

# On the Computation of Criticality in Statistical Timing Analysis

S.Ramprasath

Department of Electrical Engineering  
Indian Institute of Technology Madras  
ee10s056@ee.iitm.ac.in

V.Vasudevan

Department of Electrical Engineering  
Indian Institute of Technology Madras  
vinita@ee.iitm.ac.in

**Abstract**—Due to the statistical nature of gate delays in current day technologies, measures such as path criticality and node/edge criticality are required for timing optimization. Node criticalities are usually computed using the complementary path delay. In order to speed up computations, it has been recently proposed that the circuit delay be used instead. In this paper, we show that there is a monotonic relationship between the node criticalities computed using the circuit delay and the complementary delay. They are not equal, but they can be used interchangeably. We discuss the sources of error in this computation and propose methods for more accurate computations. We also introduce a measure that is very easy to compute and is an approximate indicator of criticality. Since it is easy to compute, it can also be used effectively for pruning the number of edges involved in criticality computations thus improving the speed of criticality computations. The speedup obtained can be as large as an order of magnitude for some of larger circuits in the ISCAS benchmarks.

**Index Terms**—Statistical timing, Criticality, Statistical distance

## I. INTRODUCTION

With the advent of the deep sub-micron technologies, increasing process parameter variations have changed the fundamental requirements of timing analysis tools. Path delays are statistical and are better described by a probability density function (PDF). As a result, one of several paths could potentially become critical on a particular die thereby making it difficult to optimise a circuit for better timing yield. Measures that are used for optimization of the circuit are path criticality [1], [4], [13]–[15] and node/edge criticality [1]–[4], [7]. The path criticality is the probability that a path is a critical path. Node/edge criticality is the probability that the node/edge belongs to atleast one dominant critical path. Node/edges with large criticality values are candidates for optimization. In this paper, we focus on the computation of node/edge criticality.

Node/edge criticality computations are performed after a forward and reverse traversal of the timing graph to obtain the arrival time (AT) and required time (RT) at each node/edge. Typically, the method used is the cutset based method [2], [5], in which the first step is to identify all the cutsets of the timing graph. Node/edge criticality is computed by finding the probability that the path delay (sum of AT and RT) associated with a particular node/edge is greater than the statistical maximum of the path delays associated with all the other edges of a cutset (complementary path delay). This is done for each cutset. The algorithm for computation of criticality has linear

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

IEEE/ACM International Conference on Computer-Aided Design (ICCAD) 2012, November 5-8, 2012, San Jose, California, USA

Copyright ©2012 ACM 978-1-4503-1573-9/12/11... \$15.00

time complexity with respect to the number of edges in the timing graph [2], [3]. However, a straightforward implementation results in significant inaccuracies. Some pruning of nodes/edges in a cutset and possibly a localised Monte-Carlo analysis is required before the evaluation of the criticalities [3]. To reduce the time required to compute the node/edge criticality as well as path criticality, the circuit delay has been used instead of the complementary path delay in [6], [7], [15]

In this paper, we look at the issue of computing node/edge criticalities. We make two contributions. First, we take a relook at the method for criticality computation proposed in [6]. We show that if Clark's formula is used, the node criticality computed using the identity relationship cannot have the same value as the node criticality computed using the conventional definition. However, there is a monotonic relationship between the criticalities computed using the two definitions and they can be used interchangeably. We also discuss the sources of error in this computation and methods to improve the accuracy of the method.

Our second contribution is to explore the use of a statistical distance measure as an indicator of node criticality. Ideally, we would like a measure that varies monotonically with the conventional definition of criticality, but is easily computable. We propose a measure that is based on the Kullback-Leibler (KL) divergence for Gaussian PDFs. It is only approximately monotonic, but it is a good indicator of criticality and is very easy to compute. As a result, it can be used very effectively as a preliminary step to prune the number of edges in each cutset, thus improving the speed of criticality computations.

The paper is organized as follows. Section II contains background details and definitions. Section III explores the computation of node criticality using the circuit delay and the results of this computation. In section IV we derive a measure that is easily computable and show that it can be used as an approximate indicator of criticality and can be used for pruning. It also contains the results obtained using this measure. Section V concludes the paper.

## II. BACKGROUND

This section contains the background details and defines the terms used in criticality computations.

The *Timing graph* is a directed acyclic graph  $G(V,E)$  comprising of nodes ( $V$ ) and edges ( $E$ ). The nodes represent the gate terminals. They are connected using edges that have an associated delay. This delay is a random variable with an associated PDF. The timing graph is augmented with a virtual source and sink node. The virtual source node is connected to all the primary inputs and all the outputs are connected to the virtual sink node.

A *Cutset*  $\Sigma$  divides the timing graph into two disjoint sets. Each cutset is a set of nodes/edges such that every path from the source

to the sink is associated with exactly one member of each cutset.

A forward and reverse SSTA performed on the timing graph gives the *arrival time* ( $AT$ ) and the *required time* ( $RT$ ) at a particular node. As the name suggests,  $AT$  ( $RT$ ) is the latest time at which a signal transition at the input (output) reaches the current node.  $AT$  at every node is the maximum of the arrival times at its incoming edges.  $RT$  at every node is computed using the maximum of the required times at the outgoing edges. The path delay  $e_i$  associated with a node/edge is given by

$$e_i = AT_i + RT_i \quad (1)$$

It represents the statistical maximum of the delays of all the paths between the virtual source and sink nodes passing through the node. The *complementary path delay* ( $e_{i'}$ ) of a node or edge within a cutset is defined as follows

$$e_{i'} = MAX(e_j, \forall e_j : e_j \in \Sigma, e_j \neq e_i)$$

Process parameter variations are typically correlated and the quad-tree model is used to model this [8]. All edge delays,  $AT$ 's and  $RT$ 's are represented in the canonical form [1] using the principal components (PC)  $p_j$  and are written as

$$e_i = \mu_i + \sum_{j=1}^k a_{ij} p_j + \xi_i q_i$$

Here, the weight  $a_{ij}$  is the sensitivity of  $e_i$  with respect to PC  $p_j$  and  $k$  is the total number of PCs. The mean of the distribution is  $\mu_i$ .  $q_i$  is uncorrelated with all the other principal components and represents an independent process variation with sensitivity  $\xi_i$ .

SSTA involves a  $SUM$  and  $MAX$  operation at every step. The  $SUM$  is a linear operation and the sum of two Gaussian random variables is Gaussian. However,  $MAX$  is a nonlinear operator and Clark's formula [10] is used to approximate the result of a  $MAX$  operation as a Gaussian random variable. If  $x$  and  $y$  are two Gaussian random variables, the approximation  $z$  to  $MAX(x, y)$  is obtained by matching the moments. The mean, standard deviation and the correlation with another random variable  $w$  is given as follows

$$x = \mathcal{N}(\mu_x, \sigma_x); y = \mathcal{N}(\mu_y, \sigma_y); \rho_{xy} = \rho$$

$$z = \mathcal{N}(\mu_z, \sigma_z) = MAX(x, y)$$

$$a = \sqrt{\sigma_x^2 + \sigma_y^2 - 2\rho\sigma_x\sigma_y} \quad (2)$$

$$\alpha = \left( \frac{\mu_x - \mu_y}{a} \right) \quad (3)$$

$$\mu_z = \mu_x \Phi(\alpha) + \mu_y \Phi(-\alpha) + a\phi(\alpha) \quad (4)$$

$$\sigma_z^2 = (\mu_x^2 + \sigma_x^2)\Phi(\alpha) + (\mu_y^2 + \sigma_y^2)\Phi(-\alpha) + (\mu_x + \mu_y)a\phi(\alpha) - \mu_z^2 \quad (5)$$

$$\rho_{wz} = \left( \frac{\sigma_1 \rho_{wx} \Phi(\alpha) + \sigma_2 \rho_{wy} \Phi(-\alpha)}{\sigma_z} \right) \quad (6)$$

$$\Phi(-\alpha) = 1 - \Phi(\alpha)$$

Here,  $\phi$  is the normalized Gaussian PDF  $\mathcal{N}(0, 1)$  and  $\Phi$  is the cumulative distribution function (CDF) of  $\phi$ . Clark's formula can be adapted to use the principal components.

