A New Method for Design of Robust Digital Circuits

Dinesh Patil, Sunghee Yun, Seung-Jean Kim, Alvin Cheung, Mark Horowitz and Stephen Boyd
Department of Electrical Engineering, Stanford University, Stanford, CA 94305-9510
{*ddpatil,sunghee.yun,akcheung,sjkim,horowitz,boyd*}*@stanford.edu*

Abstract

As technology continues to scale beyond 100nm, there is a significant increase in performance uncertainty of CMOS logic due to process and environmental variations. Traditional circuit optimization methods assuming deterministic gate delays produce a flat "wall" of equally critical paths, resulting in variation-sensitive designs. This paper describes a new method for sizing of digital circuits, with uncertain gate delays, to minimize their performance variation leading to a higher parametric yield. The method is based on adding margins on each gate delay to account for variations and using a new "soft maximum" function to combine path delays at converging nodes. Using analytic models to predict the means and standard deviations of gate delays as posynomial functions of the device sizes, we create a simple, computationally efficient heuristic for uncertainty-aware sizing of digital circuits via Geometric Programming. Monte-Carlo simulations on custom 32bit adders and ISCAS'85 benchmarks show that about 10% to 20% delay reduction over deterministic sizing methods can be achieved, without any additional cost in area.

1. Introduction

Extensive research has been done on automatic circuit sizing for minimizing delay under an area or power constraint, using both model based [2] and simulation based [1] approaches. While these approaches have been very successful, they assume deterministic gate delay models and invariably result in large number of equally critical paths that form a wall in the path delay histogram [13]. As we scale technology to the sub-100nm feature size, both intrinsic device variations and process lithography control issues are increasing the statistical variability of each gate in a circuit [4]. This delay variation causes the expected delay for a circuit, which is the expected value of the maximum of all the path delays, to grow larger as the wall of critical paths gets taller. Statistical Static Timing Analysis (SSTA) using Monte Carlo simulations show that such deterministic optimization drives the sizing into an extremely variation sensitive corner. On the other hand, guard banding – designing the circuit to meet specs in the worst-case corner, results in an exceedingly pessimistic timing estimate, due to the inherent assumption of complete correlation among all the gates. Figure 1 compares the delay of a deterministically optimized 32 bit adder as estimated by Deterministic Static Timing Analysis (DSTA) using typical and worst case models to the delay distribution of the same circuit obtained after Monte Carlo SSTA.

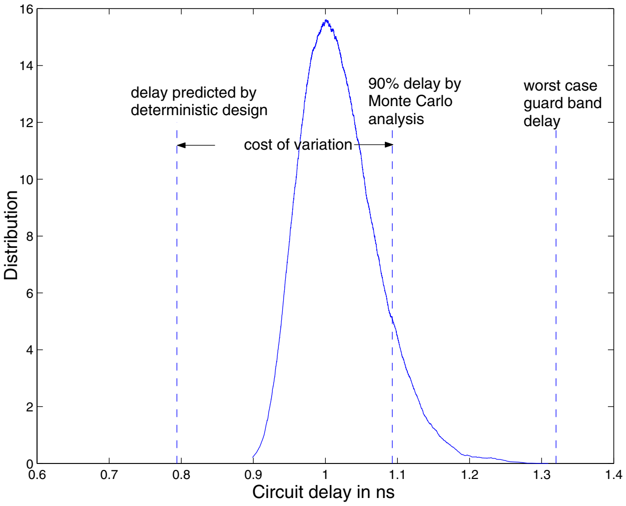

Figure 1. Monte Carlo Analysis on a deterministically sized 32-bit adder

Clearly, the statistical variability causes a significant error, about 20%, if we look at the difference between the estimated deterministic delay and the mean of the expected delay with statistical variations. The error is even larger if we want the delay that say 90% of the distribution will meet. This large error has led a number of researchers to create statistical timing analyzers, using approaches that range from Monte Carlo analysis to propagating delay Probability Distribution Functions (PDFs) through the netlist

[10, 11, 12]. While these techniques have been successfully integrated into SSTA, it is less clear how to extend these techniques to solve the circuit sizing problem. Furthermore, while they are clearly needed to accurately estimate the timing of these circuits, it is less clear that this degree of fidelity is needed to optimize their device sizes.

