Novel Sizing Algorithm for Yield Improvement under Process Variation in Nanometer Technology

Seung Hoon Choi
Intel Corporation
Hillsboro, OR 97124, U. S. A.
seung.h.choi@intel.com

Bipul C. Paul
Dept. of ECE, Purdue University
W. Lafayette, IN 47907, U. S. A.
paulb@ecn.purdue.edu

Kaushik Roy
Dept. of ECE, Purdue University
W. Lafayette, IN 47907, U. S. A.
kaushik@ecn.purdue.edu

ABSTRACT

Due to process parameter variations, a large variability in circuit delay occurs in scaled technologies affecting the yield. In this paper, we propose a sizing algorithm to ensure the speed of a circuit under process variation with a certain degree of confidence while maintaining the area and power budget within a limit. This algorithm estimates the variation in circuit delay using statistical timing analysis considering both inter- and intra-die process variation and resizes the circuit to achieve a desired yield. Experimental results on several benchmark circuits show that one can achieve up to 19% savings in area (power) using our algorithm compared to the worst-case design.

Categories and Subject Descriptors

B.8.2[**Performance and Reliability**]:Performance Analysis and Design Aids

General Terms: Algorithms, Performance, Design, Reliability.

1. INTRODUCTION

As silicon industry is moving towards the end of the roadmap, the device parameters (such as channel length, oxide thickness, threshold voltage, random placement of dopants in channel, etc) are expected to have large variations. Consequently, a large variability in performance among different chips is expected. The process variations can be classified as systematic or random. While systematic variations are deterministic in nature and are caused by the structure of a particular gate and its topological environment, random variations are unpredictable in nature. Random variations include variations in the effective channel length of devices, doping profiles, oxide thickness and transistor width. Variations in doping profile are very important in advanced technologies because it may lead to potentially large change in threshold voltage [1]. Furthermore, intrinsic fluctuations are independent of transistor location on a chip. The process parameter fluctuations cannot be eliminated by external control of the manufacturing process and hence, a statistical design methodology is required considering the randomness of the process parameter variation. Various aspects of the process parameter variation including methodology, analysis, synthesis and modeling are addressed in [2].

Conventional sizing tools size the gates to optimize area and power consumption while meeting the desired delay constraint [3], [4]. Usually these tools find the critical points of the circuit through static timing analysis, which affect the critical path delay. The tool then sizes the transistor widths to meet the desired delay constraint while keeping the power consumption and area within a limit. However, due to random process parameter variation, a large number of chips may not meet the required delay. Consider an example pdf (probability density function) of delay shown in Figure 1due to process variation. In this example, the distribution is assumed to be normal [5]. This figure shows that 50% of the total number of dies will not meet the desired delay constraint, which will affect the final yield drastically. One way to counter this effect is to set the target delay considering worst case process variation. For example, one can choose the 6σ point in Figure 1as the target delay for designing under worst case process variation. Consequently, while the yield is expected to improve significantly, the area and power overhead to meet the worst case delay constraint may not be acceptable. This is because, only a very few dies will have the worst case delay due to process variation, and setting the target delay based on those dies will result in unacceptable power consumption in most of the dies. Hence, beyond a certain point the improvement in yield will be masked by the increase in the area and the power overhead.

Furthermore, resizing the gate also changes the delay spectrum of the circuit (i.e. σ also changes along with the mean of delay distribution). This is because the variation in the transistor threshold voltage (hence, the variation in delay) is a strong function of transistor width due to the random placement of dopants in very short-channel devices [6]. A proper design technique is therefore, necessary to achieve optimum yield with minimum increase in the area and power overhead.

Several attempts have been made to model the effect of process variation on delay using *statistical* timing analysis [7],[8],[9]. These analyses considered inter- and intra-die process variation and in some cases also modeled spatial correlations between transistors [8]. However, attempts have rarely been made to size the gates considering the statistical nature of process variation. In [10], the gate sizing considering a statistical delay model was proposed using non-linear programming. However, the use of non-linear programming makes it less practical in real circuits.

In this paper, we propose a statistical design technique considering both inter- and intra-die variation of process parameters. The idea is to resize the transistor widths with minimal increase in area and power consumption while improving the confidence that the circuit meets the delay constraint under process variation. We have developed a sizing

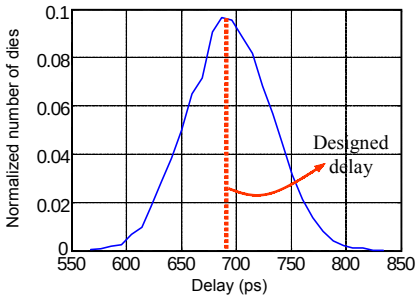


Figure 1. pdf of delay due to process variation: an example.