The *local criticality* or the tightness probability of a node  $e_i$  with respect to  $e_j$  is defined as

$$\Phi(\alpha_{local(ij)}) = P(e_i \geq e_j) \quad (7)$$

The *global criticality* of a node  $e_i$  is defined as

$$\Phi(\alpha_{global(i)}) = P(e_i \geq e_{i'}) \quad (8)$$

The appropriate values of the mean, standard deviation and correlation coefficient are used to evaluate  $\alpha_{ij}$  and  $\alpha_i$  according to (3). The definition of criticality is thus based on the Gaussian approximation, since it uses the CDF  $\Phi(\alpha)$  to determine the required probability.

### III. CRITICALITY OBTAINED FROM THE CIRCUIT DELAY

To reduce the time required for criticality computations, [6] proposes using the following identity.

$$P(A \geq B) = P(A \geq A \& A \geq B) = P(A \geq MAX(A, B)) \quad (9)$$

where  $A$  and  $B$  are two random variables with arbitrary distributions. If this identity is used, it is possible to use the circuit delay instead of the complementary path delay to compute the criticality, thus reducing the time required for the computation.

Equation (9) is valid as long as the  $MAX$  operation used is ideal. However, the  $MAX$  operation used in SSTA is an approximation using the Clark's formulation. In this case equation (9) does not hold, as explained below.

Using the same notations as in the previous section, if  $z = MAX(x, y)$ , we have

$$\mu_x \leq \mu_z \quad (10)$$

A simple argument shows that equation (10) must be true. Let us generate two random variates  $X$  and  $Y$  with the distributions of  $x$  and  $y$  respectively. Since  $Z = MAX(X, Y)$ ,  $Z \geq X$  and  $Z \geq Y$ . Therefore,  $E[Z] \geq E[X]$  i.e.,  $\mu_x \leq \mu_z$ . This inequality also holds when Clark's formula is used, as shown in the Appendix A.

Since all the random variables are normal,

$$P(x \geq y) = \Phi\left(\frac{\mu_x - \mu_y}{\sqrt{\sigma_x^2 + \sigma_y^2 - 2\rho_{xy}\sigma_x\sigma_y}}\right) = \Phi(\alpha) \quad (11)$$

With  $z = MAX(x, y)$ , the probability that  $x \geq z$  is given by

$$P(x \geq z) = \Phi\left(\frac{\mu_x - \mu_z}{\sqrt{\sigma_x^2 + \sigma_z^2 - 2\rho_{xz}\sigma_x\sigma_z}}\right) = \Phi(\beta) \quad (12)$$

From equations (10) and (12)

$$P(x \geq z) \leq \Phi(0) \implies P(x \geq z) \leq 0.5 \quad (13)$$

Therefore  $P(A \geq B)$  is given by  $\Phi(\alpha)$ , which has a range between 0 and 1. But the  $P(A \geq MAX(A, B))$  or  $\Phi(\beta)$  has a range between 0 and 0.5. This is why the second method proposed in [6] requires a reconstruction formula.

The inequality (13) is valid at all points except when  $z = x$  i.e., when the two have identical means and standard deviations and the correlation is one. This anyway cannot be handled by Clark's formulation. In this case, however,  $x$  is clearly the dominant random variable and its criticality can be assigned. If the correlation is even slightly less than one, the probability will be close to 0.5. To have continuity, it makes sense to assign the value of 0.5 when the correlation is one.

#### A. Closed form expression and Monotonicity

Using the definitions of  $\alpha$  and  $\beta$  in equations (11) and (12), after some manipulations of Clark's formula, it can be shown that

$$\beta = \frac{\alpha\Phi(-\alpha) - \phi(\alpha)}{\sqrt{[\Phi(-\alpha) + (\alpha\Phi(\alpha) + \phi(\alpha))(\alpha\Phi(-\alpha) - \phi(\alpha))]} \quad (14)$$

Instead of solving this equation to get  $\Phi(\alpha)$  from  $\Phi(\beta)$ , we show that there is a one-to-one relationship between the two and they can be used interchangeably. We prove it by showing that  $\beta$  is a strictly increasing function of  $\alpha$ .

*Proof:* Let  $t_1$  be the numerator and  $t_2$  be the denominator in equation (14). For  $F(\alpha)$  as defined in Appendix A,  $t_1 = -F(\alpha)$ . Therefore,

$$\frac{dt_1}{d\alpha} = \Phi(-\alpha) > 0 \quad (15)$$

Since  $\Phi$  is the CDF of Gaussian distribution, it is strictly greater than zero for all  $\alpha < \infty$ . Therefore, the numerator is a strictly increasing function of  $\alpha$ .

Substituting for  $\Phi(\alpha)$  in terms of  $\Phi(-\alpha)$  and using the definition of  $t_1$ ,  $t_2^2$  can be written as

$$t_2^2 = \Phi(-\alpha) + t_1(\alpha - t_1)$$

The derivative of  $t_2^2$  with respect to  $\alpha$  is given by

$$\frac{dt_2^2}{d\alpha} = 2\Phi(\alpha)t_1 \quad (16)$$

Now, from (4) and Appendix A we get

$$\mu_x - \mu_z = at_1 \quad (17)$$

Since  $a > 0$ , from Appendix A, we get

$$t_1 < 0$$

From equation (16), the denominator decreases with increasing values of  $\alpha$ . Therefore the ratio of the two is a strictly increasing function of  $\alpha$ . Thus the map is a one-to-one map. ■

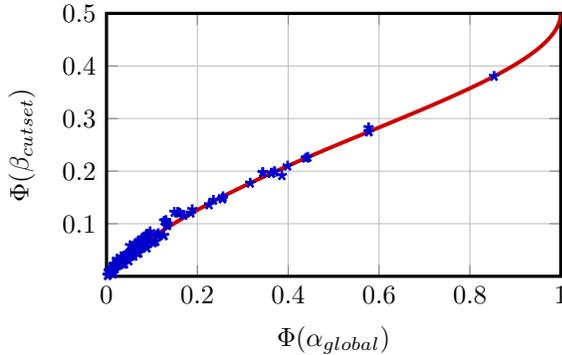


Fig. 1.  $\Phi(\alpha)$  vs  $\Phi(\beta)$ ; The red line is obtained from the analytical formula and the blue points are  $\Phi(\beta)$  computed using the cutset maximum for a carry select adder.

Fig. 1 contains a plot of  $\Phi(\beta)$  as a function of  $\Phi(\alpha)$  obtained using the analytical formula. It is evident that it is monotonic and its range is  $[0, 0.5]$ . Since the map is monotonic, it is possible to use  $\Phi(\beta)$  instead of  $\Phi(\alpha)$  as a measure of criticality or create a lookup table from equation (14) to get back  $\Phi(\alpha)$  from  $\Phi(\beta)$ .

#### B. Computation of $\Phi(\beta)$

$\Phi(\beta)$  can be computed using the circuit delay obtained after doing an SSTA. Ideally, the maximum of the path delays of any cutset must be equal to the circuit delay. However, practically there is a 1 – 2% difference in the maximum across cutsets. Since  $\Phi(\alpha)$  is defined relative to a cutset, this difference results in an error in the  $\Phi(\alpha)$  computed using  $\Phi(\beta)$ . It can be seen from Fig. 1 that over a wide range of  $\Phi(\alpha)$ , the slope of the function is less than one which

implies small differences in  $\Phi(\beta)$  can result in large errors in  $\Phi(\alpha)$  computed using  $\Phi(\beta)$ .

This error is significant, unless  $\Phi(\beta)$  is defined in each cutset with respect to the maximum of that particular cutset. If this is done, the speedup obtained is not as large. But there is still a reduction in the number of *MAX* operations. Assuming that evaluation of the tightness probability and the *MAX* operation take almost the same time, if linear-time bookkeeping [3] is used, a total of  $(4n - 2)$  *MAX* operations are required to compute global criticality using complementary path delay. Here  $n$  is the number of nodes in a cutset. If  $\Phi(\beta)$  is computed using the cutset maximum,  $(2n - 1)$  *MAX* operations are required. Getting back  $\Phi(\alpha)$  from  $\Phi(\beta)$  involves a lookup operation in a sorted array, which is a constant overhead. Therefore, overall, there is still some savings in time possible.