Our approach is based on the intuition that the circuit sizing problems tend to have large relatively flat minima. The sizer mostly needs to avoid making bad choices (or having variation push the solution into a bad case) rather than choosing the precisely correct value. As a result we took a different approach to the problem. We asked what small changes could we add to the current sizing approach to improve its performance when dealing with circuits with statistical variation. Our goal was to see how well we could do by extending current optimization techniques. Following this approach, the new uncertainty aware sizing algorithm we present here is an extension of the deterministic method. We augment the gate delay models using margins related to the standard deviation. The path delays at converging nodes are combined using the *soft maximum* function in order to correctly capture the statistical behavior of the max of a set of random variables.

The next section provides a quick overview of the sizing problem, and reviews the solution for deterministic circuits. These techniques are then extended in §3 to provide the optimizer some indication of the uncertainty of each gate delay. While the techniques in §3 can be used with any delay model, the models we used for the expected delay and the sigma of the delay are described in §4. These models are then used to produce the results that are shown in §5.

Throughout this paper, bold capital letters, *e.g.*, $\mathbf{R}$, $\mathbf{X}$, and $\mathbf{Y}$, denote random vectors or variables, while the corresponding lower case letters denote their particular realizations. We use $\mu(\mathbf{X})$ to denote the expected value of a random variable $\mathbf{X}$, $\sigma(\mathbf{X})$ to denote its standard deviation, and $\mathbf{Q}_\alpha(\mathbf{X})$ to denote the α^{th} percentile point on its probability distribution curve. For a vector v, the i^{th} component is denoted as v_i.

2. Circuit Sizing

Assuming the objective is minimizing the circuit delay T_{cycle} (*i.e.*, the maximum of the set T_{O} of signal arrival times at the circuit outputs), under constraints on total area A, the deterministic optimization problem can be formulated as shown below (1) [1].

$$\begin{array}{ll} \text{minimize} & T_{\text{cycle}} \\ \text{subject to} & \max(T_o) \leq T_{\text{cycle}}, \\ & A \leq A^{\max}, \\ & f_j(w) \leq 1, \quad j = 1, \ldots, m. \end{array} \tag{1}$$

Here $A^{\max}$ is a given limit on total circuit area, w is the vector of transistor sizes (or cell sizes in case of standard cell design) and $f_j(w)$ represent, for each gate j, a set of constraints on its device sizes, signal slopes, and delay propagation from its inputs to the output.

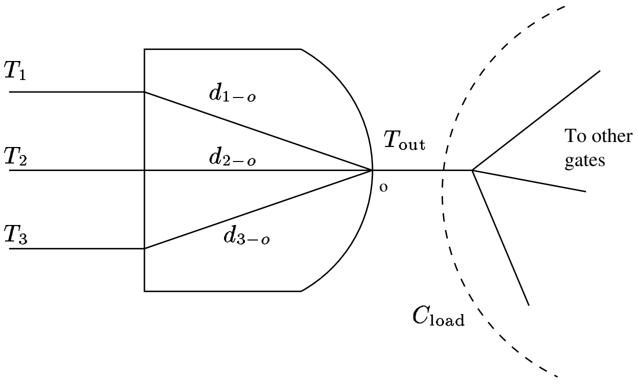

Figure 2. Gate delay constraints for problem (1)