This research was supported by DARPA MARCO Gigascale Silicon Research Center (SA3273JB), SRC (1078.001) and Intel Corporation.

DAC 2004, June 7-11, 2004, San Diego, CA, USA.

tool using Lagrangian relaxation algorithm [11] for global optimization of transistor widths. The algorithm first estimates the expected delay variation at the primary output of a given circuit based on *statistical* static timing analysis. Using the delay distribution obtained at the primary output and the desired yield, the algorithm optimizes the area by increasing the size of transistors to achieve desired delay while reducing the transistor sizes in the off critical paths. The contribution of this work is to provide a sizing algorithm to ensure the speed of a circuit under process variations with a certain degree of confidence while keeping the area and power budget within a limit. We consider the variations not only in channel length, width, oxide thickness, threshold voltage of the transistor but the effect of random placement of dopants as the dominant parameters for process variation. While the variations in channel length, width and oxide thickness are expected to have spatial correlation between adjacent transistors [8], random placement of dopants make every transistor in the circuit independent. We model these variations in our analysis and incorporate them into the sizing tool. We also consider inter- and intra-die variation of process parameters to achieve more realistic design. We used the proposed sizing tool to synthesize several ISCAS benchmark circuits and compare the estimated yield with the circuit synthesized by conventional sizing tool.

The rest of the paper is organized as follows. In section 2, we discuss the *statistical* timing analysis of a circuit considering both inter- and intra-die variations. Section 3 describes the sizing algorithm based on Lagrangian relaxation in detail. In section 4, we discuss the experimental results on ISCAS benchmark circuits. Section 5 draws the conclusion.

2. STATISTICAL TIMING ANALYSIS

Statistical static timing analysis is performed to estimate the variability in circuit delay under process variation. Process variations can be categorized as inter-die and intra-die variations. Due to inter-die variations, the same device on a chip can have different characteristics across different dies (i.e., dies from one wafer, from wafer to wafer, and from wafer lot to wafer lot). Intra-die variations, on the other hand, are the variations of transistor characteristics within a single chip. Both inter- and intra-die variations are expected to be truly random in nature in future technologies. While intra-die variations in terms of transistor length, transistor width and oxide thickness, are expected to exhibit spatial correlations among devices located close to each other, random placement of dopants in sub-50 nanometer transistors is expected to make every transistor in a die independent in terms of threshold voltage. We incorporate both inter- and intra-die variations in our timing analysis.

The most accurate way of incorporating the process variation effects into timing analysis is to perform a full-scale transistor-level Monte-Carlo simulation of a circuit, which requires large computational overhead. Hence, in our analysis the effect of process parameter variations on the gate delay is pre-characterized and accessed on the fly during *statistical* timing analysis. The pre-characterization table contains statistical information on the delay of a gate considering process variation. It is assumed that inter- and intra-die process variations are statistically independent [8]. This reduces the complexity of *statistical* timing analysis because the effects of inter- and intra-die process parameter variation on delay can be analyzed in isolation.

For example, we model the variation in transistor length L_{total} as the sum of inter-die variation (L_{inter}) and intra-die variation (ΔL_{intra}) as follows.

$$\sigma^2_{L_{total}} = \sigma^2_{L_{inter}} + \sigma^2_{\Delta L_{intra}}$$

where σ is the standard deviation. Accordingly, the effect of the variation in transistor length on the delay can be simplified to [8]:

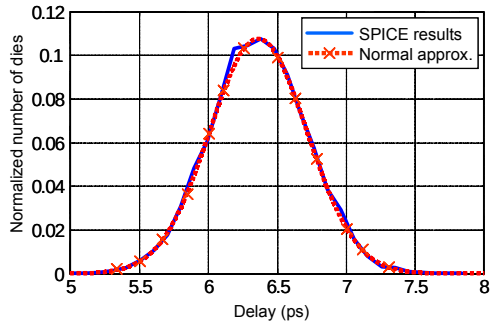
$$\sigma^2_{delay,total} = \sigma^2_{delay,inter} + \sigma^2_{delay,intra}$$

The effects of variation in the threshold voltage, oxide thickness, and transistor width on delay are also incorporated in a similar way.