#### C. Results

Fig. 1 shows  $\Phi(\beta)$  obtained using the cutset maximum for a carry select adder. It can be seen to match the analytical curve quite closely. Similar plots are seen for the other circuits. Tables I & II show the error in global criticality ( $\Phi(\alpha_{global})$ ) obtained from  $\Phi(\beta)$  computed using the cutset maximum. The circuits were synthesized using the 180nm UMC libraries. The K-center pruning algorithm proposed in [3] was used to improve the accuracy of the computations. The following two quad-tree structures QT1 and QT2 for  $V_T$ ,  $L$  and  $W$  of gates were used.

QT1: 5 layer quad-tree with 90% of the variation in  $V_T$ ,  $L$  and  $W$  of the gates is assigned to the bottom most layer and 2.5% each to the rest of the layers, so that the correlation is largely due to the circuit topology.

QT2: Each of the five quad-tree layers get 20% of the variation in the three parameters.

A 10% variation from the nominal value was considered for each of the parameters for the simulations. The lookup table for obtaining  $\Phi(\alpha_{global})$  from  $\Phi(\beta)$  had 10000 values and was generated off-line using the equation (14).

From the tables it is clear that we can get upto 30% reduction in run-time on using  $\Phi(\beta)$  (with respect to cutset max) without significant loss in accuracy. The run-times include both pruning and then finding global criticality for the un-pruned nodes. The low speedup obtained in some of the circuits is due to the small number of nodes remaining after pruning.

The accuracy of the computation is comparable to what has been reported in [3], [15]. In some of the benchmarks (c6288,s38584 for example), the maximum error is a little large in all three cases. This is also seen in the results of [3], [15]. Monte Carlo sampling was used in [3] to reduce the error. We have found that error can be reduced by changing the order in which the *MAX* operations are performed. We have implemented the Cluster Max Binary tree (CMBT) technique proposed in [11]. Table III shows that there is a considerable improvement on using CMBT to find the cutset maximum. The drawback of using CMBT is its run-time of  $O(n^2 \log(n))$ . The run time costs of using CMBT algorithm for criticality computation using complementary path delays is prohibitive, but it could possibly be used to accurately find the cutset maximum.

#### IV. STATISTICAL DISTANCE

In this section, we arrive at an easily computable measure that can be used as an approximate indicator of criticality. To do this we start with the Kullback-Leibler (KL) divergence between the path delay associated with each node and the circuit delay. Intuitively, if this divergence is small, the PDF of the path delay associated with a node

TABLE I

ISCAS85 BENCHMARKS: [PS1] –  $\Phi(\alpha_{global})$  FOUND FROM THE COMPLEMENTARY PATH DELAY USING LINEAR TIME BOOKKEEPING TECHNIQUE; [PS2] –  $\Phi(\alpha_{global})$  FOUND FROM  $\Phi(\beta)$  W.R.T CIRCUIT DELAY; [PS3] –  $\Phi(\alpha_{global})$  FOUND FROM  $\Phi(\beta)$  W.R.T CORRESPONDING CUTSET MAXES; RMS AND MAX INDICATE THE ROOT MEAN SQUARED AND MAXIMUM ERROR IN  $\Phi(\alpha_{global})$  FOR EACH OF THE CASES; SPEEDUP IS THE RATIO OF RUN-TIMES, E.G. FOR [PS2]. SPEEDUP =  $\frac{T([PS1])}{T([PS2])}$ ; FOR ALL THE THREE CASES THE CUTSETS WERE PRUNED USING K-CENTER CLUSTERING [3]

| Benchmark          | Quad-tree QT1 |      |       |      |         |       |      |         |      | Quad-tree QT2 |      |       |         |      |       |         |  |  |
|--------------------|---------------|------|-------|------|---------|-------|------|---------|------|---------------|------|-------|---------|------|-------|---------|--|--|
|                    | [PS1]         |      | [PS2] |      |         | [PS3] |      |         |      | [PS1]         |      | [PS2] |         |      | [PS3] |         |  |  |
|                    | RMS           | max  | RMS   | max  | Speedup | RMS   | max  | Speedup | RMS  | max           | RMS  | max   | Speedup | RMS  | max   | Speedup |  |  |
| c432               | 0.01          | 0.02 | 0.13  | 0.38 | 1.59    | 0.01  | 0.05 | 1.25    | 0.02 | 0.06          | 0.13 | 0.34  | 1.56    | 0.02 | 0.06  | 1.28    |  |  |
| c499               | 0.02          | 0.05 | 0.04  | 0.10 | 1.69    | 0.05  | 0.16 | 1.40    | 0.03 | 0.09          | 0.05 | 0.12  | 1.73    | 0.10 | 0.24  | 1.41    |  |  |
| c880               | 0.00          | 0.01 | 0.01  | 0.02 | 1.05    | 0.00  | 0.01 | 1.04    | 0.02 | 0.05          | 0.02 | 0.05  | 1.02    | 0.02 | 0.05  | 1.03    |  |  |
| c1355              | 0.02          | 0.08 | 0.05  | 0.19 | 1.55    | 0.04  | 0.11 | 1.33    | 0.02 | 0.09          | 0.06 | 0.21  | 1.55    | 0.05 | 0.14  | 1.32    |  |  |
| c1908              | 0.02          | 0.12 | 0.07  | 0.24 | 1.69    | 0.03  | 0.13 | 1.35    | 0.03 | 0.21          | 0.10 | 0.34  | 1.65    | 0.04 | 0.22  | 1.34    |  |  |
| c2670              | 0.02          | 0.05 | 0.02  | 0.05 | 1.03    | 0.02  | 0.05 | 1.03    | 0.03 | 0.07          | 0.03 | 0.07  | 1.07    | 0.03 | 0.07  | 1.04    |  |  |
| c3540              | 0.02          | 0.08 | 0.05  | 0.17 | 1.39    | 0.02  | 0.08 | 1.22    | 0.02 | 0.11          | 0.04 | 0.16  | 1.28    | 0.03 | 0.13  | 1.14    |  |  |
| c5315              | 0.03          | 0.08 | 0.07  | 0.15 | 1.31    | 0.03  | 0.08 | 1.20    | 0.05 | 0.10          | 0.11 | 0.17  | 1.19    | 0.06 | 0.15  | 1.13    |  |  |
| c6288              | 0.02          | 0.11 | 0.03  | 0.24 | 1.34    | 0.02  | 0.17 | 1.20    | 0.03 | 0.20          | 0.05 | 0.31  | 1.37    | 0.04 | 0.24  | 1.23    |  |  |
| c7552              | 0.00          | 0.00 | 0.00  | 0.00 | 1.02    | 0.00  | 0.00 | 1.03    | 0.00 | 0.00          | 0.00 | 0.00  | 1.01    | 0.00 | 0.00  | 1.02    |  |  |
| block-carry adder  | 0.01          | 0.05 | 0.02  | 0.06 | 1.83    | 0.01  | 0.03 | 1.47    | 0.02 | 0.06          | 0.03 | 0.08  | 1.85    | 0.02 | 0.05  | 1.47    |  |  |
| carry-bypass adder | 0.02          | 0.07 | 0.06  | 0.30 | 1.28    | 0.02  | 0.10 | 1.18    | 0.04 | 0.11          | 0.07 | 0.26  | 1.19    | 0.04 | 0.13  | 1.09    |  |  |
| carry-select adder | 0.03          | 0.11 | 0.03  | 0.18 | 1.65    | 0.03  | 0.12 | 1.37    | 0.05 | 0.20          | 0.06 | 0.41  | 1.70    | 0.04 | 0.18  | 1.39    |  |  |

