A Statistical Approach to Area-Constrained Yield Enhancement for Pipelined Circuits under Parameter Variations*

Animesh Datta, Swarup Bhunia, Saibal Mukhopadhyay, and Kaushik Roy
School of Electrical and Computer Engineering, Purdue University, IN, USA
{*adatta, bhunias, sm, kaushik*}@ecn.purdue.edu

Abstract *- Under inter- and intra-die parameter variations, delay of a pipelined circuit follows a statistical distribution. Hence, a pipelined circuit suffers yield loss with respect to violation of target delay constraint unless an overly pessimistic worst-case design approach is followed. We propose a statistical approach for pipeline design to enhance yield with respect to a target delay under an area budget. Right choice of the number of pipeline stages to enhance yield under an area constraint is addressed using simple statistical yield models. Next, individual stages are designed for maximizing yield under area constraint for the stages. Once the independently optimized stages are combined to form a pipeline, we propose a final global optimization step to improve pipeline yield with no area overhead, based on a concept of area borrowing. Optimization results show that, the proposed statistical design approach for pipeline improves the overall yield up to 12% over conventional design for equal area.*

1. Introduction

Increasing inter-die and intra-die variations in the process parameters, such as channel length (L), width (W), oxide thickness (T_{ox}), threshold voltage etc., result in large variation in the delay of logic circuits [1]. Consequently, designing high-performance circuits with high yield (probability that the fabricated chip will meet a certain delay target) under parameter variations has emerged as a serious challenge in nano-meter scale designs [1, 5]. Pipelining data and control paths are popularly used in high-performance system design to improve throughput [3]. In a synchronous pipelined circuit, the throughput is determined by the slowest pipe segment [3]. However, under parameter variations, as the delay of a stage follows statistical distribution, the slowest stage is not readily identifiable.

Overall pipeline delay also follows a statistical distribution, which depends on the delay distributions of individual stages and the electrical/spatial correlation among them. For all practical purposes, delay distribution of a stage can be assumed to be Gaussian, and, thus, delay distribution for a pipelined circuit can also be estimated as a Gaussian random variable [1, 6, 10]. Since delay of a pipeline is statistical in nature, pipeline yield with respect to meeting a delay target depends on the nature of the delay distribution, which, for a Gaussian distribution is determined by the mean (μ) and standard deviation (σ) of the distribution.

Delay of a pipelined circuit determines its operating frequency and throughput. During the design phase of a pipeline, overall delay of the pipeline is changed by reducing the delay of the slowest stage. There are multiple design techniques to trade-off pipeline delay for power or die-area using logic synthesis, gate/wire sizing etc. Unless a worst-case design is chosen for a pipeline, which guarantees to satisfy the target delay at the worst process corner, a pipeline design is bound to suffer yield loss in terms of failure to meet a delay constraint. However, a worst-case design is overly pessimistic in terms of area/power requirement. Hence, a design methodology, which addresses yield optimization of the pipeline under statistical delay variation with minimum impact on area/power, is becoming mandatory.

Traditionally, the pipeline operating frequency has been enhanced by: a) increasing the number of pipeline stages, which, in essence, reduces the logic depth and hence, the delay of each stage; and b) balancing the delay of the pipe stages, so that the maximum stage delay is optimized [3]. However, it has been shown that if intra-die parameter variation is considered, reducing the logic depth increases the variability (defined as standard deviation/mean) [5]. A gate sizing technique to ensure yield under process variation circuits has been proposed in [2]. Statistical timing analysis in combinational circuits and latch-based pipeline designs under parameter variations are addressed in [1, 6, 7]. However, none of these works present a statistical design of pipelined circuit to enhance yield under a design constraint on area.

In this paper, we propose a statistical design framework for maximizing yield of a pipelined circuit under area budget. We have observed that pipeline yield depends on the number of pipeline stages; delay distribution of individual stages and spatial/electrical correlations among stage delays. It is also observed that irrespective of the correlation among stages, improving yield of a stage also improves the yield of the pipeline. We have proposed a hierarchical design flow for pipelined circuits consisting of three steps: 1) selection of appropriate number of pipeline stages (N); 2) optimization of individual stages for maximizing yield of a stage under constraint on stage area; and 3) a final optimization step on the complete pipeline after the stages are independently optimized.