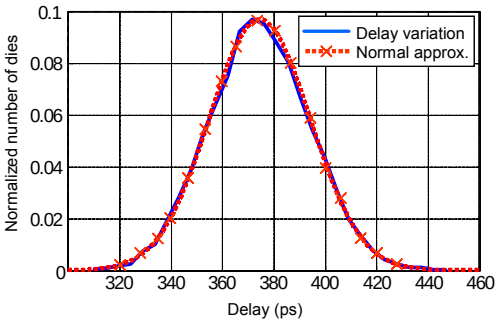
As explained above, inter- and intra-die variations are analyzed in isolation. That is, $\sigma_{delay,inter}$ and $\sigma_{delay,intra}$ are independently calculated and combined to obtain the overall distribution of delay. During *statistical* timing analysis, the signal arrival time is calculated at each gate by propagating the delay from the primary input in the circuit. Under process variation, the signal arrival time is also a distribution which is propagated during the timing analysis. The variation in the arrival time is obtained from the statistical information stored in the pre-characterization table. We maintain two tables - one for inter-die variation and the other for intra-die variation. In the following subsections, we explain how to propagate the effect of process variation during *statistical* timing analysis considering both inter- and intra-die variation.

2.1 Inter-die variation

Considering that transistor parameters remain constant within a single die, evaluation of the effect of inter-die variation on circuit delay is straightforward. For example, due to inter-die variation, if the device length in a die becomes *'L+ΔL'*, this will remain the same for all transistors in that die. Hence, under inter-die process variations, all identical gates in a die will have the same delay. It is therefore, easy to pre-characterize the delay of gates for all possible combination of process parameters. We generate the pre-characterization table through Monte-Carlo simulation using SPICE considering the inter-die variations in threshold voltage, oxide thickness, transistor width and transistor length. We assume normal distribution [5] for all inter-die process parameter variations. It is also assumed that the corresponding gate delay variation can be modeled as normal distribution. Figure 2 (a) shows one example of delay variation for an inverter obtained using



(a) Delay variation of a gate



(b) Delay variation of a die

Figure 2. Normal approximation of inter-die delay variation

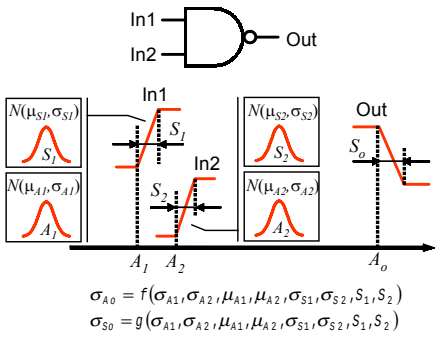


Figure 3. 2-input NAND gate example of calculating intra-die delay variation

SPICE Monte-Carlo simulation for BPTM 70*nm* technology [12] under inter-die variation. It can be observed from the figure that the above assumption is reasonable.

Each point in the delay distribution represents the delay of a gate in a die for a corresponding process corner. The circuit delay is calculated for all process corners and the overall delay distribution for the circuit under inter-die process variation is obtained. Figure 2 (b) shows the delay distribution at the primary output of an example circuit (ISCAS c432) under inter-die variation. It can be seen from the figure that the distribution can also be approximated as normal distribution.

2.2 Intra-die variation

Unlike inter-die variation, transistors within a die are expected to have different process parameters under intra-die variation. In static timing analysis, signal arrival time at the output of a gate is calculated by adding the gate delay to the signal arrival time at the input. Considering intra-die process variation, both gate delay and input signal arrival time are to be considered as random variables. For example, in a two-input gate, the random variable A_o, which represents the worst-case arrival time at the output, can be expressed as

$$A_o = \max(A_1 + D_1, A_2 + D_2) \qquad (1)$$

where A_1 and A_2 are the random variables which represent input arrival times, respectively. Random variables D_1 and D_2 represent the corresponding pin-to-pin delays. Then $f_{Ao}(x)$, the probability density function for A_o, can be calculated as below.

$$f_{Ao}(x) = f_{A_1+D_1}(x)F_{A_2+D_2}(x) + F_{A_1+D_1}(x)f_{A_2+D_2}(x)$$

$F_A(x)$ represents the probability that $A \le x$. Although we assume that A_1 and D_1 (A_2 and D_2) have normal distributions, this does not guarantee that A_o can also be modeled by normal distribution. However, authors in [13] showed that the error in assuming normal distribution for A_o is negligible. Therefore, in our analysis, A_o is assumed to have normal distribution.

The relationship shown in eq. (1) imposes additional constraints on the pre-characterization of multiple input gates under intra-die variation. Unlike the switching under inter-die variation (where both input arrival time and slope are considered to be a single value in a die), both input arrival time and slope are now represented by a distribution. Let us consider the switching of a two-input NAND gate shown in Figure 3. The arrival time at the output, A_0, depends on the input signals, *In1* and *In2*, and the gate delay. Input signals consist of signal arrival time (A_1, A_2) and the slope (S_1, S_2). In our analysis, both signal arrival time and slope are considered as normal distributions. Therefore, the statistical property (σ_{Ao}, σ_{So}) of the signal at the output of the gate is a function of input arrival times, slopes and temporal correlations between them. Considering this, we generate the pre-characterization table for σ (intra-die variation), while the mean of the arrival time and slope at the output are calculated on the fly using Sakurai's delay model [15]. For the ease of analysis, the temporal proximity of two latest-arriving inputs is considered for multiple input gates.