TABLE II  
ISCAS89 BENCHMARKS

| Benchmark | Quad-tree QT1 |      |                     |      |         |                        |      |         |      | Quad-tree QT2 |      |                     |         |      |                        |         |  |  |
|-----------|---------------|------|---------------------|------|---------|------------------------|------|---------|------|---------------|------|---------------------|---------|------|------------------------|---------|--|--|
|           | CPD           |      | $\Phi(\beta_{ckt})$ |      |         | $\Phi(\beta_{cutset})$ |      |         |      | CPD           |      | $\Phi(\beta_{ckt})$ |         |      | $\Phi(\beta_{cutset})$ |         |  |  |
|           | RMS           | max  | RMS                 | max  | Speedup | RMS                    | max  | Speedup | RMS  | max           | RMS  | max                 | Speedup | RMS  | max                    | Speedup |  |  |
| s953      | 0.01          | 0.02 | 0.01                | 0.02 | 1.02    | 0.01                   | 0.02 | 1.02    | 0.02 | 0.05          | 0.02 | 0.05                | 1.07    | 0.02 | 0.05                   | 1.04    |  |  |
| s1196     | 0.03          | 0.08 | 0.08                | 0.21 | 1.53    | 0.03                   | 0.08 | 1.27    | 0.04 | 0.10          | 0.14 | 0.35                | 1.52    | 0.03 | 0.08                   | 1.26    |  |  |
| s1238     | 0.03          | 0.11 | 0.08                | 0.17 | 1.44    | 0.03                   | 0.11 | 1.23    | 0.04 | 0.16          | 0.09 | 0.20                | 1.47    | 0.05 | 0.14                   | 1.24    |  |  |
| s1423     | 0.02          | 0.05 | 0.02                | 0.05 | 1.01    | 0.02                   | 0.05 | 1.02    | 0.00 | 0.00          | 0.00 | 0.00                | 1.00    | 0.00 | 0.00                   | 1.01    |  |  |
| s1488     | 0.03          | 0.09 | 0.03                | 0.07 | 1.41    | 0.03                   | 0.05 | 1.21    | 0.02 | 0.05          | 0.02 | 0.03                | 1.28    | 0.02 | 0.04                   | 1.16    |  |  |
| s1494     | 0.01          | 0.02 | 0.01                | 0.02 | 1.03    | 0.01                   | 0.02 | 1.03    | 0.03 | 0.04          | 0.03 | 0.04                | 1.02    | 0.03 | 0.04                   | 1.02    |  |  |
| s5378     | 0.02          | 0.07 | 0.08                | 0.18 | 1.17    | 0.02                   | 0.05 | 1.14    | 0.04 | 0.13          | 0.13 | 0.33                | 1.15    | 0.03 | 0.11                   | 1.10    |  |  |
| s9234     | 0.01          | 0.02 | 0.02                | 0.04 | 1.14    | 0.01                   | 0.03 | 1.13    | 0.04 | 0.07          | 0.06 | 0.14                | 1.17    | 0.04 | 0.07                   | 1.12    |  |  |
| s13207    | 0.02          | 0.07 | 0.02                | 0.12 | 1.23    | 0.01                   | 0.04 | 1.15    | 0.04 | 0.14          | 0.03 | 0.10                | 1.23    | 0.02 | 0.06                   | 1.16    |  |  |
| s15850    | 0.02          | 0.04 | 0.02                | 0.06 | 1.02    | 0.01                   | 0.04 | 1.01    | 0.02 | 0.04          | 0.02 | 0.06                | 1.01    | 0.02 | 0.03                   | 1.01    |  |  |
| s35932    | 0.00          | 0.04 | 0.01                | 0.15 | 1.34    | 0.00                   | 0.03 | 1.22    | 0.00 | 0.04          | 0.01 | 0.15                | 1.32    | 0.00 | 0.03                   | 1.24    |  |  |
| s38417    | 0.02          | 0.06 | 0.12                | 0.18 | 1.09    | 0.02                   | 0.07 | 1.06    | 0.02 | 0.06          | 0.14 | 0.20                | 1.05    | 0.03 | 0.09                   | 1.04    |  |  |
| s38584    | 0.07          | 0.18 | 0.09                | 0.24 | 1.01    | 0.08                   | 0.23 | 1.00    | 0.10 | 0.26          | 0.13 | 0.36                | 1.00    | 0.12 | 0.38                   | 1.00    |  |  |

is more nearly equal to the PDF of the circuit delay implying that the node has a larger criticality. Similarly, a large divergence would mean a low criticality. For Gaussian PDFs, there is an analytical formula for the KL divergence, which makes it easily computable. If  $\mathcal{N}(\mu_i, \sigma_i)$  is the PDF of the path delay associated with the  $i^{th}$  node/edge and the PDF of the circuit delay is  $\mathcal{N}(\mu_c, \sigma_c)$ , the KL divergence between node  $i$  and circuit delay is given by

$$D_{KL} = \frac{(\mu_c - \mu_i)^2}{2\sigma_c^2} + \frac{\sigma_i^2 - \sigma_c^2}{2\sigma_c^2} + \frac{1}{2} \log \left( \frac{\sigma_c^2}{\sigma_i^2} \right) \quad (18)$$

One drawback is that the KL divergence has a very large dynamic range, at times, over four orders of magnitude. The range also varies significantly as the correlation structure of the parameters changes. However, it is possible to make some simplifications and come up with a measure that is a function of the approximate KL divergence and has a range between zero and one.

From Appendices A and B,  $\mu_c > \max(\mu_i, \mu_{i'})$  and  $\sigma_c < \max(\sigma_i, \sigma_{i'})$ , where the subscript  $i'$  denotes the mean and the standard deviation of the complementary path delay. Based on these results, we can show that the ratio  $\frac{\sigma_i}{\mu_i}$  has an upper bound which is the maximum value of  $\frac{\sigma_D}{\mu_D}$  over all gates in the circuit, where  $\sigma_D$

TABLE III  
IMPROVEMENT IN ERROR ON USING CLUSTER MAX BINARY TREE TECHNIQUE(CMBT) [11] TO FIND THE MAXIMUM OF CUTSETS; W/O CMBT – ARE THE ERRORS FROM [PS3] OF TABLES I AND II; CMBT – USING THE SAME PRUNING SCHEME BUT USING CMBT TO FIND THE MAXIMUM OF CUTSETS

| Benchmark     | w/o CMBT |      | CMBT |      |
|---------------|----------|------|------|------|
|               | RMS      | max  | RMS  | max  |
| Quad-tree QT1 |          |      |      |      |
| c499          | 0.05     | 0.16 | 0.01 | 0.04 |
| c6288         | 0.02     | 0.17 | 0.02 | 0.11 |
| s38584        | 0.08     | 0.23 | 0.04 | 0.11 |
| Quad-tree QT2 |          |      |      |      |
| c499          | 0.10     | 0.24 | 0.02 | 0.05 |
| c1355         | 0.05     | 0.14 | 0.02 | 0.08 |
| c5315         | 0.06     | 0.15 | 0.04 | 0.08 |
| c6288         | 0.04     | 0.24 | 0.03 | 0.16 |
| s38584        | 0.12     | 0.38 | 0.08 | 0.19 |

and  $\mu_D$  represent the standard deviation and the mean delay of a gate. From the results of appendices C and D, both the *SUM* and *MAX* operations result in a PDF that has a  $\frac{\sigma}{\mu}$  that is bounded by the maximum  $\frac{\sigma}{\mu}$  of their arguments. Therefore, for each gate, the *SUM*

and  $MAX$  operator result in a  $\frac{\sigma}{\mu}$  that is bounded by  $\max\{\frac{\sigma_j}{\mu_j}\}$ , with the index  $j$  running over the inputs to the gate and the delay from each input to the output. Starting from the source node, if we look at the bound at each level, it is clear that the ratio has an upper bound which is the maximum value of  $\frac{\sigma_D}{\mu_D}$  over all gates in the circuit.

For current day technologies, the standard deviation of the gate delay is a very small fraction of the delay itself. The upper bound therefore is quite small. If we assume a 10% variation in the threshold voltage, lengths and widths,  $\frac{\sigma_D}{\mu_D} \approx 0.17$  (for a simplified model in which the sensitivity to each source of variation is proportional to the delay). In the discussions that follow, we do not require the actual value of this ratio. However we assume that it is much smaller than one.

From the above arguments,  $\frac{\mu_c}{\sigma_c} > \frac{\mu_i}{\sigma_c} \gg \frac{\sigma_i}{\sigma_c}$ . The second term in equation (18) is therefore much smaller than the first term and can be neglected. The third term in  $D_{KL}$  is significant only if  $\sigma_i$  is significantly different from  $\sigma_c$ . In this case  $\mu_i$  will also be significantly different from  $\mu_c$  and the criticality will consequently be quite low. Assuming that we do not require the exact values of  $D_{KL}$  for low criticality nodes,  $D_{KL}$  can be simplified to

$$D_{KL} \approx \frac{(\mu_c - \mu_i)^2}{2\sigma_c^2}$$

Fig. 2 shows the approximate  $D_{KL}$  as a function of the full expression for  $D_{KL}$  for two different quad-trees. QT3 has 90% of the variation in the topmost layer of the quad-tree which implies the parameter variations are highly correlated. QT1 and QT3 captures the behaviour in the two extreme cases. It is clear that a deviation occurs only at larger values of  $D_{KL}$ , for which the criticality is low. Independent of the quad-tree, we find that  $D_{KL} < 1$  if  $\Phi(\alpha_{global}) > 0.1$ .

