Efficient Statistical Timing Analysis Through Error Budgeting

Vishal Khandelwal, Azadeh Davoodi and Ankur Srivastava
Department of Electrical and Computer Engineering, University of Maryland - College Park.
{vishalk,azade,ankurs}@glue.umd.edu

ABSTRACT

In this paper we propose a novel technique for optimizing the runtime in statistical timing analysis. Given a global acceptable error budget at the primary output which signifies the difference in the area of the accurate and approximate timing CDFs, we propose a novel formulation of budgeting this global error across all nodes in the circuit. This node error budget is used to simplify the computation of arrival time CDFs at each node using approximations. This simplification reduces the runtime of statistical timing analysis. We investigate two ways of exploiting this node error budget, firstly through piecewise linear approximation ([4]) and secondly though hierarchical quadratic approximation. Experimental results on ISCAS/MCNC benchmarks show that our approach is at most 3 times faster than accurate statistical timing analysis and had a very small error. We also found quadratic piecewise approximation to be more accurate than linear approximation but at lesser gains in runtime.

1 Introduction

Growing importance of fabrication variability and estimation uncertainty has lead to increased significance of statistical timing analysis. Several researchers have investigated this issue in detail [3, 2, 1, 8, 11, 12, 13, 6, 4]. Statistical timing analysis problem essentially takes a DAG G = (V,E) as input with each node delay and arrival time represented as a distribution. It calculates the distribution of the arrival time at the primary outputs (POs) of the DAG. One of the most important issue in statistical timing analysis is the runtime. The latest work by Devgan et. al [4] proposes an approach for fast statistical timing analysis in which after the node arrival time CDF is evaluated, the CDF is approximated by a piecewise linear approach. This simplification results in massive gains in runtime.

Our work builds upon this approach for statistical timing analysis. The key problem in the approach presented in [4] is that whenever a signal is approximated by piecewise linearization, this linearization is performed using an arbitrary and predecided number of lines. Having too few lines could result in large amount of error and too many lines could result in large execution runtime. Hence an adaptive way of determining the degree of approximation for each signal is needed which can effectively perform a tradeoff between gain/loss in runtime with increase/decrease in error. In order to achieve this tradeoff we investigate the way error gets propagated in statistical timing analysis. We propose a closed form expression for this error propagation. Using this expression, we propose the philosophy of error budgeting. The error budgets at each node are used to approximate the node delay PDFs and arrival time CDFs. We investigate two kinds of approximation strategies: linear (traditional) and hierarchical quadratic. This entire statistical timing analysis framework is put together in the SIS framework. Experimental results show that our budgeting approach comes very close to accurate statistical timing estimation (without any approximation) but can be at most 3 times faster. Comparatively, the traditional approach [4] had a large error in the output arrival time CDF. We also found the quadratic approximation to be much more accurate than linear approximation but with lesser gains in runtime.

The rest of the paper is organized as follows. Section 2 describes the motivation and the statistical timing framework of this work. Section 3 contains the proposed error budgeting formulation. Section 4 the proposed the linear and quadratic approximation strategy. The results are presented in section 5 and conclusion in section 6.

2 Motivation and STA Framework

In this paper, we propose a novel approach for speeding up statistical timing analysis by effectively controlling the amount of error injected for gains in runtime. Given the distribution of the arrival time at the primary inputs and the distribution of the gate delays, the problem is to evaluate the distribution of arrival time at the intermediate nodes as well as the output nodes in the circuit. Similar to static timing analysis, statistical timing analysis traverses the circuit topologically from the primary inputs to the primary outputs generating the arrival time distribution at the out of each intermediate node.

The SUM and the MAX operation in the statistical timing framework need to be computed on the distributions of arrival times and gate delays. In [4], the authors propose to model the arrival times as cumulative density functions (CDFs) and the gate delays as probability density functions (PDFs) as shown in figure 1(a). t_1 and t_2 denote the range of the distributions in both the cases as shown in the figure.

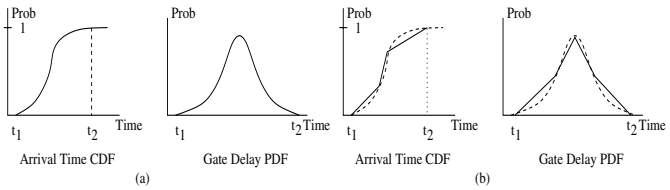


Figure 1: Distributions and their Linear Approximations

For computational efficiency of the SUM and MAX operations of statistical timing, these CDFs and PDFs are approximated using techniques of piecewise linear and quadratic approximations. The details of these modelings are given later in section 4. In figure 1(b), the CDF and PDF are shown under the piecewise linear approximation scheme. We will now discuss the SUM and MAX operation under these CDFs and PDFs. We assume that the arrival times and gate delays are independent of each other. The issue of statistical dependence due to re-convergent fanouts needs to be resolved [4], [1], [3]. In [4], the authors propose an efficient heuristic technique based on common mode removal approach which we have implemented in this work.