2.3 Random placement of dopants

In scaled CMOS devices, there exists a statistical fluctuation in the number of dopants, which can be translated into a threshold-voltage variation [6]. This discrete dopant effect on threshold voltage variation is incorporated by employing the following equation [6] into the simulator while generating the pre-characterization table for both inter- and intra-die variation.

$$\sigma_{Vth} = \frac{q}{C_{ox}}\sqrt{\frac{N_a W_{dm}^0}{3LW}}$$

σ_{Vth} represents the standard deviation of threshold variation due to random placement of dopants, q is electron charge, C_{ox} is the oxide capacitance, N_a is the substrate doping concentration, and W_{dm}^0 represents the maximum depletion layer width. L and W are the channel length and width of the transistor, respectively.

3. SIZING ALGORITHM FOR YIELD IMPROVEMENT

In this section, a gate-sizing algorithm is proposed to improve the *yield* of a circuit under process variation. The algorithm proposed here is based on a well-known technique to a nonlinear optimization problem: Lagrangian relaxation (LR) [10]. First, we explain the sizing algorithm based on LR and then describe our proposed algorithm for sizing considering process variation.

3.1 Lagrangian relaxation

Chen et. al [14] proposed the use of LR for simultaneous sizing of gate and interconnects of a combinational circuit to optimize the total area while maintaining a delay constraint. The convergence of the algorithm was proven and the optimality was verified. In our experiments, it is assumed that there are no interconnect components in the circuit. However, this algorithm can be extended to incorporate the interconnect sizing as well, as explained in [14].

Figure 4 shows an example circuit representation for LR. The circuit consists of n gates, which are to be resized, and s primary inputs. Logic gates and primary inputs are called components. In addition, we add two virtual components, one connecting all primary inputs (component *9* in Figure 4) and the other connecting primary outputs (component *0*). Therefore, for a circuit with n gates and s inputs, there are $n+s+2$ components. Edge numbers follow their driver gates, i.e., the output of gate i is denoted as edge i. Components and edges are numbered in reverse topological order.

Our objective is to minimize the total area (or equivalently the power consumption) which can be represented by $\Sigma\alpha_i x_i$, $i=1,...,n$ where x_i is the gate size and α_i is an arbitrary constant multiplier for gate i, which can vary depending on the objective of optimization. Conventionally, the gates are sized (i.e., all transistors in a particular gate are sized by the

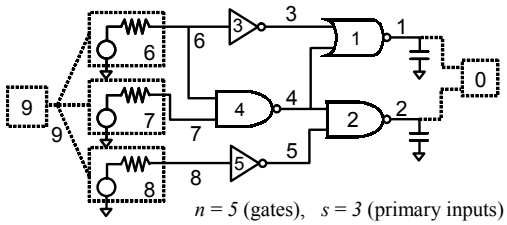


Figure 4. Circuit representation for Lagrangian relaxation

same factor) to achieve minimum area of the circuit while meeting a given delay constraint. In our analysis, in addition to the delay constraint at the primary output, A_0, we also have the *yield* constraint, γ_0 Hence, the sizing problem is formulated as follows.

$$\begin{aligned} &Minimize \sum_{i=1}^{n} \alpha_i x_i \\ &Subject\ to \sum_{i \in p} D_i \le A_0 \qquad \forall p \in P \\ &\qquad L_i \le x_i \le U_i \qquad i = 1,\ldots,n \\ &\qquad Yield\ \gamma \ge \gamma_0 \end{aligned}$$

L_i and U_i represent the lower bound and upper bound of the size of gate i, respectively. P is the set of possible paths in a circuit. D_i represents the delay of gate i in a path p. Compared to the original problem formulated in [14], we added an extra constraint for *yield*. Note that the complexity of the problem is exponentially dependent on the number of components in the circuit ($O(e^n)$). To reduce the complexity to a linear one, the delay constraints on all the paths are transformed into the delay constraints on each gate in the circuit. Therefore, the sizing problem (which is called the primal problem; ***PP***) is redefined as follows.