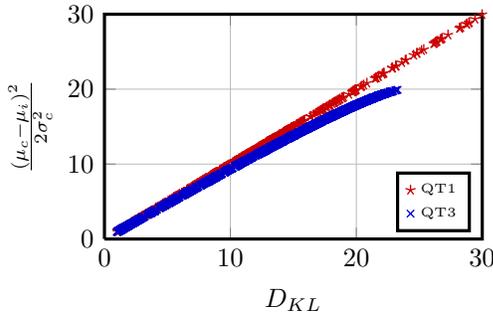


Fig. 2.  $\frac{(\mu_c - \mu_i)^2}{2\sigma_c^2}$  versus  $D_{KL}$  for all nodes in s13207 benchmark for two quad-trees

As  $\sigma_c$  is constant for a given circuit, it is simply a scaling factor. The qualitative behaviour will not change if  $\sigma_c$  is replaced by  $\mu_c$  and the factor of 2 is ignored, so that the range of this “distance” is  $[0, 1]$ . As  $(\mu_c - \mu_i)$  (from Appendix A) is always non-negative,

$$D = \frac{(\mu_c - \mu_i)}{\mu_c} \quad (19)$$

can be used in place of  $D_{KL}$ .

At first sight, it seems as though  $\sigma_i$  and the correlation with other nodes are ignored entirely in this measure. However, both  $\sigma$  and  $\rho$  are required for computation of  $\mu_c$ . Hence they are taken into account implicitly. But it is not possible to get the same resolution in criticality that can be obtained using the complementary path delay or the circuit delay. The following section discusses this.

#### A. $D$ as an indicator of criticality

Let  $\mathcal{N}(\mu_i, \sigma_i)$  be the path delay through node  $i$  and  $\mathcal{N}(\mu_{i'}, \sigma_{i'})$  be the complementary path delay for node  $i$ . From equation (4) and Appendix A,

$$\mu_c - \mu_i = aF(\alpha)$$

where  $a = \sqrt{\sigma_i^2 + \sigma_{i'}^2 - 2\rho_{ii'}\sigma_i\sigma_{i'}}$  and  $\alpha = \frac{\mu_i - \mu_{i'}}{a}$ . The maximum value of  $a$  occurs when  $\rho_{ii'} = -1$ , when  $a = \sigma_i + \sigma_{i'}$ . From appendices C and D and earlier arguments,  $\frac{\sigma_i}{\mu_i} < k$ , where  $k = \max\{\frac{\sigma_D}{\mu_D}\} \ll 1$ . Therefore

$$\frac{\mu_c - \mu_i}{\mu_c} \leq 2kF(\alpha) \quad (20)$$

$F(\alpha)$  decreases monotonically with  $\alpha$  and therefore decreases monotonically with  $\Phi(\alpha)$ . Hence the distance measure  $D$  has an upper bound which is monotonic with  $\Phi(\alpha)$ .

But  $D$  itself is not monotonic with respect to  $\Phi(\alpha)$ . More than one path can have the same  $\mu$  but different  $\sigma$  and different correlation with circuit delay. Fig. 3 shows an example plot of  $\Phi(\alpha_{global})$  vs  $D$  for all nodes of c499 and the carry-bypass adder with quad-tree QT1.  $\Phi(\alpha_{global})$  was found using complementary path delay after

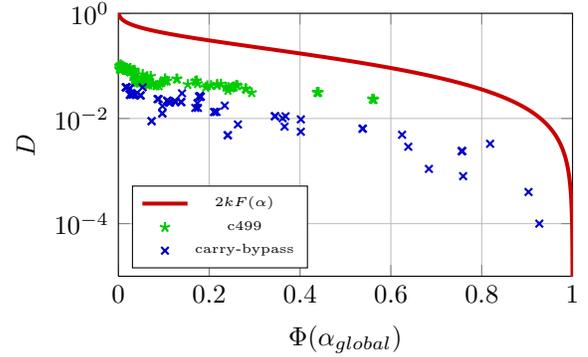


Fig. 3.  $D$  vs  $\Phi(\alpha_{global})$  for all nodes in c499 benchmark and the carry bypass adder for quad-tree QT1

the nodes were pruned using K-center clustering. The range of  $D$  in c499 is  $[0.1, 0.02]$  and is about two orders of magnitude for the carry bypass adder. The figure also shows the upper bound. It is a conservative estimate, as expected.

Although non-monotonic, it is clear that  $D$  can be used as an approximate indicator of criticality. One problem however is that nodes that have similar criticalities have different values of  $D$  in different circuits, making it difficult to map a fixed range of values of  $D$  to a particular value of  $\Phi(\alpha)$ . However, although termed “global criticality”, node criticality as a measure is only relevant within a cutset. If we use the same rationale for  $D$ , the ratio  $\frac{D}{D_{min}}$ , where  $D_{min}$  is the minimum value of  $D$  in the cutset, should be a reasonable indicator of relative criticality within the cutset. This is shown in the table below for a few cutsets in c499 and the carry bypass adder. The least critical nodes are easily identifiable by their relatively large values of  $\frac{D}{D_{min}}$  and can be used for pruning. The most critical node in a cutset will have  $\frac{D}{D_{min}} \approx 1$ . (Resolving  $\Phi(\alpha)$  when the difference in  $\frac{D}{D_{min}}$  between two nodes is less than two is not possible). But its relative criticality within the cutset can be inferred based on the maximum value of  $\frac{D}{D_{min}}$  within the cutset. However some non-monotonicities are present as seen in  $\sum_7$  of the carry bypass adder. Hence  $D$  is an approximate indicator of criticality.

TABLE IV  
 $\Phi(\alpha_{global})$  AND  $\frac{D}{D_{min}}$  OF NODES FROM CUTSETS  $\Sigma_i$  OF C499 AND  
 CARRY-BYPASS ADDER BENCHMARKS

| $\Phi(\alpha)$     | $\frac{D}{D_{min}}$ | $\Phi(\alpha)$ | $\frac{D}{D_{min}}$ | $\Phi(\alpha)$ | $\frac{D}{D_{min}}$ |
|--------------------|---------------------|----------------|---------------------|----------------|---------------------|
| carry-bypass adder |                     |                |                     |                |                     |
| $\Sigma_{15}$      |                     | $\Sigma_{33}$  |                     | $\Sigma_7$     |                     |
| 0.68               | 1.00                | 0.76           | 1.00                | 0.54           | 1.0                 |
| 0.24               | 4.38                | 0.11           | 8.54                | 0.17           | 2.5                 |
| 0.07               | 8.05                | 0.11           | 9.22                | 0.10           | 2.71                |
| —                  | —                   | 0.03           | 11.94               | 0.18           | 4.11                |
| c499               |                     |                |                     |                |                     |
| $\Sigma_5$         |                     | $\Sigma_6$     |                     | $\Sigma_{11}$  |                     |
| 0.28               | 1.00                | 0.28           | 1.00                | 0.29           | 1.00                |
| 0.26               | 1.17                | 0.26           | 1.17                | 0.24           | 1.11                |
| 0.25               | 1.09                | 0.25           | 1.10                | 0.24           | 1.27                |
| 0.17               | 1.41                | 0.17           | 1.43                | 0.20           | 1.36                |
| 0.01               | 2.46                | 0.01           | 2.47                | —              | —                   |

Based on these discussions, we have a two level pruning algorithm that prunes out most of the nodes that have low criticality. The remaining nodes can be pruned using K-center pruning, if required.

### B. Pruning

Pruning using  $D$  is a two step procedure. Firstly the least critical nodes in the circuit (nodes that have  $\mu \ll \mu_c$ ) are pruned. This step is independent of the correlation structure or the circuit. The second step involves pruning of nodes that are non-dominant within the cutset to which they belong.