In [4], the authors show that the CDF of the arrival time $C_o^x(t)$ at the output due to input pin x is given by the convolution of the input arrival time CDF $C_x(t)$ with the PDF of the pin-to-pin gate delay $P_o^x(t)$ as given by equation 1. This follows from the fact that the probability distribution of the sum of two independent random variables is the convolution of their probability distributions.

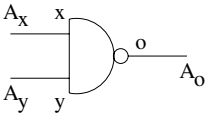


Figure 2: Gate with x and y input pins and output o

$$C_o^x(t) = \int_0^t (C_x(t-\tau) * P_o^x(\tau) d\tau) \quad (1)$$

Similarly, the CDF $C_o(t)$ after the MAX operation on the arrival time CDFs $C_o^x(t)$ and $C_o^y(t)$ at the output pin o (refer to figure 2) can be computed from equation 3. The CDF of the maximum of two independent random variables is the product of their CDFs.

$$A_o(t) = MAX(A_o^x(t), A_o^y(t)) \quad (2)$$

$$C_o(t) = C_o^x(t) * C_o^y(t) \quad (3)$$

Hence statistical timing operations SUM and MAX are now performed by doing a convolutions followed by a multiplication. Hence the arrival time distribution at the output of the gate, given the input arrival time distributions and the gate delay distribution can be given by equation 4.

$$C_o(t) = (C_x(t) \otimes P_o^x(t)) * (C_y(t) \otimes P_o^y(t)) \quad (4)$$

Now that we have the formulations for the MAX and SUM operation for statistical timing using CDFs and PDFs, we can run statistical timing analysis similar to conventional static timing. Equation 4 can used to evaluate the output CDFs at each gate in the circuit. In order to speed up statistical timing evaluation, the approach of [4] linearizes the arrival time CDF into a prespecified number of lines. It also approximates the arbitrary node delay PDF into stepwise function. The authors then formulated a closed form expression for evaluating equations 1 and 3 when the arrival time CDFs were represented using a piecewise linear approximation and the node delay PDF was represented using a stepwise approximation. This results in huge speed ups in runtime when compared with a traditional point-wise convolution based approach. The overall runtime of timing analysis depends upon the total number of lines used

0-7803-8702-3/04/$20.00 ©2004 IEEE.

to represent the arrival time CDF and the total number of steps used to represent the node delay PDF.

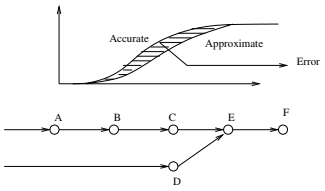


Figure 3: Error Budgeting

In this paper we propose novel ways of controlling this tradeoff between the overall error and runtime. Specifically, we have investigated two issues in this direction.

1. Given an error budget that the user specifies, identify the degree of approximation needed for each individual node arrival time CDFs and node delay PDFs
2. Investigating better approximation strategies like quadratic (instead of linear) for improving error and same runtime

Figure 3 illustrates the basic philosophy behind our approach. Given, node delay distributions in a DAG, the approach in [4] topologically computes the arrival time CDFs at each node. Whenever a new CDF is computed it is simplified by representing it as a piecewise linear approximation. This simplification adds an error into the statistical timing estimation which is controllable by the number of lines used to approximate the CDFs. Finally, the CDF at the output has some error when compared with the accurate arrival time CDF. In this work, we define this error as follows

$$ERROR = \int_{t_{min}}^{t_{max}} |C_{accurate} - C_{estimate}|dt \quad (5)$$

Essentially, this is the total area in the entire range of interest where the actual signal is different from the approximate signal. Let us suppose that we are provided a total error budget E that the user is willing to tolerate at the primary output. Given this error budget, we would like to assign it to all nodes in such a way that maximum gains in runtime occur. Traditionally, this global error budget is essentially spread uniformly. This is not a very effective strategy of distributing the global error since the DAG may have unbalanced paths. Consider the example DAG shown in figure 3. Approximating all node CDFs with the same number of points would not be the best idea since node D is not critical. Hence the global arrival time CDF at node F has low sensitivity to the amount of error in the arrival time CDF at node D. Hence runtime speed-ups could be achieved by adding more error at D by approximating it in lesser number of lines. We call this concept *Error Budgeting*, since through this approach we strive to control the amount of error in the final output CDF for gains in runtime. We also investigate better approximation techniques like quadratic approximation for lesser error. The budgeting and approximation schemes are integrated into one statistical timing system.

3 Error Budgeting

In this section we will delve into the details of our budgeting formulation that distributes the global error budget at the PO to each node which can then be utilized for speeding up statistical timing analysis. The error budget at the primary outputs is defined in equation 5. In order to distribute this global error budget we need to investigate the way error in arrival time CDFs and node delay PDFs interact when subjected to SUM and MAX operations.