PP: Sizing for *yield*

$$\begin{aligned} &Minimize \sum_{i=1}^{n} \alpha_i x_i \\ &Subject\ to \\ &\quad a_j \le A_0 \qquad j \in input(0)\ /^*\ outputs\ ^*/ \\ &\quad a_j + D_j \le a_i \qquad i = 1,\ldots,n\ \forall j \in input(i) \\ &\quad D_i \le a_i \qquad i = n+1,\ldots,n+s\ /^*inputs/ \\ &\quad L_i \le x_i \le U_i \qquad i = 1,\ldots,n \\ &\quad Yield\ \gamma \ge \gamma_0 \end{aligned} \tag{2}$$

a_i represents the signal arrival time at edge i and D_i is the delay associated with gate i. Note that the path-based problem is transformed to a global problem where A_0 is now the constraint on the circuit delay, not on any specific path in a circuit.

In solving this problem, ***PP*** is first translated into a mathematical equation introducing a Lagrangian multiplier λ [11] for each constraint on arrival time as follows.

$$\begin{aligned} Minimize: L_\lambda(x,a) = &\sum_{i=1}^{n} \alpha_i x_i + \sum_{j \in input(0)} \lambda_{j0}(a_j - A_0) \\ &+ \sum_{i=1}^{n} \sum_{j \in input(i)} \lambda_{ji}(a_j + D_i - a_i) + \sum_{i=n+1}^{n+s} \lambda_{mi}(D_i - a_i) \end{aligned} \tag{3}$$

λ_{ji} corresponds to the input edge j and output edge i of gate i. m in λ_{mi} is equal to $n+s+1$ (virtual input node).

Minimizing L_λ using LR consists of iterating the following two steps: 1) calculating the optimal size of the circuit for the current λ values and 2) updating λ to the direction of the optimal solution. Calculation of the optimal size involves the sizing of each gate in such a way that L_λ is

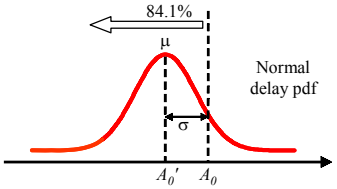


Figure 5 Calculating A_0' in sizing algorithm

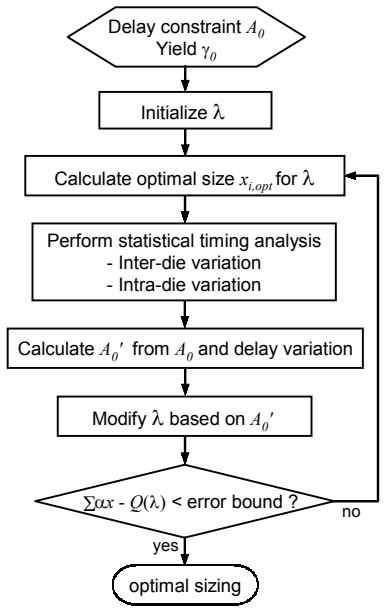


Figure 6. Sizing algorithm for yield improvement

locally minimized. Since D_i in L_λ is a function of gate size x_i, the optimal size of a gate can be obtained by solving $dL_\lambda/dx_i = 0$. In [14], Elmore delay model was used for D_i, however, we use Sakurai's delay model [15] for better accuracy in our analysis. While updating λ in step 2), arrival-time information at each gate input/output of a circuit is utilized. That is, λ for the next iteration is determined by the current status of delay constraints imposed on each gate after calculating the optimal size (step 1)) [14].

Minimizing L_λ provides the minimum size of a circuit while satisfying the delay constraint, A_0 at the primary output. A more detailed explanation including mathematical proofs on sizing algorithm based on LR can be found in [14].

3.2 Sizing considering process variation

In this subsection, we explain the proposed algorithm for resizing considering process variation. Conventional sizing method based on LR algorithm explained above considers the delay constraint A_0 and also the circuit delay as constant values. However, in our analysis delay at the primary output is represented by a probability density function (pdf) considering the process variation. The delay pdf is obtained using *statistical* timing analysis as explained in section 2. Also note that unlike conventional sizing method, we have introduced an additional constraint for *yield* as shown in eq. (2). We incorporate this by modifying the delay constraint based on the delay pdf at the primary output.

Figure 5 shows an example on how the *yield* constraint is introduced into the sizing algorithm. Assuming a normal distribution of delay at the primary output, the modified delay constraint, A_0' should be equal to A_0 - σ in order to achieve, for example, 84.1% yield under process variation, where σ is the standard deviation of the pdf. Similarly, for any different type of delay distribution and *yield* constraint, the delay constraint can be modified accordingly. Figure 6 shows the flow diagram of the proposed sizing algorithm with *yield* consideration. It starts with a given delay constraint A_0 and *yield* constraint γ_0. Then the initial λ values are chosen so that λ in Ω_λ, where Ω_λ represents the set of λ values that satisfies the optimality condition [14]:

$$\sum_{i \in output(k)} \lambda_{ki} = \sum_{j \in input(k)} \lambda_{jk} \qquad for\ 1 \le k \le n+s$$