Note that, unlike conventional pipeline design, we add an additional optimization step (step 3) in our design flow. This is based on the observation that even though individual stages are optimized for yield under stage area constraint, yield of some stages can be perturbed (at the expense of change in stage area), to improve the overall yield of pipeline. Let us take an example pipeline with overall yield Y consisting of stage 1 and 2, which are optimized to achieve pipe-stage yield of Y_1 and Y_2 for target area of A_1 and A_2, respectively. Now, let us assume that an increase in area targeted for stage 1 by ΔA_1 improves the pipeline yield by ΔY. Now, if we let stage 2 compensate for the area increase in stage 1, yield for stage 2 may decrease. However, if

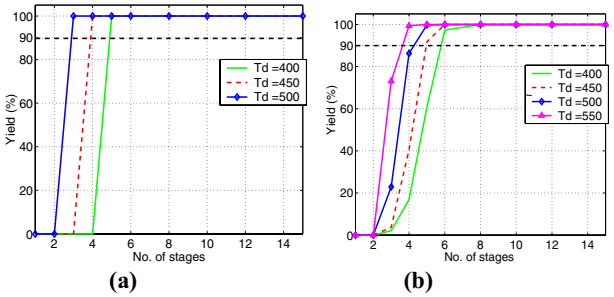


Figure 1: Plot of yield vs. number of stages (a) with only intra-die variation (b) with inter-die and intra-die variation for a 120-long inverter chain pipeline

* The work is sponsored in part by Marco Gigascale Systems Research Center (GSRC) and Semiconductor Research Corp. (grant no. 1078.001)

the decrease is less the ΔY, we have a net increase in pipeline yield for the same total area. We refer to this concept as **area borrowing**. To make the concept of area borrowing effective, we propose a heuristic for area allocation to the stages during step 3.

The rest of the paper is organized as follows; section 2 formulates the problem of yield enhancement of a pipelined circuit under an area constraint. In section 3 we present the pipeline design flow under statistical delay variation. Section 4 presents the yield improvement results on an example pipeline. Finally, section 5 concludes the paper.

2. Problem Formulation

To enhance the yield of a pipeline design under statistical delay variation, we need to consider its impact on the overall area (i.e. area of combinational logic + area of sequential elements). Thus, the pipeline yield enhancement problem can be formulated as:

$$\begin{aligned} &\text{Maximize } Yield = Y = f\left(N, \{\mu_i, \sigma_i; \forall i = 1,\ldots,N\}\right) \\ &\text{Subject to } \sum_{i=1}^{N}\left(A_{COMB-i} + A_{SEQ-i}\right) \le A_{TARGET} \end{aligned} \quad (1)$$

where, N is the total number of pipeline stages, μ_i and σ_i are the mean and the standard deviations of the delay of the ith stage, A_{COMB-i} is the and A_{SEQ-i} are the area of the combinational and the sequential stage logic, respectively, and A_{TARGET} is the maximum bound on the total area. The function 'f' represents the dependence of the yield (Y) on the total number of stages, and the delay distribution of the individual stages.

The overall delay of a pipeline is determined by the delay of the slowest pipeline stage. Hence, the overall pipeline delay (T_P) is given by:

$$T_P = \underset{i=1,\ldots,N}{Max}(SD_i) = Max\left(SD_1, SD_2, \ldots, SD_N\right) \quad (2)$$

where, SD_i represents the delay of the ith stage which is defined as Gaussian random variable ($SD_i \sim N(\mu_i, \sigma_i)$). The mean and the standard deviation of T_P can be estimated by following the method proposed in [7, 10].

Using (2), the yield of the pipeline design (i.e. probability of meeting a target delay T_d) is defined as:

$$Y = \Pr\{\max_{i=1,\ldots,N} SD_i < T_d\} = \Pr\{\bigcap_{i=1,\ldots,N}(SD_i < T_d)\} \quad (3)$$

The exact estimation of (3) is possible by assuming the stage delay (SD_i) to be *independent* Gaussian random variable, as:

$$Y = \Pr\{\bigcap_{i=1,\ldots,N}(SD_i < T_d)\} = \prod_{i=1}^{N}\Phi\left(\frac{T_d - \mu_i}{\sigma_i}\right) = \prod_{i=1}^{N} Y_i \quad (4)$$

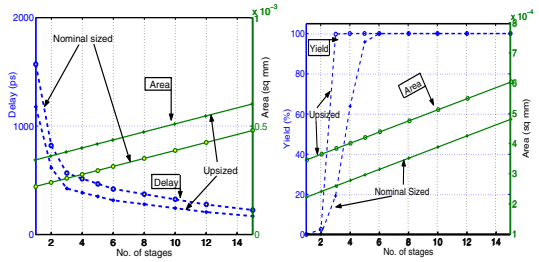
where, Φ represents the Cumulative Distribution Function (CDF) and Y_i is the yield of the ith stage. If the variables are correlated such a simplification is not possible. To estimate P_D considering correlated SD_is, we approximate the overall pipeline delay (T_P) as a Gaussian random variable (with μ_T and σ_T estimated using the method described in [10]). Using this assumption P_D is given by:

$$P_D = \Pr\{T_D \le T_{TARGET}\} = \Phi\left(\frac{T_{TARGET} - \mu_T}{\sigma_T}\right) \quad (5)$$

2.1 Impact of number of stages on yield and area

From (1), it can be observed that any change in the number of stages changes the pipeline yield with respect to a target delay, latency and total area. Below, we discuss these effects in details.