For instance, Figure 2 shows a typical gate for which we can write:

$$T_{\text{out}} = \max_{i=1,2,3}(T_i + d_{i-o}). \tag{2}$$

where T_i is the signal arrival time of input i and d_{i-o}, the typical gate delay from input i to the output o is a function of the load capacitance C_{load}, transistor sizes w, channel length L, supply voltage V_{dd}, threshold voltage V_{th}, oxide thickness t_{ox}, and mobility μ_0:

$$d_{i-o} = f(C_{\text{load}}, w, L, V_{\text{dd}}, V_{\text{th}}, t_{\text{ox}}, \ldots). \tag{3}$$

Rising and falling delays are considered separately. The complexity of solving this optimization problem depends on the form of the d_{i-o} and other constraints. In particular, if d_{i-o} is a generalized posynomial [20], this becomes a geometric program, which can be efficiently solved using convex optimization techniques.

3. Sizing for Robust Design

In presence of variations each d_{i-o} is a random variable with mean $\mu(d_{i-o})$ given by (3) and the standard deviation $\sigma(d_{i-o})$ modeled as a function of $C_{\text{load}}, w, L, V_{\text{dd}}, V_{\text{th}}, \sigma(V_{\text{th}}), \sigma(\mu_0), \mu_0$ etc..

The deterministic algorithm only considers $\mu(d_{i-o})$ and results in many equally critical paths, which is mainly responsible for the statistical delay spread. The exact statistical sizing problem considering detailed distribution and propagation of each gate delay is computationally intractable. We want to achieve statistical tuning without having to propagate PDFs but instead, propagate a delay number that represents the tail of the distribution.

3.1. Augmenting the mean delay

We propose to use gate delays D_{i-o} defined as

$$D_{i-o} = \mu(d_{i-o}) + \kappa_j \sigma(d_{i-o}) \tag{4}$$

in (2) in place of d_{i-o}. In other words D_{i-o} for the j^{th} gate includes extra margins (scaled $\sigma(d_{i-o})$) to account for the variation and uncertainty in the gates delays. We call κ_j *margin coefficients*. This can be interpreted as adding a delay penalty term to each gate that is proportional to its delay uncertainty.

3.2. Use of soft maximum

Since T_{out} in (2) is a maximum of a set of input delays that are random, the distribution of T_{out} is shifted to the right of all the input delay distributions. This shift is more pronounced when several of the input arrival time distributions are near the maximum, and negligible when, say, one of the inputs arrive much later than the others. To take into account the right shift caused by taking the maximum of a set of random variables, we propose to use a *soft maximum* function smax_p defined as

$$\text{smax}_p(x) = \left(\sum |x_i|^p\right)^{1/p},$$

where p is the exponent that represents the penalty for closeness of arguments and the sum accounts for increase in uncertainty with every extra input. This steers the optimizer away from making the paths equally critical. The soft max retains in spirit the fact that under variations even a path with smaller mean can contribute to the delay spread at the converging node, while it asymptotically approaches the max function.

Combining the two techniques we can write (2) for the gate in Figure 2 as:

$$T_{\text{out}} = \left(\sum_{i=1,2,3} |T_i + D_{i-o}|^p\right)^{1/p}.$$

Since these techniques retain the computational merits of the deterministic sizing problem (like sparsity), the algorithm is easily scalable to larger circuits. Moreover, if the $\mu(d_{i-o})$ and $\sigma(d_{i-o})$ of gate delays are generalized posynomials (which is the case if we use the Elmore delay model [6], the velocity saturated delay model [8], or curve fit model [2]), then the problem can be cast as a generalized geometric program (GGP) [20], which can be solved globally with great efficiency. A crude search loop in the $p - \kappa$ space around the basic optimization routine can easily be implemented to obtain the best statistical sizing (as validated by SSTA).