The algorithm then calculates the optimal size of gates for λ and subsequently *statistical* timing analysis is performed to obtain the delay pdf at the primary output. Note that the delay pdf changes after each iteration because the variation in the threshold voltage (hence, the variation in delay) is a strong function of transistor width [6]. Based on the given *yield* constraint γ_0 and the delay pdf, A_0 is modified to a new constraint A_0' as illustrated above. Unlike conventional sizing methodology, the delay constraint is modified after each iteration during the minimization of L_λ in eq. (3). λ is then updated based on the delay constraint A_0' and the timing information in the circuit. This is repeated until L_λ is minimized, i.e., $\Sigma\alpha x - Q(\lambda)$ is less than a user-defined error bound, where $Q(\lambda)$ is the optimal solution for L_λ at each iteration.

The modification of delay constraint A_0' after each iteration as explained in our algorithm is however, not straightforward in a circuit with multiple primary outputs. For example, let us assume that there are two primary outputs in a circuit; gate *i* and gate *j*. Also assume that *statistical* timing analysis after k^{th} iteration shows that $\mu_i < \mu_j$, where μ_i and μ_j represent the mean delays at the output of the gate *i* and gate *j*, respectively. Under the assumption of normal delay pdf and the target yield of 84.1%, there can be two cases depending on the value of σ (it is also assumed that the delay distributions at different primary output gates are correlated):

1. $\mu_i + \sigma_i < \mu_j + \sigma_j$ (Figure 7 (a))
   New delay constraint A_0' is equal to $A_0 - \sigma_j$ based on the delay variation at the output of gate *j*.
2. $\mu_i + \sigma_i > \mu_j + \sigma_j$ (Figure 7 (b))
   In this case, the calculation of A_0' should be different. Note that although gate *i* is considered to produce the worst-case delay under process variation, conventionally gate *j* provides the worst delay without considering the process variation ($\mu_i < \mu_j$). Hence, the new delay constraint, A_0' for $(k+1)^{th}$ iteration is equal to $A_0 - \sigma_i + (\mu_j - \mu_i)$.

On the other hand, if we modify the delay constraint as explained in case 1), the algorithm will size the gates to meet the delay constraint $A_0 - \sigma_i$ at the output of gate *j* resulting in larger area.

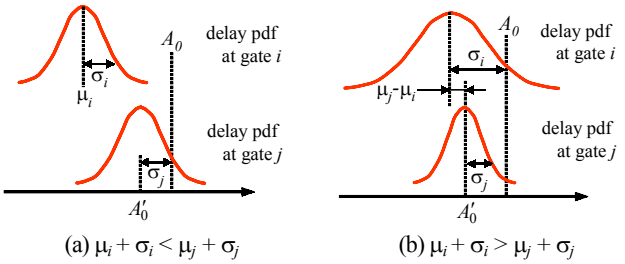
It is mathematically proven in [14] that sizing algorithm using LR always converges to an optimal solution. Considering process variation, we modify the delay constraint in each iteration based on the delay distribution at the primary output ($A_0' = A_0 - \sigma$). Hence, the convergence with A_0' is guaranteed as long as the change in σ is small. It is observed that the change in σ due to the change in circuit size from iteration to iteration is considerably small, which ensures the convergence of the algorithm.

In the following section, we discuss the experimental results on several ISCAS benchmark circuits implemented using our proposed sizing algorithm.

4. EXPERIMENTAL RESULTS

Our proposed sizing algorithm was used to resize several ISCAS benchmark circuits considering the process parameter variations. All the circuits were synthesized with BPTM 70*nm* technology [12]. We assumed 15% (3σ) variation in all the process parameters such as the width, length and the oxide thickness in our analysis for both inter- and intra-die variations. The variation in the transistor threshold voltage was governed by the effect of the random placement of dopants as mentioned in section 2.3.

Considering that our sizing algorithm resizes the circuit to meet a certain delay constraint, it is important to know the possible range of the delay that can be achieved for a circuit and the desired delay constraint. For example, consider the area-vs.-delay curve (Figure 8) for ISCAS benchmark circuit 'c432' obtained using LR sizing algorithm. The area



(a) $\mu_i + \sigma_i < \mu_j + \sigma_j$ (b) $\mu_i + \sigma_i > \mu_j + \sigma_j$

Figure 7. Calculation of A_0' for different scenarios

(the sum of transistor width) in the plot represents the minimum circuit size for the corresponding delay. In the plot, 'min delay' represents the minimum delay that can be achieved by resizing the circuit and 'max delay' is the circuit delay with all gates having minimum size. The difference between the min delay and the max delay is denoted as $slack_{del}$. Also shown in the figure is the standard deviation (σ) for the delay variation due to inter- and intra-die variation, which is obtained from *statistical* timing analysis. Let us first assume that our delay constraint corresponds to the 50% of $slack_{del}$. Then by using the proposed algorithm, the circuit can be resized to meet, for example, 84.1% yield for the minimum increase in the area (Δ_{area}). Compared to this Δ_{area}, the increase in area in the case of delay constraint equal to 90% of $slack_{del}$ is much smaller as shown in the figure. Therefore, the effectiveness of our sizing algorithm in terms of minimum increase in area is dependent on target delay constraint.