3.1 Error in SUM Operation

Figure 4 illustrates a situation in which the SUM operation is performed on two signals, one of which is represented as a CDF and other as a PDF (just like equation 1). The figure illustrates two representations for the input CDF and PDFs, one of which is accurate and one of which is an approximation. In this section we will discuss the error in the output CDF after the SUM operation as a function of the errors in the input CDF and PDF. The accurate output CDF is given by

$$C_{out}^{accurate}(t) = \int_0^t C_{in}^{accurate}(t-\tau)P_{node}^{accurate}(\tau)d\tau \quad (6)$$

The approximate output CDF is given by

$$C_{out}^{aprrox}(t) = \int_0^t C_{in}^{approx}(t-\tau)P_{node}^{approx}(\tau)d\tau \quad (7)$$

Since the SUM operation is essentially a convolution operation, the range of the output CDF is defined as follows. If the input CDF starts at t1 and ends at t2 (after t2 the CDF=1) and the input PDF starts

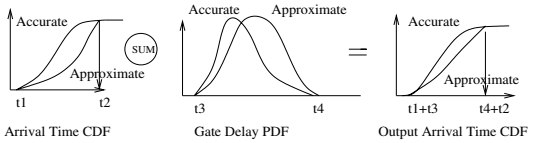


Figure 4: Error in SUM

Figure 5: Error in MAX

at t3 and ends at t4, then the output CDF starts at t1+t3 and ends at t2+t4. Note that here the assumption is that both accurate and approximate curves start and end at the same delay value. As it would be clear in the next section, the way piecewise linear approximation or quadratic approximation is performed, this range does not change. Hence t1,t2,t3,t4 are the same for original and approximate curve. The error in the output CDFs is given by

$$Err_{out} = \int_{t_{min}}^{t_{max}} |C_{out}^{accurate} - C_{out}^{approx}|dt \quad (8)$$

The error term can be re-written as follows

$$Err_{out} = \int_{t_{min}}^{t_{max}} |\int_0^t C_{in}^{approx}(t-\tau)P_{node}^{approx}(\tau)d\tau - \int_0^t C_{in}^{accurate}(t-\tau)P_{node}^{accurate}(\tau)d\tau|dt \quad (9)$$

The range of the integral t_{min}, t_{max} is simply t1+t3 and t2+t4 respectively.

$$Err_{out} = \int_{t_{min}}^{t_{max}} |\int_0^t (C_{in}^{approx}(t-\tau)P_{node}^{approx}(\tau) - C_{in}^{accurate}(t-\tau)P_{node}^{accurate}(\tau))d\tau|dt \quad (10)$$

Let us suppose that $C^{approx} = C^{accurate} + \delta C$ and $P^{approx} = P^{accurate} + \delta P$. Plugging this relation in equation 10 gives us the following result.

$$\int_{t_{min}}^{t_{max}} |\int_0^t (C_{in}^{accurate}(t-\tau)\delta P(\tau) + P^{accurate}(\tau)\delta C(t-\tau) + \delta P(\tau)\delta C(t-\tau))d\tau|dt \quad (11)$$

Ignoring the second order term $\delta P(\tau)\delta C(t-\tau)$ and using the relation $|a+b| \leq |a| + |b|$, the above equation could be rewritten as

$$\int_{t_{min}}^{t_{max}} (\int_0^t |C_{in}^{accurate}(t-\tau)\delta P(\tau)|d\tau + \int_0^t |P^{accurate}(\tau)\delta C(t-\tau)|d\tau)dt \quad (12)$$

$$\int_{t_{min}}^{t_{max}} (\int_0^t |C_{in}^{accurate}(t-\tau)\delta P(\tau)|d\tau + \int_0^t |P^{accurate}(\tau)||\delta C(t-\tau)|d\tau)dt \quad (13)$$

Let the error in the input CDF be $E1 = \int_{t1}^{t2} |(C_{node}^{accurate} - C_{node}^{approx})|dt$ and error in input PDF $= E2 = \int_{t3}^{t4} |(P^{accurate} - P^{approx})|dt$. Since $0 \leq C_{in}^{accurate}(t) \leq 1$ and $E1 \geq |\delta C(t-\tau)|$ it can clearly be seen that equation 13 is always $\leq$ the following

$$Err_{out} \leq \int_{t_{min}}^{t_{max}} (\int_0^t |\delta P(\tau)|d\tau + E1\int_0^t |P^{accurate}(\tau)|d\tau)dt \quad (14)$$

$$Err_{out} \leq \int_{t_{min}}^{t_{max}} (E1+E2)dt = (E1+E2)(t_{max} - t_{min}) \quad (15)$$

Equation 15 gives an upper bound on the output CDF error based in the input errors. The range t_{max}, t_{min} is simply the range on which the output arrival time signal is defined. Therefore the output error is a linear combination of input errors.