3.3. Validation

Consider two (Gaussian for convenience) random variables $\mathbf{R}_1$ and $\mathbf{R}_2$. Let $\mu(\mathbf{R}_1) = 1$ and $\sigma(\mathbf{R}_1) = 0.1$ while we sweep $\mu(\mathbf{R}_2)$ from 0.7 to 1.3 and $\sigma(\mathbf{R}_2) \in (0.05, 0.1, 0.15)$. Let $\mathbf{Y} = \max(\mathbf{R}_1, \mathbf{R}_2)$. We find $\mathbf{Q}_{68}(\mathbf{Y})$ and $\mathbf{Q}_{95}(\mathbf{Y})$ (*i.e.* $\mathbf{Q}_{1\sigma}(\mathbf{Y})$ and $\mathbf{Q}_{2\sigma}(\mathbf{Y})$) of the random variable $\mathbf{Y}$ using Monte Carlo samples. These are plotted as solid curves of varying $\mu(\mathbf{R}_2)$ for three values of $\sigma(\mathbf{R}_2)$ in figure 3. We then define $\widehat{\mathbf{Q}}_{68}(\mathbf{Y})$ and $\widehat{\mathbf{Q}}_{95}(\mathbf{Y})$ as:

$$\begin{aligned} \widehat{\mathbf{Q}}_{68}(Y) &= \text{smax}_{50}(\mu(\mathbf{R}_i) + 0.5\sigma(\mathbf{R}_i)), \quad i = 1,2 \\ \widehat{\mathbf{Q}}_{95}(Y) &= \text{smax}_{40}(\mu(\mathbf{R}_i) + 1.65\sigma(\mathbf{R}_i)), \quad i = 1,2 \end{aligned} \tag{5}$$

in order to fit $\mathbf{Q}_{68}(\mathbf{Y})$ and $\mathbf{Q}_{95}(\mathbf{Y})$ by choosing the right p and κ (plotted as dashed curves).

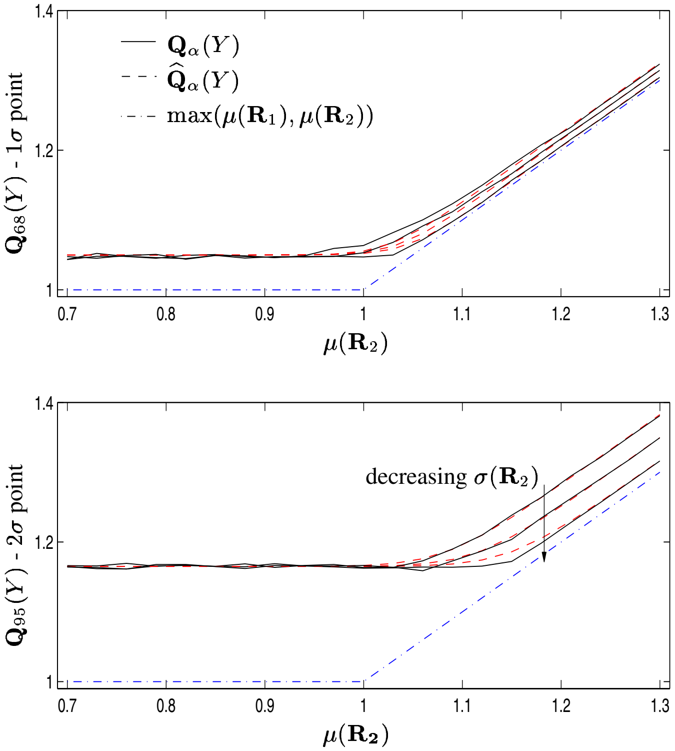

Figure 3. Soft max with margins – validation for 1σ and 2σ points

The plots in figures 3 shows that our smax_p and $\kappa\sigma$ margins give close estimates of the $\mathbf{Q}_{68}(\mathbf{Y})$ and $\mathbf{Q}_{95}(\mathbf{Y})$ for specific values of p and κ. Here $\mathbf{R}_1$ and $\mathbf{R}_2$ represent the delay of two converging critical paths which can vary by 30% from each other in their mean and differ by 50% in their standard deviation. For simplicity, we consider a uniform p and κ for all gates. A value of p between 30 and

50 and margin coefficient between 0.5 and 2.5 give good statistical sizing in most circuits.