The differences in Δ_{area} for the benchmark circuits are summarized in Table 1. The second column shows the $slack_{del}$ of different circuits in terms of σ (when the circuit delay is minimum, i.e., min delay) . It varies from 1.99 to 6.59 for different circuits. The third and fourth columns represent the percentage increase in area for 84.1% yield with delay constraints equal to 50% and 90% of $slack_{del}$, respectively. As expected, in the case of smaller target delay (50% of $slack_{del}$), the increase in area is larger for the same yield improvement under process variation.

We also compared our algorithm with the worst-case design methodology. For this purpose, we assume the target delay as the 90% of $slack_{del}$ for all benchmark circuits. The comparison results are shown in Table 2. In our experiment, circuits are first sized without considering process parameter variation. This corresponds to the sizing of the circuit for 'designed delay' shown in Figure 1. The area for the 'Nominal' design (without the variability taken into account, i.e., 50% yield) is shown in the third column of the table which represents the sum of transistor widths. The fourth column shows the area after resizing the circuits using our algorithm considering process variation. It can be seen that with small increase in area the yield can be improved to 84.1% (corresponding to σ) under process variation. The sixth column shows

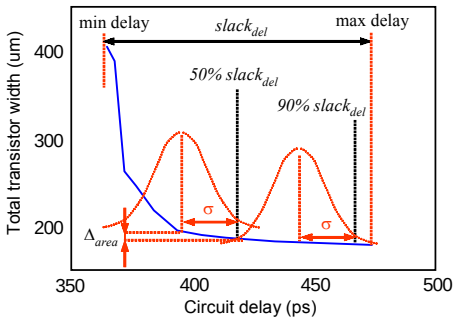


Figure 8. Circuit size vs. delay for c432

Table 1. The dependence of Δ_{area} on delay constraint

	$slack_{del}$ / σ	Δ_{area}(%) when target delay is	
		0.5 $slack_{del}$	0.9 $slack_{del}$
c432	4.95	1.57	0.65
c499	2.79	11.57	1.85
c1908	2.35	6.33	0.97
c2670	2.41	0.78	0.26
c3540	2.06	0.51	0.16
c6288	2.01	2.15	0.15
c74181	1.99	8.95	1.52
c74182	4.27	2.49	1.08
c74283	6.59	5.80	0.54
c74L85	4.43	3.83	2.67

the optimum area for 99.9% yield using our algorithm. We assume that the target delay is reduced by 3σ (99.9% yield) from the nominal design while resizing under process variation. In this case, while the yield improves, the increase in area is larger than the previous case. Hence, one can make a trade off between the yield and the area (power) budget. Furthermore, in some circuits (c1980, c6288, c74181), it is not possible to achieve 99.9% yield by resizing the gate. This is because for any circuit, there is a minimum delay that can be achieved by resizing the gate. The eighth column in the table (labeled as 'Worst design area') shows the circuit area for the worst-case design. For worst-case design, the circuit is sized assuming the worst process corner, i.e., all transistors will have worst parameter variations. For example, the transistor length is assumed to be '$L + \Delta L$', where ΔL represents the worst-case variation in L. Other process parameters are also considered in a similar way. It should be noted that while the yield under process variation is expected to improve in the worst-case design, the increase in area is large. It can be seen that the saving in area is as large as 19% (c74L85) using our proposed sizing algorithm compared to the worst-case design. Furthermore, in many cases (denoted as * in the table), it is impossible to size the circuit to achieve the desired delay using the worst-case design methodology. Columns 10 and 11 show the ratio of σ to mean delay with 84.1% yield for inter- and intra-die variation, respectively. It is observed while the ratio remains almost same for inter-die variation, the ratio for intra-die variation varies depending on the depth of the circuit. It should also be noted that the overall variation in delay is dominated by inter-die variation. The last column shows the runtime of our sizing algorithm for 84.1% yield. As explained in section 2, both inter- and intra-die process variations are considered through *statistical* timing analysis. It is observed in the experiments that the majority of the program runtime is attributed to the analysis of inter-die variation. In each iteration, the circuit was simulated for 10,000 different process corners to incorporate the inter-die variation as explained in section 2. The runtime can be reduced by decreasing the number of this simulation while maintaining the accuracy by using intelligent sampling techniques as explained in [16].