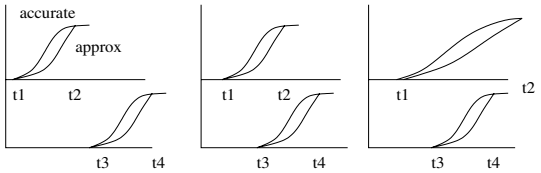


Figure 6: Error Bound in MAX

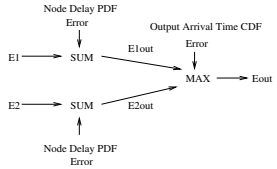


Figure 7: Error Injection in a Gate

3.2 Error in MAX Operation

Figure 5 illustrates a similar situation for the MAX operation. The input CDFs have the range (t1,t2) and (t3,t4) both for the accurate and approximate cases. The output CDF which is a multiplication of the input CDFs has the range (t_{min}, t_{max}) = (max(t1,t3), max(t2,t4)). Once again the error in the output CDF is given as follows

$$Err_{out} = \int_{t_{min}}^{t_{max}} |(C_{out}^{approx}(t) - C_{out}^{accurate}(t)|dt \quad (16)$$

Let $C_{in1}^{approx}, C_{in2}^{approx}$ and $C_{in1}^{accurate}, C_{in2}^{accurate}$ denote the accurate and approximate CDFs for the input signals. Let $C_{in1}^{approx} = C_{in1}^{accurate} + \delta C_{in1}$ and $C_{in2}^{approx} = C_{in2}^{accurate} + \delta C_{in2}$. Using these relations and simplifying, we can write equation 16 as follows

$$Err_{out} = \int_{t_{min}}^{t_{max}} |C_{in1}^{accurate}\delta C_{in2} + C_{in2}^{accurate}\delta C_{in1}|dt \quad (17)$$

Let E1 = $\int_{t1}^{t2} |C_{in1}^{approx} - C_{in1}^{accurate}|dt$ and E2 defined similarly for the second input. Using $|a+b| \leq |a| + |b|$, it can be shown that following must hold.

$$Err_{out} \leq \int_{t_{min}}^{t_{max}} |C_{in1}^{accurate}\delta C_{in2}|dt + \int_{t_{min}}^{t_{max}} |C_{in2}^{accurate}\delta C_{in1}|dt \quad (18)$$

$$Err_{out} \leq E1 + E2 \quad (19)$$

Although equation 19 is an upper bound on the error, this bound is not good enough since it does not capture the criticality of the inputs. As discussed in the previous section, the error in a non-critical fanin would not affect the output error too much. Unfortunately, equation 19 does not capture this philosophy. Hence we refine this bound by making some approximations on the input CDFs.

Figure 6 illustrates three possible overlaps between the two input CDFs. In the first case, the CDFs have no overlap whatsoever. Here $t1 \leq t2 \leq t3 \leq t4$. In such a situation,$C_{out}$ will be zero until t_{min} = max(t1,t3) = t3 and will become 1 at t_{max} = max(t2,t4). Essentially the second signal is always more critical than the first one. Also, since, we have assumed the range of approximate and accurate curves to be the same, the error is zero outside it. If we focus on equation 18, the second term must be zero since over the range t_{max}, t_{min} = (t4,t3), the error in the first signal is zero. Hence the entire output error is contributed by E2. Analytically, this means that since signal-2 is critical, the error contributed by signal-1 does not affect the output signal.

In the second case in figure 6, the two input signals overlap such that $t1 \leq t3 \leq t2 \leq t4$. Here t_{min}, t_{max} = (max(t1,t3)=t3, max(t2,t4)=t4). In such a case, we assume that the two CDFs are lines with the following slopes

$$S1 = 1/(t2 - t1) \quad (20)$$

$$S2 = 1/(t4 - t3) \quad (21)$$

This approximation is needed in order to evaluate a closed form expression for the output error in terms of the input errors. Hence the input CDF $C_{in1}(t)$ = S1(t - t1) $\forall$ t $t1 \leq t \leq t2$ and input CDF $C_{in2}(t)$ = S2(t - t3) $\forall$ t $t3 \leq t \leq t4$. Let us also approximate the error between the accurate and approximate CDFs to be uniformly distributed. This is illustrated in the following equations

$$\delta C_{in1}(t) = E1/(t2 - t1) \quad t1 \leq t \leq t2 \quad (22)$$

$$= 0 \quad otherwise \quad (23)$$

$$\delta C_{in2}(t) = E2/(t4 - t3) \quad t3 \leq t \leq t4 \quad (24)$$

$$= 0 \quad otherwise \quad (25)$$

Once again we would like to re-iterate the assumption that the range of accurate and approximate CDFs are the same. This issue will be further explained later. Equation 18 has two terms, each corresponding to the error contributed by the respective inputs. For each term, the entire range of integration is split into two parts: from t_{min} = $t3$ to $t2$ and from $t2$ to t_{max}. The first term in equation 18, $\int_{t_{min}}^{t_{max}} |C_{in1}^{accurate}\delta C_{in2}|dt$ therefore gets split into two integrals. Using the simplifying assumptions given by equations 20 and 24, this term can be written as follows

$$\int_{t_{min}}^{t_{max}} |C_{in1}^{accurate}\delta C_{in2}|dt = K2E2 \quad (26)$$

Here $K2 = (S1/t4-t3)((t2^2-t3^2)/2-t1(t2-t3))+(t4-t2)/(t4-t3)$. Similarly the second term in equation 18 can be simplified as follows

$$\int_{t_{min}}^{t_{max}} |C_{in2}^{accurate}\delta C_{in1}|dt = K1E1 \quad (27)$$

Here $K1 = (S2/(t2-t1))((t2^2-t3^2)/2 - t3(t2-t3))$. Therefore the total error is given by $K1E1 + K2E2$.