(1) *The effect of number of stages on the yield variation*:



(a) (b)

Figure 2: Plot of (a) design delay and total area (b) yield (for a target delay of 450ps) vs. number of stages

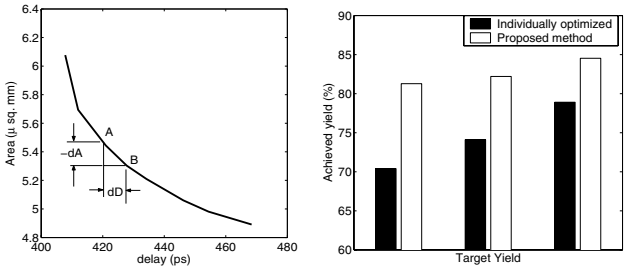
Increasing number of stages has a positive impact on the yield. The actual amount of effect depends on the logic depths and measure of inter and intra-die variations. For example, let us consider pipelining of a chain of 120 identical inverters. The number of stages is chosen in such a way that each stage has equal number of inverters. It was observed that, any N greater than or equal to 4 ensures the target yield of 90% at 450ps (Fig 1(a)) in the presence of only intra-die variation (σ_{vth_intra}= 30mV). However, under both inter and intra-die variation (σ_{vth_inter}= 30mV, σ_{vth_intra}= 30mV), delay analysis show that number of stages required to ensure the same yield (at the same target delay) is 5 or more (Fig. 1(b)). Hence, proper choice of the number of stage is necessary to enhance yield under process variation.

(2) *The effect of number of stages on the total area*:

The choice of N affects the total area in different ways depending on whether the design area is dominated by area of the combinational logic or sequential elements. When the total design area is dominated by the combinational logic, increasing N (which increases the number and area of sequential elements) helps to improve yield without much area penalty. If the area of the sequential elements is higher, increasing N may result in a large area overhead. Under this condition, if multiple N realizes the same yield, the lowest value of N helps to minimize the total area. Hence, a proper choice of N to enhance yield under an area constraint depends on the relative contribution of the combinational and sequential elements to the total design area.

2.2 Delay distribution of the individual stages

It can be observed from (3) and (4) that the yield strongly depends on the delay distribution parameters of the individual stages. In particular, increasing the mean delay of the individual stages increases the mean delay of the overall pipeline, thereby reducing the yield. Hence reduction of the mean and the standard deviation of the stage delays essentially increase the probability that each stage meets the delay target (this probability is used here as the stage yield). However, reducing the mean delay of the stages increases the stage area. To understand that let us re-consider the pipelined circuit of 120 inverters described in the previous sub-section. In this circuit, increasing the area of the inverters in each stage reduces the overall pipeline delay (thereby improving the yield) but increases the circuit area (Fig. 2(a)). It is interesting to note that, using the larger inverter size the target yield (i.e. 90% at target delay of 450ps) can be realized using a smaller number of stages (4 stages instead of 5 as shown in Fig. 2(b)). However, the overall circuit area increases (Fig. 2(b)). Hence, to enhance yield of the overall pipeline under an area constraint, each stage needs to be individually optimized to maximize "stage yield" while ensuring the stage area constraint.



(a) (b)

Figure 3: (a) Area vs. delay plots of a logic stage; (b) Effect of pipeline stage area allocation on design yield

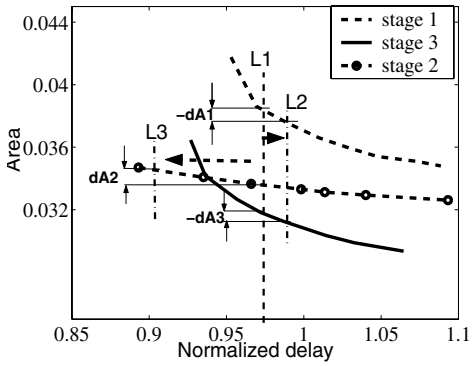


Figure 4: Effect of area allocation on pipeline design yield

2.3 Property of the "Max Function"

From (5) it can be understood that the delays of the individual stages do not completely determine the overall yield. The overall yield is determined by combining the individual stage yields and correlation among themselves using the "Max function" (2) [10].