Algorithm 1 describes the first level of pruning. We can get an approximate value for  $D_{thresh}$  using the threshold value of local criticality (which is an upper bound for the global criticality) used in K-center pruning. The value recommended in [3] is 0.05. Using the corresponding value of  $\alpha$  and  $k = 0.17$  (for 10% variation in  $V_T$ ,  $L$  and  $W$ ), we get a pruning threshold of  $D_{thresh} = 0.53$ . Nodes were pruned using different values of  $D_{thresh}$  with the quad-tree QT1. This quad-tree was chosen since the correlations in the circuit will be mostly due to circuit topology rather than the technology. The ISCAS85 benchmark contains circuits that capture a wide variety of circuit topologies. Therefore a pruning strategy derived using QT1 and the ISCAS85 benchmark circuits should be applicable when used with other quad-trees and circuits.

The resulting error in  $\Phi(\alpha_{global})$  averaged over all circuits in the ISCAS85 benchmarks is plotted in Fig.4. The figure shows that a much smaller value of  $D_{thresh} = 0.15$  is sufficient. It will be seen from the results that this threshold can be used independent of the quad-tree and the circuit. The same threshold for example, could be used for the tree QT2 and circuits in ISCAS89 benchmarks.

#### Algorithm 1 Pruning using $D$

```

1: for cutset  $\Sigma_i$  in  $\{\Sigma_1, \Sigma_2 \dots \Sigma_n\}$  do
2:   for node  $j$  in  $\Sigma_i$  do
3:     if  $D_j > D_{thresh}$  then
4:       mark node  $j$  as pruned
5:     end if
6:   end for
7: end for

```

Algorithm 2 describes the second level of pruning within a cutset. In this case, the difference in the normalised distance between two nodes of a cutset is compared. Following a similar procedure as the first level of pruning, Fig. 5 shows the maximum error in  $\Phi(\alpha_{global})$  averaged over all ISCAS85 benchmarks. In this case, the threshold  $\Delta D_{thresh}$  was empirically found to be 14.

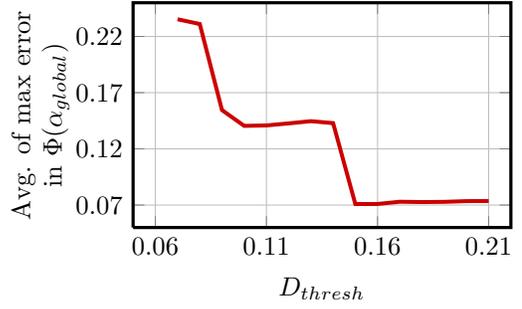


Fig. 4.  $D_{thresh}$  vs max error in  $\Phi(\alpha_{global})$  averaged over all ISCAS85 benchmarks for quad-tree QT1

#### Algorithm 2 Pruning using $\frac{\Delta D}{D_{min}}$

```

1: for cutset  $\Sigma_i$  in  $\{\Sigma_1, \Sigma_2 \dots \Sigma_n\}$  do
2:   for node  $j$  in  $\Sigma_i$  &  $j$  do
3:     for node  $k$  in  $\Sigma_i$ ,  $k \neq j$  do
4:       if  $\left(\frac{D_j - D_k}{D_{min_i}}\right) > \Delta D_{thresh}$  then
5:         mark node  $j$  as pruned
6:       else if  $\left(\frac{D_k - D_j}{D_{min_i}}\right) > \Delta D_{thresh}$  then
7:         mark node  $k$  as pruned
8:       end if
9:     end for
10:   end for
11: end for

```

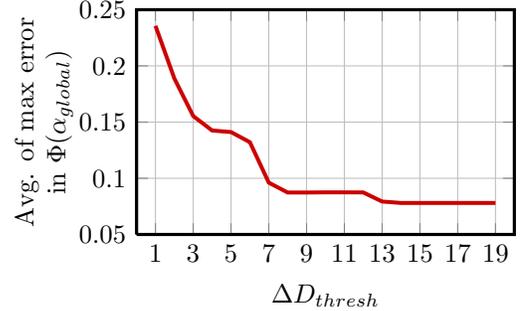


Fig. 5.  $\Delta D_{thresh}$  vs max error in  $\Phi(\alpha_{global})$  averaged over all ISCAS85 benchmarks for quad-tree QT1

Even after pruning using Algorithms 1 and 2, there could be some low criticality nodes that need to be pruned. These nodes are then removed using K-center clustering. The number of nodes left for the clustering step is usually much less than the original number of nodes in most circuits, resulting in a significant speedup for many circuits.

### C. Results

Tables V and VI show the error in  $\Phi(\alpha_{global})$  and the possible speedup for the following cases:

1. Cutsets are pruned using K-center clustering and  $\Phi(\alpha_{global})$  is computed using complementary path delay.
2. Cutsets are pruned using Algorithms 1 and 2. This is followed by K-center clustering and pruning on the remaining nodes.  $\Phi(\alpha_{global})$  is computed using the complementary path delay.
3. Cutsets are pruned using Algorithms 1 and 2. This is followed by K-center clustering and pruning on the remaining nodes.  $\Phi(\alpha_{global})$  is computed from  $\Phi(\beta_{cutset})$ .

TABLE V

ISCAS85 BENCHMARKS: [PS1] –  $\Phi(\alpha_{global})$  FOUND USING COMPLEMENTARY PATH DELAY AFTER PRUNING CUTSETS USING K-CENTER CLUSTERING; [PS2] –  $\Phi(\alpha_{global})$  FOUND USING COMPLEMENTARY PATH DELAY AFTER PRUNING THE CUTSETS USING  $D$  WITH ALGORITHM 1 AND 2 FOLLOWED BY K-CENTER CLUSTERING OF THE REMAINING NODES; [PS3] – PRUNING SCHEME SAME AS [PS2] BUT  $\Phi(\alpha_{global})$  FOUND FROM  $\Phi(\beta_{cutset})$ ; RMS AND MAX INDICATE THE ROOT MEAN SQUARED AND MAXIMUM ERROR IN  $\Phi(\alpha_{global})$  FOR EACH OF THE CASES; SPEEDUP IS THE RATIO OF RUN-TIMES FOR EACH CASES, I.E. FOR PS2 SPEEDUP =  $\frac{T([PS1])}{T([PS2])}$ ;

| Benchmark          | Quad-tree QT1 |      |       |      |         |       |      |         |       | Quad-tree QT2 |       |      |         |       |      |         |  |  |
|--------------------|---------------|------|-------|------|---------|-------|------|---------|-------|---------------|-------|------|---------|-------|------|---------|--|--|
|                    | [PS1]         |      | [PS2] |      |         | [PS3] |      |         | [PS1] |               | [PS2] |      |         | [PS3] |      |         |  |  |
|                    | RMS           | max  | RMS   | max  | Speedup | RMS   | max  | Speedup | RMS   | max           | RMS   | max  | Speedup | RMS   | max  | Speedup |  |  |
| c432               | 0.01          | 0.02 | 0.02  | 0.07 | 1.99    | 0.02  | 0.07 | 3.29    | 0.02  | 0.06          | 0.03  | 0.15 | 1.95    | 0.03  | 0.15 | 3.01    |  |  |
| c499               | 0.02          | 0.05 | 0.01  | 0.05 | 1.10    | 0.02  | 0.10 | 1.58    | 0.03  | 0.09          | 0.02  | 0.05 | 1.15    | 0.02  | 0.05 | 1.69    |  |  |
| c880               | 0.00          | 0.01 | 0.02  | 0.08 | 10.87   | 0.02  | 0.08 | 12.66   | 0.02  | 0.05          | 0.02  | 0.05 | 25.59   | 0.02  | 0.05 | 27.26   |  |  |
| c1355              | 0.02          | 0.08 | 0.02  | 0.08 | 1.10    | 0.02  | 0.09 | 1.49    | 0.02  | 0.09          | 0.02  | 0.10 | 1.04    | 0.02  | 0.07 | 1.53    |  |  |
| c1908              | 0.02          | 0.12 | 0.02  | 0.12 | 1.34    | 0.03  | 0.12 | 1.91    | 0.03  | 0.21          | 0.04  | 0.23 | 1.32    | 0.04  | 0.20 | 1.97    |  |  |
| c2670              | 0.02          | 0.05 | 0.02  | 0.05 | 11.96   | 0.02  | 0.05 | 15.01   | 0.03  | 0.07          | 0.03  | 0.07 | 8.88    | 0.03  | 0.07 | 7.41    |  |  |
| c3540              | 0.02          | 0.08 | 0.02  | 0.08 | 1.73    | 0.02  | 0.08 | 2.46    | 0.02  | 0.11          | 0.02  | 0.11 | 2.00    | 0.03  | 0.13 | 2.78    |  |  |
| c5315              | 0.03          | 0.08 | 0.03  | 0.08 | 2.56    | 0.03  | 0.13 | 3.75    | 0.05  | 0.10          | 0.05  | 0.10 | 4.13    | 0.07  | 0.21 | 5.91    |  |  |
| c6288              | 0.02          | 0.11 | 0.02  | 0.14 | 1.07    | 0.02  | 0.15 | 1.37    | 0.03  | 0.20          | 0.03  | 0.19 | 1.07    | 0.04  | 0.31 | 1.35    |  |  |
| c7552              | 0.00          | 0.00 | 0.00  | 0.00 | 88.77   | 0.00  | 0.00 | 119.44  | 0.00  | 0.00          | 0.00  | 0.00 | 88.30   | 0.00  | 0.00 | 127.38  |  |  |
| block-carry adder  | 0.01          | 0.05 | 0.01  | 0.05 | 1.08    | 0.01  | 0.03 | 1.12    | 0.02  | 0.06          | 0.02  | 0.07 | 1.09    | 0.02  | 0.06 | 1.61    |  |  |
| carry-bypass adder | 0.02          | 0.07 | 0.02  | 0.07 | 2.62    | 0.02  | 0.14 | 3.97    | 0.04  | 0.11          | 0.04  | 0.11 | 3.55    | 0.04  | 0.12 | 4.72    |  |  |
| carry-select adder | 0.03          | 0.11 | 0.03  | 0.11 | 1.31    | 0.03  | 0.17 | 2.02    | 0.05  | 0.20          | 0.05  | 0.20 | 1.28    | 0.05  | 0.24 | 1.98    |  |  |