Now let us consider the final case in figure 6. In this case one input signal completely engulfs the other. In this case t_{min}, t_{max} is given by (max(t1,t3)= t3, max(t2,t4) = t2). Under similar simplifying assumptions form equations 21 and 22, we can re-express equation 18 as $K1E1 + K2E2$ with

$K1 = (S2/(t2-t1))((t4^2-t3^2)/2 - t3(t4-t3)) + (t2-t4)/(t2-t1)$
$K2 = (S1/t4-t3)((t4^2-t3^2)/2 - t1(t4-t3))$

It can be seen that in all cases the error is bounded by $K1E1+K2E2$ where K1 and K2 can be calculated using the proposed expressions. This gives us a compact and effective way of estimating the output error given the input errors and the ranges in which the input CDFs exist. It should be noted that the upper bound property may not hold anymore.

The assumptions made on the nature of the CDFs considering them to be linear ramps as given by equations 20 and 21 can be relaxed for better accuracy. We could also consider them to be gaussian (or any other distribution) and evaluate closed form expressions for $K1$ and $K2$ (under the assumption that the error is uniformly distributed).

Having delved into the details of how the error propagates in the SUM and MAX function, now we will describe the way error budgeting is performed for each node.

3.3 Error Budgeting for Runtime Optimization

Given a user defined error budget at the primary outputs of a DAG, we would like to assign error budgets to individual nodes in the DAG such that overall error budget constraint is satisfied and maximum gains in runtime could be achieved. Given the input DAG, let us add a sink node and add directed edges from all POs to this sink node. We also assume this sink node has zero delay.

Figure 7 illustrates the way error is injected into a gate. There are two inputs with errors E1 and E2. First these input signals are SUMmed with the corresponding input pin to output delay. At this point there is an error injected that corresponds to the error corresponding to linear approximation or quadratic approximation of the node delay PDFs as shown in figure 7. These CDFs are then MAXed together to get the node output arrival time CDF. Another error is added here which corresponds to the linear approximation or quadratic approximation of the output CDF as shown in figure 7. Hence there are two kinds of errors associated with a gate: first is the one that gets injected due to simplification of the node delay PDFs and other due to simplification of the node output CDF. Hence in the entire DAG, each node has two variables corresponding to node delay PDF simplification (assuming all gates are 2 inputs) and one variable for node output CDF simplification. Therefore there are 3n error variables, where n is the number of nodes. Errors need to be assigned to these variables such that the overall sum of the errors for all variables is maximized and the error budget at the sink node is satisfied. Formally this can be written as follows.

$$Maximize \sum_{\forall nodes:i} (e_{input-j:i}^{pdf} + e_{input-k:i}^{pdf} + e_{i:out}^{cdf}) \quad (28)$$

$$e_{sink:out} \leq ERR - BUDGET \quad (29)$$

$$e_{input-j:i}^{dummy} = K1_{ij}^{sum} e_{j:out} + K2_{ij}^{sum} e_{input-j:i}^{pdf} \quad \forall inputs - j : i \quad \forall i \quad (30)$$

$$e_{out:i}^{dummy} = K1_i^{max} e_{input-j:i}^{dummy} + K2_i^{max} e_{input-k:i}^{dummy} \quad \forall i \quad (31)$$

$$e_{out:i} = e_{out:i}^{dummy} + e_{out:i}^{cdf} \quad \forall i \quad (32)$$

Equation 30 illustrates that when the input CDF is SUMmed with the node PDF, then the output error is a linear combination of the input error and the error injected by approximating the node delay PDF. The values of the linear constants could be calculated as described in the previous subsections. Equation 31 illustrates that the output CDF error given the error injected by the MAX operation on two input signals. This error is a linear combination of these two input errors. The output CDF at node i is also approximated thereby introducing another error into the formulation as shown in equation 32. There is a global error

budget at the sink node. The objective is to maximize the total error budget since this would be directly proportional to the overall runtime improvements. There is still the issue of assigning the constants in the above equations (essentially K1, K2 etc.). The last subsection derived analytical formulae for these constants that were dependent on the range of existence of each of the signals. These ranges can be easily derived for all signals in the DAG as follows. First replace all node delays by their minimum possible values and perform as static timing analysis (this gives the lower limit on arrival time). Then replace all node delays by their maximum possible values and perform static timing analysis (this gives the upper limit on arrival time). The range of arrival times for all signals essentially gives the range which is needed by the analytical formulae to compute the K1 and K2 terms. This completes the description of the budgeting formulation.

4 Linear and Quadratic Approximation Schemes

Our error budgeting scheme discussed in section 3 allocates an error to each approximation step. If we can ensure that the approximation error introduced at each step is within the error budget, we can control the total error in the CDF of the output arrival time.

