

Statistical Rare Event Analysis Using Smart Sampling and Parameter Guidance

Yue Zhao*, Hosoon Shin*, Haibao Chen[†], Sheldon X.-D. Tan*, Guoyong Shi[†], and Xin Li[‡]

*Department of Electrical and Computer Engineering, University of California, Riverside, CA 92521

[†]School of Microelectronics, Shanghai Jiao Tong University, Shanghai, 200240, China

[‡]Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA 15213

Abstract—In this paper, we propose a new efficient statistical method for failure probability estimation of analog circuits with rare failure events, which is a time-consuming process using the existing Monte Carlo method. On top of this, the new method can also provide the estimation of parameter regions to achieve targeted performance to facilitate the design process, which is missing in the traditional fast statistical methods such as the statistical blockade based method. The new method employs two new techniques to speed up the analysis. First, to reduce the large number of samples for rare event analysis, the new approach employs a smart sample selection scheme, which can consider the effectiveness of samples and well-coverage for the parameter space. As a result, it can reduce an additional simulation costs by pruning less effective samples while keeping the accuracy of failure estimation. Second, the new approach identifies the failure regions in terms of parameters to provide a good design guideline for designers and optimization tools. This is enabled by applying the variance based feature selection to find the dominant parameters. A quasi-random sampling with dominant parameters is then applied to determine in-spec boundaries of those parameters. In addition, we also provide the complete formula for the probability determinations of failure regions in the iterative failure region searching framework. We demonstrate the advantage of our proposed method using two test benches: 6T-SRAM reading failure diagnosis with 27 process parameters, charge pump operation failure diagnosis in a PLL circuit with 81 process parameters. Experimental results show that the new method can be 4X more accurate than the recently proposed REscope method. Furthermore, the new method reduces the simulation cost by 2X than the recursive statistical blockade (RSB) method with same accuracy level. Our approach also provides the precise guidance of diverse parameters with 1.21% estimation error.

1. Introduction

As the CMOS technology scaling continues, the performance uncertainties related to the process variation have become a major concern for IC development [1]. Many IC components such as SRAM bit-cells need to be tremendously robust as they are duplicated in millions [2]. Such modules require accurate statistical failure analysis in rare event region. However, the traditional Monte Carlo (MC) based statistical analysis method faces a challenge as it may require a million times of simulations [3]. Various statistical techniques have been developed to overcome this problem in the literature [3]–[11].

One way is by means of importance sampling (IS) [4], which consists of two steps. First, it shifts the mean value of the initial performance distribution and place it on interested failure region. The standard deviation based on the shifted mean is recalculated by considering samples only placed on the failure region. The new probability density function (PDF) is generated based on updated mean and standard deviation so that more samples in the failure region can be drawn. Work in [11] applied the mixture of IS for cross-validation of

multiple failure regions due to disjointed process parameters. However, these approaches can only estimate single performance metric. Multiple important samplings are required to estimate more metrics. Also, it is difficult to calculate the failure probability for a generated distribution by IS.

The statistical blockade (SB) is another effective approach for improving the performance of MC [3]. The idea of this approach is using a threshold bound to separate an interested failure region from the whole distribution so that it can *block* some unnecessary sampling and simulation for efficiency improvement. This method builds a supervised learning model with the threshold bound and initial simulation data, which is named as “classifier”, to recognize failure samples. Thus, later samples that tend to be placed in the failure region can be captured without simulation. This approach was improved by using the recursive statistical blockade (RSB) scheme to locate the rare event failure region in an iterative way [5]. This method can improve the accuracy of the classifier iteratively by increasing the number of samples in the failure region of interest. However, this method can incur the significant extra cost as it draws more samples for the simulation. Recently, Wu *et al.* [7] applied a nonlinear SVM classifier to model nonlinear and multiple disjoint failure regions of circuits. This method used the generalized Pareto distribution (GPD) fitting for the tail of distribution to model failure probability in each iteration. However, this method cannot investigate further failure region without rerunning whole algorithm. The reason is that the pruned parameters depending on the initial samples cannot remain as important ones in the failure region, which keeps changing. Also, the previous approaches cannot provide the design guideline in terms of the design parameter boundaries to explicitly avoid performance failure, which are important for design and optimization processes.

In this paper, we propose a new statistical failure region diagnosis method. The new method is based on recursive statistical blockade [5] to gradually locate the failure regions. However, the proposed method consists of two new techniques. First, it introduces the smart sample selection scheme, which can consider an effectiveness of samples and well-coverage for parameters space so as to further reduce the simulation cost. The selection process can reduce number of samples significantly while it still keeps sufficient representative samples for accurate failure probability analysis. Second, the new approach can generate safe boundaries of parameters to satisfy the design specification. The new method first applies variance based feature selection to find the dominant parameters. A quasi-random sampling with dominant parameters is then used to quickly determine proper boundaries of those parameters. In addition, we also provide complete formulas to determine the probabilities of failure regions in the iterative failure region searching framework.