Traditionally, the pipeline stages are designed for equal delay to maximize the throughput [5]. Such that considering independent stages a pipeline design represent the condition $Y_1 = Y_2 = ... = Y_N = Y^{1/N}$. However, from (4) it can be observed that a proper allocation of yield (say Y_i') among different stages such that $(Y_1'Y_2'\ldots Y_N') > (Y_1Y_2\ldots Y_N)$ can improve the overall yield. However, such area allocation has to consider its impact on the total area.

It should be noted that, as the target delay of a combinational logic increases to dD, the area required to realize the logic with that target delay reduces by dA (Fig. 3(a)). This is due to the fact that smaller sized logic gates can be used to realize a larger target delay. However the rate of change of area with delay (slope of area vs. delay curve $\partial A/\partial D$) varies over a range for different target delay. This observation plays an important role in enhancing the pipeline yield under constant area.

Let us now analyze the use of proper "stage area" allocation for maximizing the yield of pipeline under an area constraint. To show this effect, we have performed experiments with a 3-stage ALU-Decoder circuit pipeline structure. First, we optimized the combinational logic of each stage for minimum area (using independent stage delay model (4) for simplicity, in reality stage delays are correlated and corresponding results are presented in section 4) for a specific target pipeline yield (say $Y = 0.8$). For pipeline yield target $Y = 0.8$, yield target for each stage becomes $(0.80)^{1/3} = 0.9283$. In the next step, we have introduced proper imbalance among the three stages (by transistor sizing) in such a way that the total area remains constant but overall design yield improves. To understand the reason behind this yield improvement, let us consider the area vs. delay curves for each stage (Fig. 4). They are initially designed for equal yields and delay distribution parameters as indicated by line $L1$ in Fig. 4. This results in yield of Y_0 for each stage (pipeline yield = Y_0^3). The total area for this design is the sum of the stage areas ($A1+A2+A3$). Now, we allocate a lower yield to stages 1 and 3 by reducing the area of stage 1 and 3 (by $dA1$ and $dA3$) which increases their delays to line L2 in Fig. 4. This reduces yields of stages 1 and 3, to Y_1, Y_3 (say $Y_1 = 0.915$, $Y_2 = 0.92$, and both are less than Y_0). However, this extra area ($dA1 + dA3$) can be added to the stage 2, thereby reducing its delay to line $L3$. This improves yield of stage 2 (i.e. $Y_2 = 0.98 > Y_0$). In this case, $(Y_1 \times Y_2 \times Y_3 = 0.825) > (Y_0^3 = 0.8)$, and hence, the overall pipeline yield improves. For different target yields this trend has been observed in the 3 stage pipelined circuit as shown in Fig. 3(b). However, introducing excess imbalance in stage delays by area allocation, we might get diminishing returns when pipeline performance is governed by the mean delay (μ) of the slowest stage. Hence, it is necessary to appropriately apply area allocation among the stages. Such an area allocation is possible only if all the stages are considered together after individually and independently optimizing each pipeline stage. When correlations among the stage delays are considered similar trend is observed in our simulations results (section 4).

2.4 Complete optimization of pipeline for yield enhancement

Based on the above observations, we have developed a general design flow for enhancing yield under an area constraint (i.e. to solve the problem in (1)). Fig. 5 shows different steps of the design flow for yield enhancement for a target delay (T_d) and area (A_{TARGET}). As the number of pipeline stages (say, N) has a strong impact on the overall pipeline yield and area, in the first step we choose an optimum value of N that ensures a certain target yield while minimizing the total area. In the second step we optimize each stage individually for the target delay T_d and for the yield Y_i. In the final step we perform the total pipeline optimization by considering all the stages together to enhance the yield by proper area allocation for different stages. In the next section we present the detailed procedure for each step in Fig. 5.

3. Pipeline Design Flow under Statistical Delay Variation

In this section, we propose a statistical design flow for a pipelined

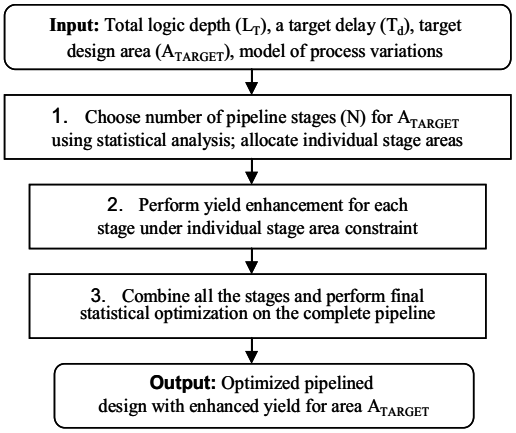


Figure 5: Complete pipeline design optimization methodology for yield enhancement

circuit for yield enhancement under area (or power) constraint. As described in section 2.4, we divide the complex problem of yield enhancement for pipeline design in three separate steps (Fig. 5). We propose statistical design approach for all three steps. For steps 2 and 3, we have used transistor size as the design parameter to vary. We have shown how existing gate-level sizing algorithms can fit into the proposed statistical pipeline design framework. However, the proposed framework is equally amenable for other optimization techniques (e.g. logic synthesis).