TABLE VI  
ISCAS89 BENCHMARKS

| Benchmark | Quad-tree QT1 |      |       |      |         |       |      |         |       | Quad-tree QT2 |       |      |         |       |      |         |  |  |
|-----------|---------------|------|-------|------|---------|-------|------|---------|-------|---------------|-------|------|---------|-------|------|---------|--|--|
|           | [PS1]         |      | [PS2] |      |         | [PS3] |      |         | [PS1] |               | [PS2] |      |         | [PS3] |      |         |  |  |
|           | RMS           | max  | RMS   | max  | Speedup | RMS   | max  | Speedup | RMS   | max           | RMS   | max  | Speedup | RMS   | max  | Speedup |  |  |
| s953      | 0.01          | 0.02 | 0.04  | 0.11 | 12.18   | 0.04  | 0.11 | 13.00   | 0.02  | 0.05          | 0.04  | 0.12 | 12.77   | 0.04  | 0.12 | 12.30   |  |  |
| s1196     | 0.03          | 0.08 | 0.03  | 0.09 | 1.64    | 0.03  | 0.11 | 2.51    | 0.04  | 0.10          | 0.04  | 0.10 | 1.65    | 0.04  | 0.14 | 2.53    |  |  |
| s1238     | 0.03          | 0.11 | 0.03  | 0.13 | 1.81    | 0.03  | 0.10 | 2.59    | 0.04  | 0.16          | 0.04  | 0.17 | 1.73    | 0.04  | 0.14 | 2.57    |  |  |
| s1423     | 0.02          | 0.05 | 0.02  | 0.05 | 62.73   | 0.02  | 0.05 | 74.87   | 0.00  | 0.00          | 0.00  | 0.00 | 69.81   | 0.00  | 0.00 | 85.68   |  |  |
| s1488     | 0.03          | 0.09 | 0.03  | 0.08 | 2.20    | 0.02  | 0.04 | 3.25    | 0.02  | 0.05          | 0.03  | 0.16 | 2.50    | 0.03  | 0.16 | 3.76    |  |  |
| s1494     | 0.01          | 0.02 | 0.04  | 0.07 | 22.91   | 0.04  | 0.07 | 23.65   | 0.03  | 0.04          | 0.03  | 0.04 | 48.73   | 0.03  | 0.04 | 45.69   |  |  |
| s5378     | 0.02          | 0.07 | 0.04  | 0.09 | 6.16    | 0.04  | 0.09 | 9.02    | 0.04  | 0.13          | 0.04  | 0.13 | 8.71    | 0.04  | 0.13 | 9.50    |  |  |
| s9234     | 0.01          | 0.02 | 0.01  | 0.02 | 4.37    | 0.01  | 0.03 | 7.05    | 0.04  | 0.07          | 0.04  | 0.07 | 4.43    | 0.04  | 0.07 | 7.13    |  |  |
| s13207    | 0.02          | 0.07 | 0.02  | 0.07 | 3.14    | 0.01  | 0.04 | 4.91    | 0.04  | 0.14          | 0.04  | 0.14 | 2.92    | 0.02  | 0.06 | 4.50    |  |  |
| s15850    | 0.02          | 0.04 | 0.03  | 0.07 | 28.35   | 0.03  | 0.07 | 95.94   | 0.02  | 0.04          | 0.02  | 0.04 | 28.62   | 0.02  | 0.03 | 42.88   |  |  |
| s35932    | 0.00          | 0.04 | 0.02  | 0.06 | 14.07   | 0.03  | 0.06 | 21.46   | 0.00  | 0.04          | 0.00  | 0.02 | 1.66    | 0.00  | 0.03 | 2.26    |  |  |
| s38417    | 0.02          | 0.06 | 0.02  | 0.06 | 6.73    | 0.02  | 0.05 | 10.05   | 0.02  | 0.06          | 0.02  | 0.06 | 8.19    | 0.02  | 0.06 | 11.86   |  |  |
| s38584    | 0.07          | 0.18 | 0.07  | 0.18 | 19.12   | 0.04  | 0.11 | 22.58   | 0.10  | 0.26          | 0.10  | 0.26 | 32.20   | 0.08  | 0.19 | 39.37   |  |  |

The error is with respect to  $\Phi(\alpha_{global})$  computed using Monte Carlo analysis. In order to check that the threshold for pruning is applicable to other circuits, we have also used the algorithm for ISCAS89 benchmarks. It is seen that the error obtained using K-center pruning and our algorithms are comparable for both the benchmark circuits and for both quad-trees. Once again, if CMBT is used to find the maximum of each cutset, the error in case 3. can be reduced.

It is seen that a considerable speedup, sometimes about two orders of magnitude, can be obtained when  $\Phi(\alpha)$  is computed from  $\Phi(\beta_{cutset})$ . The lowest speedup is about 35%, which occurs when most cutsets have a large number of nodes that are near critical.

## V. CONCLUSION

In this paper, we have looked at two measures for criticality that can be used instead of the conventional measure obtained using the complementary path delay. There is a clear one-to-one map between the first measure and the conventional measure. So the two can be used interchangeably with approximately a 30% speedup in computation time. The second is more empirical and could require some characterization. But it is very easy to compute. Dominant high

criticality nodes and majority of the low criticality nodes can easily be identified. Therefore it can be used quite effectively for pruning and considerable speedup can be obtained especially for some of the larger benchmark circuits.

Typically statistical distance measures use only the mean and the standard deviation. There will always be some ambiguity since the correlation is not used. Ideally therefore, the measure that we seek should use the correlation and should be easy to compute. However as the number of sources of variation increase, the time required for computing the correlation itself becomes significant even if sparse arrays are used. Of course, most of the computations are easily parallelizable and it should be possible to get a good speedup by running it on multiple cores.

## APPENDIX A

For  $z = MAX(x, y)$ , we show that  $\mu_z > max(\mu_x, \mu_y)$ . Without loss of generality, assume that  $\mu_x > \mu_y$ , which means  $\alpha$  is positive. Using (4),

$$\mu_z - \mu_x = a(-\alpha\Phi(-\alpha) + \phi(\alpha)) = aF(\alpha) \quad (21)$$

It can be easily shown that

$$\frac{dF}{d\alpha} = -\Phi(-\alpha) < 0$$

Therefore  $F(\alpha)$  is a monotonically decreasing function of  $\alpha$ . Since  $\alpha$  is positive, the upper bound for  $F(\alpha)$  is  $\phi(0) > 0$ . From (4), as  $\alpha \rightarrow \infty$ ,  $\mu_z \rightarrow \mu_x$ , which means that the lower bound for  $F(\alpha)$  is zero. Since  $a > 0$ , this implies  $\mu_z - \mu_x > 0$ . Therefore,  $\mu_z > \max(\mu_x, \mu_y)$ .