Of course, in a real netlist, the number we obtain for the signal arrival time T at any net, using our heuristic, is certainly not the exact $\mathbf{Q}_\alpha(T)$ (for a specific α) on its timing distribution. It just represents a measure of the criticality of the arrival time to the overall delay. The timing results we present are always from a SSTA done after the robust optimization; SSTA is the only trustworthy method for comparing results. We use our soft max function, and the simple augmented delay expression only to *design* the circuit, and not to *analyze* it.

4. Statistical Delay Model

While the above techniques can be used with a number of different timing models, we have been using a simple analytical model to estimate delays. Ideal quadratic transistors can be nicely modeled as resistors [7], but all modern transistors are current velocity saturated. Although this makes the analysis a little more difficult, it is still possible to create simple, accurate analytical timing models, that are compatible with GP solvers. Our model uses Channel Connected Component (CCC) as the basic gate structure. This is a group of transistors that have their source/drain connected, with some transistors connecting to $V_{\rm dd}$ and others to ground. For full custom designs, each transistor can be optimized individually, while in cell based designs, all the transistors in the cell are sized together. In ISCAS'85 benchmarks, each cell may contain one or more CCCs and cell sizes are the design variables.

4.1. Mean Delay Model

We have extended the Meyer velocity saturated current model described in [5, 8] to obtain the delay of CMOS CCCs. In this model, the current I_d through a MOS transistor is:

$$I_d = \frac{W v_{\rm sat} C_{\rm ox} V_{\rm od}^2}{V_{\rm od} + E_{\rm c} L}$$

where $V_{\rm od} = V_{\rm dd} - V_{\rm th}$, $v_{\rm sat}$ is the saturation velocity and $E_{\rm c}$ is the electric field that sets the onset of velocity saturation.

To use this model for gates, we need to find the effective W and effective L for a chain of n NMOS transistors. We estimate the current by creating an effective transistor where

$$\begin{aligned} W_{\rm eff} &= \min(w_1, \ldots, w_n), \\ L_{\rm eff} &= L_{\rm min} W_{\rm eff} \sum_{i=1}^{n} 1/w_i. \end{aligned}$$

Using this current equation, we estimate the fall delay (time to discharge the output to $0.5V_{\rm dd}$) τ_d by

$$\tau_d = \frac{C_{\rm load} V_{\rm dd}}{2 I_d} + k\tau_{\rm in},$$

where $\tau_{\rm in}$ is the input slope and k is a constant determined by $V_{\rm dd}$ and $V_{\rm th}$. While formulating the gate delay constraints, the added delay due to $\tau_{\rm in}$ is absorbed in the delay of the fan-in gate.

If the input is at the bottom of the chain, then it has to discharge all the intermediate nodes. In this case we decompose the fall delay as sum of fall delays where each intermediate capacitor is discharged by the chain below it, just like in the Elmore delay calculation. The accuracy of the delay model remains well within 8% for chains of upto 4 transistors, for reasonable $C_{\rm load}$ and signal slews ($\tau_{\rm in}$). Similar expressions can be written for the rise delay through PMOS chains

The d_{i-o} can be formed by considering all chains that contain the input i and help to drive the output o and taking the maximum of these delays, for static problem formulation. The mean delay of a CCC thus obtained is a generalized posynomial [20] of its transistor widths. For ISCAS'85 circuits, the cell size are design variables. The cell delay models are obtained using posynomial fitting [20] on the cell library data.

4.2 Standard Deviation Model

We use Pelgrom's model [3] for the variation of a device current, which states that parameter variations tend to reduce as the area of the fabricated MOS structure increases. We extend this idea to the chain of n transistors by expressing the relative $\sigma(I_d)$ as

$$\frac{\sigma(I_d)}{I_d} = \frac{A_p}{\sqrt{L_{\rm eff} W_{\rm eff}}}$$

where A_p is a constant depending on the fabrication process. Thus the variance of the drain current of an n transistor chain is inversely proportional to the "electrically" effective area of the chain. The effective area is a weighted sum of the device areas, so that the contribution to the $\sigma(I_d)$ variations is weighted according to the contribution to I_d.