3.1 Choice of number of pipeline stages (*N*)

Number of pipeline stages (N) is typically specified by architectural constraints and performance (throughput) requirements of the design from system level analysis. However, if no specification of the number of stages is available from the system level we can use a simplified approach to obtain an initial estimate of the number of stages (N). Such an approximate estimation will be useful while pipelining one particular functional unit (e.g. multiplier). The N-selection problem can be formalized as:

Select N for total logic depth L_T,
Such that Y is maximized for T_d,
Subject to $\sum_{i=1}^{N}\left(A_{COMB-i}+A_{SEQ-i}\right)\le A_{TARGET}$ (6)

As discussed in section 2.1, the proper choice of N is determined by the yield requirement and the target area. A higher value of N increases the design yield at the cost of higher area for the sequential elements. On the other hand, increasing the logic area helps to increase the yield by lowering the delay of the individual stages. Hence, as discussed in the section 2.1, the impact of the choice of N on the total area, is determined by the relative magnitude of the logic area and flip-flop area. To understand this, let us consider a pipeline design of 120-long chain of inverters. Using nominal size for the inverters 90% design yield is ensured for $N \ge 5$. Upsizing the inverters (by a factor ~ 1.6, which increases the total combinational logic area by the same factor) reduces both the mean and the variability of the stage delays and helps to realize 90% design yield with $N = 4$ (smaller latch area) (Fig. 2(b)). However, it can be observed that the total area with $N = 4$ is lower than that with $N = 5$ in this design (Fig. 2(b)). Hence, in this example $N = 5$ is a better choice. Thus, under an area constraint the proper choice of N is required to maximize the yield. We will now present a simple method to estimate the optimum number of stages required to maximize the yield.

First, we estimate the maximum (N_{max}) and minimum (N_{min}) bounds of N considering combinational logic is designed with minimum and maximum sized logic gates, respectively (assuming constant size for the sequential elements). An exact estimate of the optimum value of N ($N_{min} < N < N_{max}$) can be obtained by performing detail timing analysis on the complete design. However, such an approach is very difficult at the initial design phase due to lack of exact circuit knowledge and the large computation time. Hence, we propose to use a simple approach for the selection of N. We first analyze the circuit to estimate the total number of logic depth (say, L_T) in the critical path. We also assume that, due to pipelining the total logic depth of the critical path gets equally divided among the stages (i.e. all stage has equal logic depth, say L). To estimate the delay of a stage, we assume an equivalent inverter chain model. Using this model, the stage delay distribution parameter (mean (μ_{stage}) and standard deviation (σ_{stage}) are computed as:

Table I: Procedure to select the number of pipeline stages to maximize yield under an area constraint

Input: Area constraint (A_{TARGET}), individual flip-flop area (A_{SEQ}), and total logic depth (L_T)
Output: Number of stages (N) maximizing yield for A_{TARGET}

1. Compute N_{min} using: $A_{TARGET} = A_{COMB-\max} + A_{SEQ} * N_{\min}$
2. Compute N_{max} using: $A_{TARGET} = A_{COMB-\min} + A_{SEQ} * N_{\max}$

/*$A_{COMB-max}$, $A_{COMB-min}$ are stage areas with max. and min. sized gates */

3. **for** $n_{stage} = N_{min}$ to N_{max} **do**
   /* assuming equal depth (L) and area (A_{COMB}) for all stages */
4. $A_{COMB} = \left[A_{TARGET} - A_{SEQ} * n_{stage}\right] / n_{stage}$ and $L = L_T / n_{stage}$
5. Compute (μ_{stage} ,σ_{stage}) from A_{COMB} and L
6. Compute pipeline yield for number of stage = n_{stage} as:

$$Y_{nstage} = \prod_{over\ all\ stages} \Phi\left(\frac{T_{TARTGET} - \mu_{stage}}{\sigma_{stage}}\right)$$

7. **end for**
8. Select N corresponding to the max. value of Y_{nstage}.
9. If multiple values of n_{stage} realizes the max. yield, select the highest value of n_{stage}.

