Intl. Conference on Computer Aided Design*

[11] W. Roethig, E. Melcher, and M. Dana, 'Impact of Correlations and Delay on Switching Activity of Logic Circuits: A Validation Example for Probabilistic Power Consumption Estimation', in *Proc. 1993 Intl. Workshop on Power and Timing Modelling and Optimization, France*

[12] A. Papoulis, 'Probability, Random Variables, and Stochastic Processes', McGraw-Hill Co., 1984

[13] P. H. Bardell, W. H. McAnney, and J. Savir, 'Built-in Test for VLSI: Pseudorandom Techiques', J. Wiley & Sons Inc. 1987

[14] E. M. Sentovich, K. J. Singh, C. Moon, H. Savoj, and R. K. Brayton and A. Sangiovanni-Vincentelli, 'Sequential Circuit Design Using Synthesis and Optimization', in *Proc. 1992 ICCD*