From $\sigma(I_d)$, the standard deviation of delay ($\sigma(d_{i-o})$) is obtained as

$$\sigma(d_{i-o}) = \left| \frac{\partial d_{i-o}}{\partial I_d} \right| \sigma(I_d).$$

For ISCAS'85 benchmarks we use

$$\frac{\sigma(d_{i-o})}{d_{i-o}} = \frac{A_p}{\sqrt{s}}$$

where s is the footprint area of the cell's layout.

For simplicity, we have not included the variation in delay due to wire width variation or variation in C_{load} of fan-out gates. These can be easily included in the detailed framework. Also, correlation between gates can be incorporated by adding additional margins to d_{i-o} in (4).

5. Results

The optimization algorithm was tested on two custom 32 bit adders, a Kogge-Stone (KS) and a Ladner-Fischer (LF) [17] designed in TSMC 0.18μ 1.8V CMOS with an FO4 delay of 80ps and a ISCAS'85 benchmark in bulk TSMC 0.13μ 1.08V with an FO4 delay of 130ps. For a chain of n transistors we used $\sigma(I_d)/I_d$ of 15% for $L_{\text{eff}}W_{\text{eff}}$ equivalent to that of a single minimum length transistor with $W = 1\mu$. For ISCAS'85 cell based design the 15% variation was for the minimum sized cell. Internal wire capacitances, wherever significant, are also included in the optimization. The circuits were optimized under identical load, area and other constraints for deterministic and statistical cases. For custom circuits, the area is the sum of the widths of all devices while for ISCAS circuits, it is the sum of all cell areas modeled from the library as a function of cell sizes. Signal slew rate constraints are indirectly provided by constraining the delay per logic stage, which is GGP friendly. The optimizations are done using the MOSEK [21] convex optimization package.

The results of Monte Carlo timing analysis are shown in Table 1.

Table 1. Monte Carlo timing analysis

circuit	det. sizing $\mathbf{Q}_{95}(T_{\text{cycle}})$ in ns (FO4)	stat. sizing $\mathbf{Q}_{95}(T_{\text{cycle}})$ in ns (FO4)	improvement in timing
32-bit LF adder	1.06 (13.3)	0.84 (10.5)	20.6%
32-bit KS adder	0.98 (12.3)	0.81 (10.1)	17.7%
ISCAS c880	2.2 (16.9)	1.99 (15.3)	9.5%

Figure 4 shows the PDFs of the delay of 32-bit LF adder[17] along with the $\mathbf{Q}_{95}$ points. The improvement in $\mathbf{Q}_{95}(T_{\text{cycle}})$ is quite significant after statistical sizing. The improvement in c880 is not significant due to lack of freedom caused by having only one size represent the entire cell and one cell containing possibly multiple CCCs sized to a fixed ratio.

We have observed that the results are very weakly dependent on the kind of distribution, but are slightly dependent on the model used for $\sigma(d_{i-o})$. The improvement increases as $\sigma(d_{i-o})$ depends more strongly on the effective device area than that provided by the Pelgrom's model. Also the improvement increases with increasing number of parallel outputs. The region around the optimal is largely flat, so that a change of 5 around the optimal p or 0.5 around the optimal κ results in a change of only a few ps in the $\mathbf{Q}_{95}(T_{\text{cycle}})$. So a crude search suffices, drastically reducing the optimization time. Each iteration typically consists of about 300s for optimization and 30s for 10000 sample Monte Carlo on a 2GHz Pentium PC with 1GB memory, for the presented circuits.