$$\mu_{stage} = L\mu_{inv} \text{ and } \sigma_{stage} = \sqrt{L}\sigma_{inv} \quad (7)$$

where, μ_{inv} and σ_{inv} are the mean and the standard deviation of the delay of an inverter, respectively. Using the above assumptions, for a particular choice of N we estimate the maximum area that can be used by each logic stage (A_{COMB}) using the total target area (A_{TARGET}) and the area of a sequential element (A_{SEQ}) (line 4, Table I). The equivalent inverter chain model of the stages is used to evaluate the μ_{stage} and σ_{stage} as follows:

1. Area of an inverter = $A_{inv} = A_{COMB}/L$
2. Compute μ_{inv} and σ_{inv} for area A_{inv} using circuit simulation
3. Compute μ_{stage} and σ_{stage}

(8)

Using μ_{stage} and σ_{stage} computed from (8) we estimate the pipeline yield assuming all the stages to be uncorrelated (using (4)). Finally we select the value of N that gives maximum design yield. Table I describes the procedure for N selection. It should be noted that, increasing the number of stages (under no area constraint) has a positive impact on yield, as described in section 2.1. Hence, if N selection process as described above, produces multiple values of N (all vales correspond to the same total area, A_{TARGET}) that realizes the same maximum yield, we propose to use the highest value. We have applied the above procedure to the

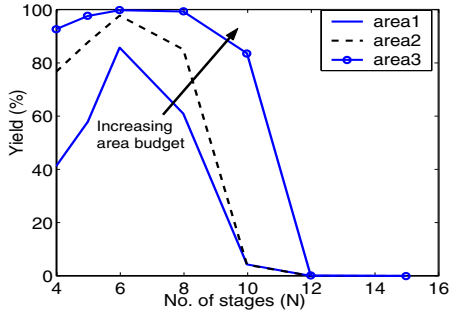


Figure 6: Effect of increasing design area target on N

example 120 inverter-chain pipeline. It can be observed that for smaller target area design yield is maximized for N = 6. Increasing the target area increases the scope of increasing number of stages as N = 8 is the best choice for a higher target area (Fig. 6).

3.2 Optimization of individual stages

Once the number of stages (N) is determined, individual stages of the pipeline can be optimized for the given design objective. In our case, each pipeline stage needs to be optimized for yield under an area constraint for the stage. The problem can formalized as below:

Maximize	Y_i for the i^{th} stage	
Subject to	$A_{COMB-i} + A_{SEQ-i} \leq A_{TARGET-i}$	(9)

The solution of the above problem can be obtained in two steps: a) allocation of target area to each pipeline stage, and b) optimizing yield of each stage for the target area allocated to that stage. We propose a simple heuristic to allocate target area to the individual stages based on the complexity of the stage logic. We assume that the complexity of the logic is linearly dependent on the number of logic gates. In that case the target area for the i^{th} stage can be determined by:

$$A_{COMB\_i} \leq (A_{TARGET} / L_T) * L_i - A_{SEQ-i} \quad (10)$$

where, A_{TARGET} is the pipeline target area, L_T is total logic depth and L_i is logic depth of the i^{th} stage.

To perform the second step, we use a gate-level transistor sizing algorithm as proposed in [4] using Lagrangian Relaxation (LR) based sub-gradient optimization. In [4], a solution for convex gate-level sizing problem is proposed to minimize maximum delay under an area constraint. We use the LR-based algorithm for maximizing yield of a stage for a given target area. We assume that for a considerable variation in mean of a stage delay distribution (μ_i) with transistor sizing, the standard deviation of stage delay (σ_i) varies in the same direction. We have observed this with Monte-Carlo simulation on several logic stages. Under this assumption, the sizing algorithm for minimizing maximum mean delay of a circuit can be used to maximize yield under statistical delay distribution.

3.3 Global optimization of pipeline yield under area constraint

In a conventional pipeline design flow, individual stages are designed and optimized independently of others for a given design objective before they are combined together to form a pipeline. We propose incorporating a final design step on the complete pipeline to improve yield while maintaining area (or power). Under statistical design approach, a final optimization of the complete pipeline design can improve the overall design yield due to the following reason:

- Although individual stages are optimized for the best possible yield under given stage area constraint, overall yield for the complete pipeline may have opportunity for improvement without area overhead. As mentioned before in section 2, we can exploit the nature of "Max function" to achieve this.