4.1 Piecewise Linear Approximation

We can approximate the PDF of gate delay and the arrival time CDF into piecewise linear PDF and CDF respectively [4]. We are given an error budget for each approximation step from the error budgeting technique explained in section 3. The piecewise linearization could be iteratively refined until the overall error is less than the budget. The piecewise linear CDF and PDF can then be decomposed into a sum of ramps as shown in figure 8(a) and (b) respectively. Hence, if an approximation step has a large error budget, we can approximate it with very few lines are get considerable runtime savings.

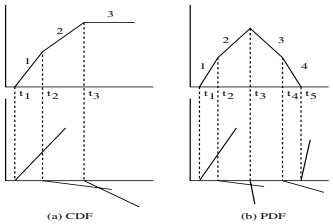


Figure 8: Decomposing CDF and PDF into sum of ramps

The SUM operation as defined before would now be applied to the piece-wise linear CDF (with n ramps) and piece-wise linear PDF (with m ramps) and result in mn convolutions. We can then add up these convolution results to get the intermediate CDF after the SUM operation. However, we will retain them in this decomposed form for the MAX operation. The convolution between two ramps with slopes s_1 and s_2, starting at t_1 and t_2 can be calculated as shown in figure 9(a). The convolution result C_o has a closed form expression given by

$$C_o = s_1 s_2 (1/6t^3 - (t_1/2 + t_2/2)t^2 + (1/2{t_1}^2 + 1/2{t_2}^2 + t_1 t_2)t - (1/6{t_1}^3 + 1/6{t_2}^3 + 1/2t_1{t_2}^2 + 1/2t_2{t_1}^2)) \quad (33)$$

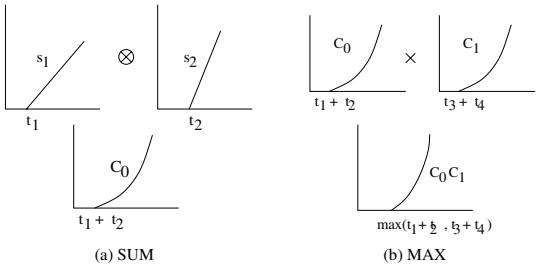


Figure 9: SUM and MAX

Hence, after the SUM operation we have each intermediate CDF represented as sum of mn cubic polynomials. The CDF of the arrival time at the output of the gate is given by the MAX operation on the CDFs obtained after the SUM operation on different input pins of the gate. The closed form expression for the resulting CDF is just the product of the CDFs from the SUM operation. Unlike the approach presented in [4], we do not linearize the CDFs after the sum operation because this step would inject unnecessary error into the CDFs. We can compute the MAX operation on the two CDFs C_0 (say with $m_0 n_0$ cubic polynomials after SUM) and C_1 (say with $m_1 n_1$ cubic polynomials after SUM) as shown in figure 9(b) by taking the product of every pair of cubic polynomials that were obtained for both the CDFs after the SUM operation as given by equation 33. The MAX operation would therefore generate $(m_0 n_0 m_1 n_1)$ polynomials of degree six which can be summed together to get the CDF of the arrival time at the output of the gate as given by equation 34.

$$C_{out}(t) = \Sigma_{i,j} C_0^i(t) C_1^j(t) \quad (34)$$

In order to propagate this to the next fanout gate, we again perform piecewise linearization of the CDF. The number of lines that this CDF is decomposed into depends on the error budget allocated to output linearization of this gate. We again repeat the iterative decomposition until the error budget is met.

4.2 Hierarchical Quadratic Approximation

The approximation of the CDFs and PDFs can also be done using hierarchical quadratic modeling [9, 10]. This has an advantage over linear approximation since quadratic approximation has lesser error. In this work, we apply the philosophy of hierarchical quadratic modeling in which the approximation is refined hierarchically till the approximation error is within the allocated error budget. We construct a minimal equidistant hierarchical grid structure as shown in figure 10(a) for each hierarchy level i. In this work we limit the maximum number of hierarchical levels to four. Each hierarchical level doubles the number of approximation quadratic polynomials used from the previous level. Between these approximation points the input signal is approximated as a quadratic such that the error in approximation is minimum. If the overall error is more than the assigned budget then another level approximating points is added.