5. SUMMARY

We proposed an algorithm to size a circuit for statistical design considering both inter- and intra-die variation. This algorithm estimates the variation in circuit delay based on *statistical* timing analysis and sizes the circuit to achieve a desired yield with minimum increase in the area and power consumption. Experimental results on several benchmark circuits show that the savings in area (hence the power) can be as large as 19% using our algorithm than the worst-case design. It was also shown that it is not possible to achieve the desired delay in many circuits using the worst-case design methodology.

REFERENCES

[1] X. Tang, V. De, and J. D. Meindl, "Intrinsic MOSFET parameter fluctuations due to random dopant placement," *IEEE Trans. VLSI Systems*, pp. 369-376, 1997.

[2] C. Visweswariah, "Death, taxes and failing chips," *Proc. DAC*, pp. 343-347, 2003.

[3] J. P. fishburn and A. E. Dunlop, "TILOS: A posynomial programming approach to transistor sizing," *IEEE Trans. CAD*, pp. 326-328, 1985.

[4] S. Sapatnekar, V. B. Rao, P. M. Vaidya, and S. M. Kang, "An exact solution of the transistor sizing problem for CMOS circuits using convex optimization," *IEEE Trans. CAD*, pp. 1612-1634, 1993.

[5] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, McGraw-Hill, 3rd edition, 1991.

[6] Y. Taur and T. H. Ning, Fundamentals of Modern VLSI Devices, Cambridge University Press, 1998.

[7] H. F. Jyu, S. Malik, S. Devadas, and K. W. Keutzer, "Statistical timing analysis of combinational logic circuits," *IEEE Trans. VLSI Systems*, pp. 126-137, 1993.

[8] A. Agarwal, D. Blaauw, V. Zolotov, S. Sundareswaran, M. Zhao, K. Gala, and R. Panda, "Path-based statistical timing analysis considering inter- and intra-die correlations," *TAU*, 2002.

[9] A. Agarwal, D. Blaauw, V. Zolotov, S. Vrudhula, "Computation and refinement of statistical bounds on circuit delay," *Proc. DAC*, pp. 348-353, 2003

[10] E. T. A. F. Jacobs, M. R. C. M. Berkelaar, "Gate sizing using a statistical delay model," *Proc. DATE*, pp. 27-30, 2000.

[11] M. S. Bazaraa, H. D. Sherali, and C. M. Shetty, Nonlinear Programming: Theory and Algorithms, Wiley, 2nd edition, 1993.

[12] BPTM, http://www-device.eecs.berkeley.edu/ptm

[13] M. R. C. M. Berkelaar, "Statistical delay calculation, a linear time method," *TAU*, 1997.

[14] C. P. Chen, C.C.N.Chu, and D.F.Wong, "Fast and exact simultaneous gate and wire sizing by Lagrangian relaxation," *IEEE Trans. CAD*, pp.1014-1025, 1999.

[15] T. Sakurai and R. Newton, "Delay analysis of series-connected MOSFET circuits," *IEEE JSSC*, pp. 122-131, 1991.

[16] R. Y. Rubinstein, *Simulation and Monte Carlo method,* John Wiley and Sons, 1981.

Table 2. Experimental results of applying the sizing algorithm to ISCAS benchmark circuits

	No. of TR	Normal (50% yield) Area (um)	84.1% yield area (um)	% increase in area	99.9% yield area (um)	% increase in area	Worst design area (um)	% increase in area	$\sigma_{inter}/\mu_{delay}$ (%)	$\sigma_{intra}/\mu_{delay}$ (%)	Run time (sec)
c432	590	178.16	179.32	0.65	182.83	2.62	185.8	4.29	5.87	1.22	35
c499	1816	537.15	547.10	1.85	717.42	33.56	*	*	5.35	0.88	367
c1908	1582	473.83	478.41	0.97	*	*	*	*	5.29	0.78	316
c2670	2394	668.23	669.96	0.26	686.95	2.80	*	*	5.17	0.92	222
c3540	3638	1126.58	1128.43	0.16	1150.94	2.16	*	*	5.25	0.57	430
c6288	9472	2444.07	2447.84	0.15	*	*	*	*	5.28	0.15	2875
c74181	372	106.58	108.20	1.52	*	*	*	*	5.12	1.26	51
c74182	92	29.56	29.88	1.08	31.24	5.68	34.31	16.41	5.34	2.85	7
c74283	188	62.46	62.80	0.54	64.83	3.79	67.19	7.57	5.42	1.71	13
c74L85	148	39.02	40.06	2.67	45.51	16.63	52.89	35.55	5.39	2.66	11

(* sizing failed for the corresponding scheme)