The optimization problem for maximizing yield of a pipeline under a given area budget can be formulated as:

Maximize $Y = \phi(\frac{T_{TARGET} - \mu_T}{\sigma_T})$ /*Y is pipeline yield, μ_T is mean, σ_T is STD of pipeline delay */

Subject to $\sum_{i=1}^{N} Area\ (Stage_i)$ = A_{TARGET} (Constant).

$M_i \leq x_i \leq U_i,\ i = 1,....,N.$ /*x_i is the size factor for a logic gate. M_i and U_i are the min. and max. size factors of a gate*/

We have developed an algorithm to solve the above problem efficiently. Table II presents this iso-area yield improvement algorithm. The algorithm employs the principle of divide-and-conquer, where we size one pair of stages at a time in such a way that the combined yield for the pair of stages is improved while the total design area is unchanged. Moreover, statistical timing analysis (based on statistical static timing analysis or SSTA as proposed in [9]) is performed over the complete pipeline, although the sizing is done for only one stage. It helps to make the algorithm computationally efficient, since we avoid application of the sizing routine on all the stages simultaneously.

The principal idea is to optimally trade-off yield among the pipeline stages under a constant area. We make use of area vs. delay trends (Fig 3(a)) of each stage to determine which stage is appropriate for improving yield (with increase in area) and which stage is appropriate for compensating the area (with certain decrease in yield) We refer this concept as *area borrowing* to imply that overall yield of the pipeline can be improved by selectively increasing area of some stages, while decreasing the same amount of area from other stages.

First, we determine the position of each stage in their area vs. delay curve, which essentially indicates how aggressively the stage is optimized for yield. We rank the stages in the ascending order of their slopes ($R_i = \left|\partial A / \partial D\right|_i$) of the area versus delay curves (step 5, Table II). Next, we create $N/2$ pair of stages by grouping each i^{th} stage with the $(N\text{-}i)^{th}$ stage in the sorted list of stages (step 6 - 8). For example, if a 5-stage pipeline has area vs. delay slopes as $R_3 > R_5 > R_2 > R_1 > R_4$, then according to the above rule we choose (R_3, R_4) and (R_5, R_1) as two pair of stages. Now for each pair, we enhance the yield of the stage with smaller slope (R_i) at the expense of certain area increase (ΔA_i) using transistor sizing algorithm described in section 3.2 (step 11). The area overhead (ΔA_i) thus incurred is compensated by the other element in the pair (with higher slope) at the expense of a

Table II: Algorithm for yield enhancement of a pipelined circuit under an area constraint

Input: A pipelined circuit, statistical delay parameters for N logic stages, target delay (T_d), target area (A_{TARGET})
Output: Pipelined design with enhanced yield

1. **for** each stage i = 1 to N **do**
2. Perform statistical timing analysis
3. Compute the area vs. delay curve
4. **end for**
5. Sort the stages (S_i) in ascending order of their R_i values /* select the pairs of stages for yield optimization */
6. **for** each stage i = 1 to $N/2$ **do**
7. $P_i = \{S_i, S_{N-i}\}$ /* create stage pairs */
8. **end for**
9. **for** each stage pair P_i from i = 1 to $N/2$ **do**
10. Determine ΔA_i for the stage pair
11. Resize stage (S_i) with lower R_i to improve yield at the expense of ΔA_i
12. Resize stage (S_{N-i}) with higher *slope* to compensate ΔA_i
13. **end for**

decrease in overall pipeline yield (step 12). However, due to the difference in slopes between the two stages in a pair, the process always increases the combined yield of the stage pairs. In the process, pipeline stages move closer in area vs. delay curve i.e. tend to be balanced with respect to their R_i values.

It is worth noting that for a particular stage pair ($\{S_i, S_{N-i}\}$), optimal possible yield that can be obtained by the area borrowing concept depends on the selection of exact area ΔA_i to trade between them (step 10). We use a simple iterative solution to obtain the best choice of ΔA_i for a stage pair by incrementally changing ΔA_i at successive steps of a fixed step size.

The LR based sizing algorithm proposed in [4] has a computational complexity of $O(n^2)$ where n is the number of logic gates to size. For m pipeline stages each having n gates the simultaneous sizing approach runs with a complexity of $O(m^2n^2)$ (with space complexity of $O(mn)$). The proposed algorithm improves the complexity to $O(mn^2)$ (with space complexity of $O(n)$). The complete pipeline design optimization algorithm proposed here is significantly faster and takes much less storage compared to the case where all the stages are sized simultaneously.