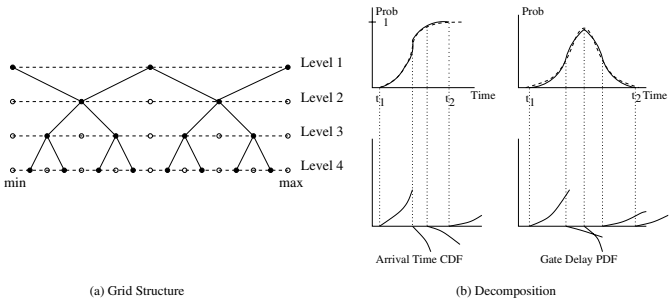


Figure 10: Grid Structure and Quadratic Decomposition

Details about the quadratic approximation techniques on hierarchical basis can be found in [9, 10]. Given a distribution as a CDF or a PDF, we can decompose it into piecewise quadratic function analogous to the piecewise linear case as shown in figure 10(b). The SUM operation would now be applied to piecewise quadratic CDF (with say n quadratics) and piecewise quadratic PDF (with say m quadratics) and result in mn polynomials of degree five. Similar to the linear case, we can derive a closed form expression for this convolution the details of which are omitted for brevity. We can compute the MAX operation on the two CDFs C_0 (say with ${}_0 n_0$ CDFs after SUM) and C_1 (say with $m_1 n_1$ CDFs after SUM) by taking the product of every pair of degree five polynomials that were obtained for both the CDFs after the SUM operation as given by equation 33. The MAX operation would therefore generate $(m_0 n_0 m_1 n_1)$ polynomials of degree ten which can be summed together to get the CDF of the arrival time at the output of the gate as given by equation 34. Although degree ten polynomials may sound too complicated, these are just close form expressions and could be implemented very easily. The output CDF needs to be approximated once again into a piecewise quadratic simplification. This approximation could be done depending on the error budget allocation for this gate.

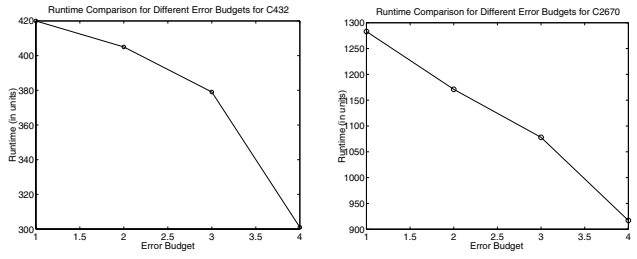
5 Experimental Results

The statistical timing analysis framework with the proposed error budgeting paradigm was implemented in SIS [7]. A topological traversal over the circuit is done in the first step to generate the error budgeting constraints using the LP formulations discussed in section 3. We use CPLEX to solve the error budgeting problem and get an error budget for each step of approximation in statistical timing analysis. We have used the ISCAS/MCNC benchmarks in SIS for our experiments. The arrival

Bench-mark	Accu-rate	Fixed 3 line	Error	Lin. Budget	Error	Quad. Budget	Error
C432	758	92	30.35	420	16.46	601	3.40
C499	1407	176	37.71	679	27.09	1056	9.45
C880	1160	151	13.87	487	8.75	863	1.11
C1908	1793	218	40.15	889	30.97	1249	2.31
C2670	2850	423	10.77	1283	4.36	1966	0.69
C3540	4071	500	11.02	1918	6.49	3146	1.171
C6288	11935	1930	13.47	5985	5.67	7473	2.71
C7552	9249	1467	5.65	3201	1.34	6562	0.48

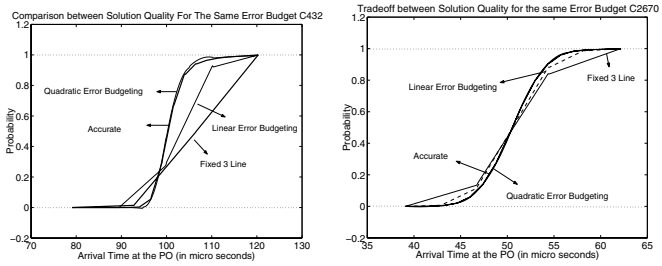
Table 1: Runtime and Error Comparison

time distributions at the primary inputs were taken to be Gaussian (in CDF form) and the gate delay distributions were taken to be Gaussian as well (in PDF form). We have ignored the global corellations in this work and reconvergent fanouts were handled similar to [4]. Since our error budgeting approach uses an adaptive scheme to approximate each distribution depending on its corresponding allocated error budget, we limit the maximum number of segments used to make piecewise linear approximation to 16 lines and the minimum segments to be 3 lines ([4] uses fixed 3 line scheme). Piecewise quadratic approximations have maximum 4 hierarchy levels (or 8 quadratic polynomials). We generate an accurate CDF for the output arrival time for each benchmark to make comparisons in runtime, error budget and the quality of solution between our adaptive approach and the 3 line fixed linearization approach proposed in [4].



(a) Benchmark C432 (b) Benchmark C2670

Figure 11: Runtime Results



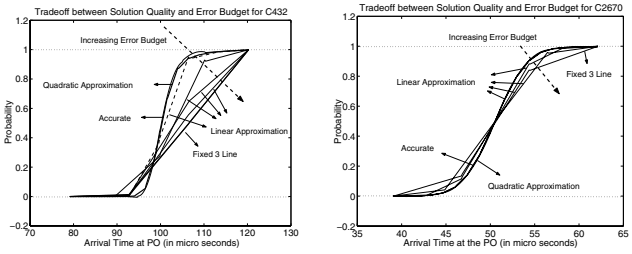
(a) Benchmark C432 (b) Benchmark C2670

Figure 12: STA Results

Table 1 shows the runtime and error comparison between fixed 3-line approximation and our adaptive error budgeting scheme (both linear and quadratic). Columns 2, 3, 5 and 7 give the runtimes for the accurate, fixed 3-line, linear and quadratic cases respectively. The comparisons for error are made with respect to the accurate case using equation 5. Our adaptive linear approximation scheme using error budgeting give solutions which are bounded by the fixed 3-line linearization scheme from [4] and the accurate solution both in terms of runtime and quality of solution. We also note that the runtime of quadratic approximation is lower than that of the accurate distribution while the solution quality obtained from the quadratic scheme is very close to the accurate one. This shows the efficiency of quadratic approximation. However, when compared with linear approximation, it has a higher runtime but better solution quality as well.