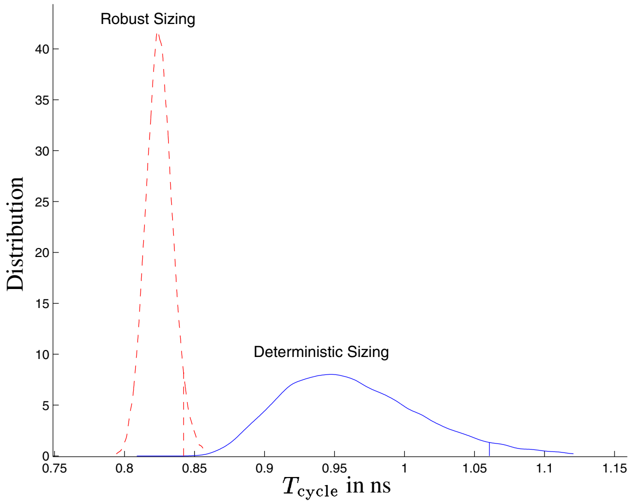

Figure 4. Deterministic versus robust sizing: Delay PDF for a 32-bit LF adder.

Figure 5 shows the μ vs. σ scatter plots of all the path delays in the LF adder for deterministic and statistical sizing. Clearly, the wall in the deterministic case is broken and the variation reduced for the statistical case at the expense of increased mean deterministic delay.

6. Conclusions

Statistical variations in device parameters will likely continue to worsen as we scale technology. It will be critical to account for these variation in both analog and digital circuits. While accurately accounting for uncertainty while sizing a digital circuit is difficult, we have shown that a few simple heuristics improve the expected performance of the resulting circuit, and can be easily fit into today's optimization tools. Our method adds a penalty to each gate that is proportional to its uncertainty, and then changes the max function to account for the added delay that occurs when a set uncertain inputs with similar expected times combine. Our method attempts to strike a balance between the goal of having the smallest delay for a given area or power, or vice versa, and preventing excessive downsizing of non critical

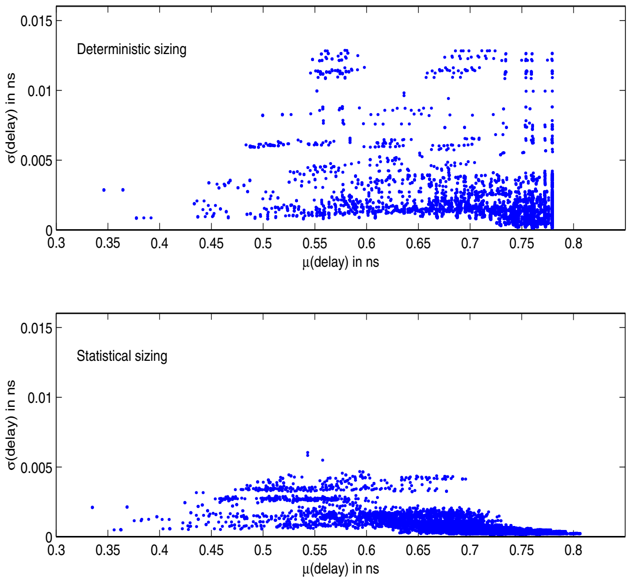

Figure 5. μ–σ scatter plot for all paths of 32bit LF adder.

paths that lead to new critical paths. We are currently working on extending this framework to optimize other transistor and circuit parameters, like $V_{\rm dd}$ and $V_{\rm th}$ to better optimize our designs.

Acknowledgements

This work was funded in part by C2S2, the MARCO Focus Center for Circuit and System Solution, under MARCO contract 2003-CT-888 and by Philips Inc.

References

[1] A. Conn, I. Elfadel, W. Molzen, P. O'Brien, P. Strenski, C. Visweswariah, and C. Whan, "Gradient-based optimization of custom circuits using a static-timing formulation", *Proc. Design Automation Conference (DAC)*, 1999, pp. 452-459.