4. Results

In this section, we present yield improvement results with the proposed design methodology for two example pipelines. The result of yield improvements obtained by applying the proposed yield optimization algorithm to a 3-stage ALU-Decoder pipeline is given in Fig. 3(b) (section 2.3).We first individually optimize the stages for target delay of 175ps and 180ps. The stages are then combined together to realize the complete pipeline resulting in the yield of 70.5% (for 175ps) and 79% (for 180ps). Application of the complete pipeline optimization algorithm improves the yield by 15% (from 70.5%) for the target delay 175ps and 7.1% (from 79%) for the target delay of 180ps over the individually optimized design.

We have also performed experiments with a 4-stage pipelined circuit (designed with ISCAS85 benchmark circuits c499, c880, c1908, c2670 as the stage logic, and edge-triggered D flip-flops as the sequential elements). Initially, we assume that the stage delays are independent Gaussian RV and use (4) to compute pipeline design yield. Results of the proposed optimization are shown in Table III and IV. Here 1st column represents the target delay for the pipeline design and 2nd column presents the yield for a 4-stage balanced pipeline where individual stages are already optimized for equal yield (Y) under the given area budget. Column 3 presents yield obtained from our proposed methodology. The 4th column shows the yield improvements. The results show up to 16% yield improvement (from the initial yield of 73%). The scope of improvement gradually reduces with higher target yield. We obtain 2% improvement in the case when the initial yield was 95% (Table III)

Table-III Yield improvement for independent stage delays

T_d (ps)	$Y_{balanced}$ (%)	$Y_{optimized}$ (%)	$Y_{increase}$ (%)
420	73.13	85.10	16.37
430	74.61	86.12	15.43
440	80.54	90.75	12.66
450	83.70	92.59	10.62
460	84.98	93.45	9.96
470	86.72	92.89	7.15
480	88.28	94.23	6.6
490	89.89	94.39	5.01
500	93.22	95.69	2.65
510	95.14	97.09	2.05

Table-IV Yield improvement for correlated stage delays

T_d (ps)	$Y_{balanced}$ (%)	$Y_{optimized}$ (%)	$Y_{increase}$ (%)
420	78.24	87.79	12.21
430	81.06	90.03	11.05
440	84.63	91.78	8.54
450	87.52	93.62	6.84
460	89.43	94.27	5.92
470	90.45	94.72	4.51
480	92.36	95.8	3.42
490	92.89	94.77	2.02
500	95.12	96.31	1.50
510	96.53	97.32	0.82

However, when **correlation** among the stage delays are considered, for the same stage delays the overall pipeline yield increases but by a lesser amount than the independent case. This reduces the yield improvement by small amount. We obtain 12% yield improvement from the initial yield of 78%. This yield improvement gradually reduces to 1% for initial yield of 96.5% (Table IV).

5. Conclusions

We have proposed a statistical approach for pipeline design under parameter variations for maximizing yield with respect to a target area. The proposed hierarchical approach can be easily extended for other optimization objective like power. A statistical design methodology is proposed using gate sizing algorithm to enhance yield of individual stages. Experimental results on example pipeline shows that significant improvement in yield can be achieved with the proposed statistical approach of pipeline design with constraint on die-area.

REFERENCES

[1] K. A. Bowman et al., "Impact of Die-to-Die and Within-Die Parameter Fluctuations on the Maximum Clock Frequency Distribution for Gigascale Integration", *JSSC'02*, pp. 183-190.

[2] S. Choi et al., "Novel Sizing Algorithm for Yield Improvement under Process Variation in Nanometer Technology", *DAC 2004*, pp. 454-459.

[3] J. L. Hennessy et al., "Computer Architecture: A Quantitative Approach", *Morgan Kaufmann*, May 2002.

[4] C. Chen et al., "Fast and Exact Simultaneous Gate and Wire Sizing by Lagrangian Relaxation", *IEEE TCAD'99*, Vol. 18, No. 7, 1999, pp. 1014-1025.

[5] S. Borkar et al., "Parameter Variations and Impact on Circuits and Microarchitecture", *DAC 2003*, pp. 338-342.

[6] H. Mahmoodi et al., "Estimation of Delay Variations Due to Random-dopant Fluctuations in Nano-Scaled CMOS circuits", *CICC* 2004, pp. 17-20.

[7] S. G. Duvall, "Statistical circuit modeling and optimization", *IWSM,* June 2000, pp. 56-63.

[8] C. E. Clark, "The Greatest of a Finite Set of Random Variables," *Operation Research*, vol. 9, 1961, pp. 85-91.

[9] H. Chang et al., "Statistical timing analysis considering spatial correlations using a single pert-like traversal", *ICCAD* 2003, pp. 621-625.

[10] A. Datta et al., "Statistical Modeling of Pipeline Delay and Design of Pipeline under Process variation to Enhance Yield in sub-100nm Technology", *DATE* 2005, pp. 926 - 931.