Figures 11(b) and 11(a) show the tradeoff between the error budget and runtime. Hence, we can exploit this tradeoff to reduce the runtime of statistical timing analysis.

Figures 12(a) and 12(b) show the CDFs at the primary outputs for two different benchmarks. The error budget assigned to both the linear approximation scheme and the quadratic approximation scheme were the same. We can see from the figures that linear approximation schemes using error budgeting gives better solution quality as compared with fixed 3-line approximation. For the same error budget, quadratic approximation scheme gives us better solution quality but at the cost of a higher runtime when compared with linear approximation scheme. We can clearly see that the solution quality of the quadratic approximation is very close to the accurate distribution for both cases but the runtime are better by 20.7% for C432 and 31.1% for C2670 respectively as shown in table 1. These observations clearly bring out the effective of the quadratic scheme over the linear scheme in terms of the solution quality. The proposed concept of error budgeting is effective in saving runtime while preserving the solution quality.



(a) Benchmark C432 (b) Benchmark C2670

Figure 13: Error Budgeting Tradeoff

The tradeoff between error budget and solution quality obtained in statistical timing analysis is another key observation from the experiments. Now we try to study the effect of changing the assigned error budget during statistical timing analysis. Figures 11(b) and 11(a) show the effect of increasing the error budget on runtime. From figures 13(a) and 13(b), we can see that as the error budget increases, the solution quality from linear approximation decreases. Hence there is a direct tradeoff between the error budget and the corresponding solution quality and runtime.

6 Conclusion and Future Work

In this work we have proposed a novel error budgeting formulation that can be used effectively to inject high error at non-critical nodes giving savings in runtime while maintaining the quality of the solution. The more the error budget that we can tolerate, the more are the runtime savings from our proposed error budgeting scheme. Our solution quality for very low error budgets is very close to the accurate distribution even though we are significantly better in terms of runtime. We have also shown that the quadratic approximation scheme gives us better quality solutions for the same error budget when compared with the linear approximation scheme.

An interesting direction of future work is to investigate a hybrid approach for error budgeting driven statistical timing analysis using hybrid linear and quadratic approximation techniques. Additionally, we also need to consider corellations between gate delay distributions which have been assumed to be independent in this work.

References

[1] A. Agarwal, D. Blaauw and V. Zolotov. "Statistical Timing Analysis for Intra-Die Process Variations with Spatial Correlations". In *Procs of ICCAD*, 2003.

[2] A. Agarwal et al. "Computation and Refinement of Statistical Bounds on Circuit Delay". In *Procs of DAC*, 2003.

[3] A. Agarwal, V. Zolotov and D. Blaauw. "Statistical Timing Analysis Using Bounds and Selective Enumeration". In *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, Vol.22, Sept. 2003.

[4] A. Devgan and C. Kashyap. "Block-based Static Timing Analysis with Uncertainty". In *Procs of ICCAD*, 2003.

[5] C. Visweswariah. "Death, Taxes and Failing Chips". In *Proc. of Design Automation Conference*, June 2003.

[6] C. Visweswariah et al. "First-Order Parameterized Block-Based Statistical Timing Analysis". In *Procs of TAU*, 2004.

[7] E.M. Sentovich, K.J. Singh, L. Lavagno, C. Moon, R. Murgai, A. Saldanha, H. Savoj, P.R. Stephan, R.K. Brayton, A.L. Sangiovanni-Vincentelli. *SIS: A System for Sequential Circuit Synthesis*. Memorandum No. UCB/ERL M92/41, Department of EECS. UC Berkeley, May 1992.

[8] H. Chang and S. Sapatnekar. "Statistical Timing Analysis Considering Spatial Correlations Using a Single Pert-Like Traversal". In *Procs of ICCAD*, 2003.

[9] H. J. Bungartz. "Higher Order Finite Elements on Sparse Grids". In *Technical Report SFB-Bericht Nr. 342/01/95 A*, Institut fur Informatik, TU Munich 1995.

[10] H. J. Bungartz and T. Dornseifer. "Sparse Grids: Recent Developments for Elliptic Partial Differential Equations". In *Technical Report TUM-19702, SFB-Bericht Nr. 342/02/97 A*, Institut fur Informatik, TU Munich 1997.

[11] J. Jess et al. "Statistical Timing for Parametric Yield Prediction of Digital Integrated Circuits". In *Procs of DAC*, 2003.

[12] R. Ahmadi and F. Najm. "Timing Analysis in Presence of Power Supply and Ground Voltage Variations". In *Procs of ICCAD*, 2003.

[13] S. Bhardwaj et al. "TAU: Timing Analysis Under Uncertainty". In *Procs of ICCAD*, 2003.