#### APPENDIX B

Clark's formulation satisfies both shifting and scaling properties of  $MAX$  operation. So without loss of generality assuming  $\mu_y = 0$  and  $\mu_x \geq \mu_y$ ,

$$\begin{aligned} \alpha &= \frac{\mu_x}{a} \\ \mu_z &= \mu_x \Phi(\alpha) + a\phi(\alpha) \\ \sigma_z^2 &= (\mu_x^2 + \sigma_x^2)\Phi(\alpha) + \sigma_y^2\Phi(-\alpha) + \mu_x a\phi(\alpha) - \mu_z^2 \\ \sigma_z^2 &= \sigma_x^2\Phi(\alpha) + \sigma_y^2\Phi(-\alpha) + \mu_x(\mu_x\Phi(\alpha) + a\phi(\alpha)) - \mu_z^2 \\ &= \sigma_x^2\Phi(\alpha) + \sigma_y^2\Phi(-\alpha) + \mu_z(\mu_x - \mu_z) \end{aligned}$$

If  $\sigma_x \geq \sigma_y$ ,

$$\begin{aligned} \sigma_z^2 - \sigma_x^2 &= (\sigma_y^2 - \sigma_x^2)\Phi(-\alpha) + \mu_z(\mu_x - \mu_z) \leq 0 \\ &\text{as } \mu_x \leq \mu_z \text{ (from Appendix A) \& } \sigma_y \leq \sigma_x \end{aligned}$$

Similarly if  $\sigma_y \geq \sigma_x$ ,

$$\sigma_z^2 - \sigma_y^2 = (\sigma_x^2 - \sigma_y^2)\Phi(\alpha) + \mu_z(\mu_x - \mu_z) \leq 0$$

Hence  $\sigma_z$  is always less than the maximum of the two  $\sigma$ 's.

#### APPENDIX C

If  $w = x + y = \mathcal{N}(\mu_w, \sigma_w)$ ,  $\frac{\sigma_w}{\mu_w} \leq \max\left(\frac{\sigma_x}{\mu_x}, \frac{\sigma_y}{\mu_y}\right)$ .

*Proof:*  $\mu_w$  and  $\sigma_w$  are given by

$$\begin{aligned} \mu_w &= \mu_x + \mu_y \\ \sigma_w &= \sqrt{\sigma_x^2 + \sigma_y^2 + 2\rho_{xy}\sigma_x\sigma_y} \end{aligned}$$

The maximum value of  $\sigma_w = (\sigma_x + \sigma_y)$ . If  $\frac{\sigma_x}{\mu_x} \geq \frac{\sigma_y}{\mu_y}$ ,

$$\begin{aligned} \implies \sigma_x\mu_y &\geq \sigma_y\mu_x \\ \sigma_x\mu_y + \sigma_x\mu_x &\geq \sigma_y\mu_x + \sigma_x\mu_x \\ \implies \frac{\sigma_x}{\mu_x} &\geq \frac{\sigma_x + \sigma_y}{\mu_x + \mu_y} \end{aligned}$$

The same argument is valid if  $\frac{\sigma_y}{\mu_y} \geq \frac{\sigma_x}{\mu_x}$ . Which implies,

$$\frac{\sigma_w}{\mu_w} \leq \frac{\sigma_x + \sigma_y}{\mu_x + \mu_y} \leq \max\left(\frac{\sigma_x}{\mu_x}, \frac{\sigma_y}{\mu_y}\right)$$

#### APPENDIX D

If  $z = MAX(x, y) = \mathcal{N}(\mu_z, \sigma_z)$ ,  $\frac{\sigma_z}{\mu_z} \leq \max\left(\frac{\sigma_x}{\mu_x}, \frac{\sigma_y}{\mu_y}\right)$ .

*Proof:* Let  $z = MAX(x, y) = \mathcal{N}(\mu_z, \sigma_z)$ . From Appendices A and B,  $\mu_z \geq \mu_x$ ,  $\mu_z \geq \mu_y$  and  $\sigma_z \leq \max(\sigma_x, \sigma_y)$ . Let  $\mu_x > \mu_y$ , which results in three possible scenarios:

1.)  $\sigma_x \geq \sigma_y$  and  $\frac{\sigma_x}{\mu_x} \geq \frac{\sigma_y}{\mu_y}$  :

$$\implies \frac{\sigma_x}{\mu_x} \geq \frac{\sigma_x}{\mu_z} \geq \frac{\sigma_z}{\mu_z}$$

2.)  $\sigma_x \geq \sigma_y$  and  $\frac{\sigma_x}{\mu_x} \leq \frac{\sigma_y}{\mu_y}$  :

$$\implies \frac{\sigma_y}{\mu_y} \geq \frac{\sigma_x}{\mu_x} \geq \frac{\sigma_x}{\mu_z} \geq \frac{\sigma_z}{\mu_z}$$

3.)  $\sigma_x \leq \sigma_y \implies \frac{\sigma_x}{\mu_x} \leq \frac{\sigma_y}{\mu_y}$  :

$$\implies \frac{\sigma_y}{\mu_y} \geq \frac{\sigma_y}{\mu_z} \geq \frac{\sigma_z}{\mu_z}$$

From the three scenarios it is clear that,

$$\frac{\sigma_z}{\mu_z} \leq \max\left(\frac{\sigma_x}{\mu_x}, \frac{\sigma_y}{\mu_y}\right)$$

#### REFERENCES

- [1] C. Visweswariah, K. Ravindran, K. Kalafala, S. Walker, and S. Narayan, "First-order incremental block-based statistical timing analysis," in *Proc. DAC*, 2004, pp. 331–336.
- [2] J. Xiong, V. Zolotov, N. Venkateswaran, and C. Visweswariah, "Criticality computation in parameterized statistical timing," in *Proc. DAC*, 2006, pp. 63–68.
- [3] H. Mogal, H. Qian, S. Sapatnekar, and K. Bazargan, "Fast and accurate statistical criticality computation under process variations," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 28, no. 3, pp. 350–363, March 2009.
- [4] X. Li, J. Le, M. Celik, and L. Pileggi, "Defining statistical timing sensitivity for logic circuits with large-scale process and environmental variations," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 27, no. 6, pp. 1041–1054, June 2008.
- [5] A. Agarwal, K. Chopra, and D. Blaauw, "Statistical timing based optimization using gate sizing," in *Proc. DATE*, 2005, pp. 400–405.
- [6] J. Xiong, V. Zolotov, and C. Visweswariah, "Incremental criticality and yield gradients," in *Proc. DATE*, 2008, pp. 1130–1135.
- [7] J. Chung, J. Xiong, V. Zolotov, and J. A. Abraham, "Path criticality computation in parameterized statistical timing analysis using a novel operator," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 31, no. 4, pp. 497–508, 2012.
- [8] A. Agarwal, D. Blaauw, V. Zolotov, S. Sundareswaran, M. Zhao, K. Gala, and R. Panda, "Statistical delay computation considering spatial correlations," in *Proc. ASP-DAC*, 2003, pp. 271–276.
- [9] H. Chang and S. Sapatnekar, "Statistical timing analysis under spatial correlations," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 24, no. 9, pp. 1467–1482, sept. 2005.
- [10] C. E. Clark, "The greatest of a finite set of random variables," *OPERATIONS RESEARCH*, vol. 9, no. 2, pp. 145–162, 1961.
- [11] D. Sinha, H. Zhou, and N. Shenoy, "Advances in computation of the maximum of a set of gaussian random variables," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 26, no. 8, pp. 1522–1533, Aug. 2007.
- [12] D. Blaauw, K. Chopra, A. Srivastava, and L. Scheffer, "Statistical timing analysis: From basic principles to state of the art," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 27, no. 4, pp. 589–607, April 2008.
- [13] F. Wane, Y. Xie, and H. Ju, "A novel criticality computation method in statistical timing analysis," in *Proc. DATE*, 2007, pp. 1–6.
- [14] Y. Zhan, A. Strojwas, M. Sharma, and D. Newmark, "Statistical critical path analysis considering correlations," in *Proc. ICCAD*, 2005, pp. 699–704.
- [15] J. Chung, J. Xiong, V. Zolotov, and J. Abraham, "Path criticality computation in parameterized statistical timing analysis," in *Proc. ASP-DAC*, 2011, pp. 249–254.