[2] S. Sapetnekar, V. Rao, P. Vaidya, and S.-M. Kang, "An exact solution to the transistor sizing problem for CMOS circuits using convex optimization", *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, pp. 1621-1634, November 1993.

[3] M. Pelgrom, C. Duinmaijer, and A. Welbers, "Matching properties of MOS transistors", *IEEE Journal of Solid State Circuits*, pp. 1433-1440, October 1989.

[4] S. Nassif, "Within-chip variability analysis", *Proc. of IEDM*, 1998, p. 283.

[5] K. Chen, C. Hu, P. Fang, M. Lin, and D. Wollesen, "Predicting CMOS speed with gate oxide and voltage scaling and interconnect loading effects", *IEEE Transactions on Electron Devices*, pp. 1951-1957, November 1997.

[6] J. Rubenstein, P. Penfield and M. Horowitz, "Signal delay in RC tree networks", *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, pp. 202-211, July 1983.

[7] M. A. Horowitz, *Timing Models for MOS circuits*, *Ph.D. Thesis*, Stanford University, 1983.

[8] K.-Y. Toh, P.-K. Ko, and R. Meyer, "An engineering model for short channel MOS devices", *IEEE Journal of Solid-State Circuits*, pp. 950-958, August 1988.

[9] S. Borkar, T. Karnik, S. Narendra, J. Tschanz, A. Keshabvarzi and V. De, "Parameter variations and impact on circuits and architecture", *Proc. Design Automation Conference (DAC)*, 2002, pp. 58-63.

[10] M. Orshansky and A. Bandyopadhyay, "Fast statistical timing analysis handling arbitrary delay correlations", *Proc. Design Automation Conference (DAC)*, 2004, pp. 337-342.

[11] A. Gattiker, S. Nassif, R. Dinakar, and C. Long, "Timing yield estimation from static timing analysis", *Proc. International Symposium on Quality Electronic Design (ISQED)*, 2001, pp. 437-442.

[12] C. Vishweswariah, K. Ravindran, K. Kalafala, S. Walker, and S. Narayan, "First-order incremental block-based statistical timing analysis", *Proc. Design Automation Conference (DAC)*, 2004, pp. 331-336.

[13] X. Bai, C. Vishweswariah, P. Strenski, and D. Hathaway, "Uncertainty-aware circuit tuning", *Proc. Design Automation Conference (DAC)*, 2002, pp. 338-342.

[14] M. Hashimoto and H. Onodera, "A performance optimization method by gate sizing using statistical static timing analysis", *Proc. ACM/SIGDA International Symposium on Physical Design*, 2000, pp. 111-116.

[15] S. Raj, S. Vrudhula, and J. Wang, "A methodology to improve timing yield in the presence of process variations", *Proc. Design Automation Conference (DAC)*, 2004, pp. 448-453.

[16] E. Jacobs and M. Berkelaar, "Gate sizing using a statistical delay model", *Proc. of Design, Automation, and Test in Europe*, 2000, pp. 283-291.

[17] S. Knowles, "A family of adders", *Proc. 15th IEEE symposium on Computer Arithmetic*, 2001, pp. 177-182.

[18] S. Kim, S. Boyd, S. Yun, D. Patil, and M. Horowitz, "A heuristic for optimizing stochastic activity networks with applications to statistical digital circuit design", Technical report, Stanford University, Stanford, CA 94305, 2004. Available from `www.stanford.edu/~boyd/heur_san_opt.html`.

[19] S. Boyd, S. Kim, L. Vandenberghe, and A. Hassibi, "A tutorial on geometric programming", Technical report, Stanford University, Stanford, CA 94305, 2004. Available from `www.stanford.edu/~boyd/gp_tutorial.html`.

[20] S. Boyd, and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, 2003.

[21] MOSEK ApS, *The MOSEK Optimization Tools Version 2.5. User's Manual and Reference*, 2002. Available from `www.mosek.com`.