The presented method has been validated on 6T-SRAM single-bit cell and circuit pump circuit in a phase-locked loop

This work is supported in part by NSF grant under No. CCF-1116882 and in part by NSF grant under No. OISE-1130402.

(PLL) with each 27 and 81 dimension of process parameters. Experimental results show that the new method is up to 159x faster than traditional Monte Carlo method with only 10.78% error. We further show that the new method can be 4X more accurate than the recently proposed REscape method. Furthermore, the new method reduces the simulation cost by 2x than the recursive statistical blockage (RSB) method with same accuracy level. Our approach also provides the precise guidance of diverse parameters with 1.21% estimation error.

The reminder of this paper is organized as follows. We review some major techniques of the blockade-based method for failure analysis in section 2. In Section 3, we explain the proposed iterative diagnosis algorithm with an overall flow and detailed descriptions of the smart sampling and parameters guidance techniques. We also introduce the mathematical framework to calculate the probability of iteratively updated failure region. Section 4 shows the experimental results for verifying the accuracy and efficiency of the proposed method. Section 5 provides conclusions of this paper.

2. Background

We first briefly review the concept of the statistical blockade (SB) approach for fast estimation of properties of rare event region. A general framework of SB is shown in Fig.1. This method starts with drawing initial samples with uniform or normal distribution to capture a crude shape of performance distribution by circuit simulation. The classifier can be built by training with initial simulation data. Once we obtain the classifier, samples that tend to fall into the failure criteria can be identified without actual simulation. With these filtered samples, SB calculates the probability of failure region by fitting samples in the proper distribution model. Thus, “classification” and “failure probability calculation” are both key steps in the SB method. The rest of this section describes these two steps.

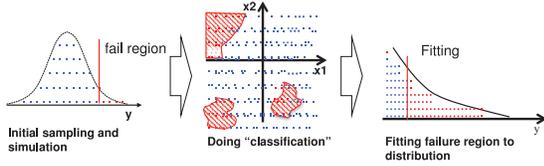


Fig. 1: General flow of statistical blockade approach

A. Classification

The classification filters samples as likely-to-fail samples for the circuit simulation. Building a classifier needs a training step with initial samples to render real shapes of the failure region. The classifier can then reduce the number of samples for real simulation because it can estimate mathematically whether a sample makes failure or not. However, the classifier is not capable of fully replacing the simulator due to its accuracy. So, a marginal filtering approach is used to improve the accuracy of classification [3], [5]–[7]. Some methods use relaxed threshold bounds instead of a real failure criterion to capture more samples to minimize the classification error. Meanwhile, it is not sufficient to use the simple and linear classifier due to the nonlinearity of the failure region [3], [5], [6]. The Gaussian radial basis function kernel (GRBF) is the suitable supervised learning model for nonlinear classification [7], [12]. GRBF can make curved boundaries to recognize patterns scattered with nonlinear and disjointed manner based on radial distances between samples. An example of GRBF classification with two

input parameters is shown in Fig. 2. The classification goal is to learn the shape of two distinct space region. Fig.2(a) is given as a training sample. Dots in “Pass” regions and failure regions will be marked as red and blue, respectively. Fig.2(b) is the classification result of another test sample done by GRBF classifier. GRBF will color every points, which are originally uncolored, based on their location and radial distances. Even though solution spaces are neither uniformly nor relevantly formed, GRBF can still recognize accurate boundaries of two different spaces.

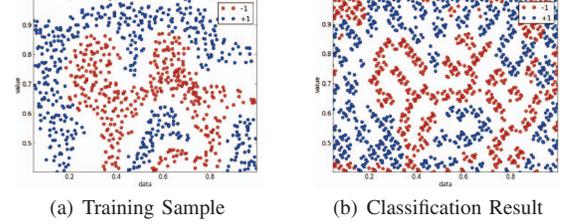


Fig. 2: The accuracy of classification by GRBF

B. Failure probability calculation

The failure samples should be fitted to a particular distribution form in order to estimate the probability of the failure region.

Suppose simulation results for performance metric Y can be fitted to a certain distribution with PDF $f_Y(y, \mu, \sigma)$ where parameters μ and σ are the mean and standard deviation of this distribution. We define $F_Y(y)$ as the cumulative density function (CDF) of the performance metric Y . If we know the threshold value t which separates a tail region from the whole distribution $f(y)$, the conditional cumulative density function (CDF) of this region can be written as follows:

$$F_t(y) = P(Y \leq y | Y \geq t) = \frac{F_Y(y) - F_Y(t)}{1 - F_Y(t)} \quad (1)$$

where $F_t(y)$ means the failure probability decided by y . Once we have a suitable fitting model for CDF of the failure region with a failure bound y , the failure probability with given values can be calculated as:

$$P(Y \geq y) = [1 - F_Y(t)] \cdot [1 - F_t(y)] \quad (2)$$

In the several generalized extreme value distributions, GPD is one of the most accurate models to describe tail distribution corresponding to failure region [13]. With the location parameter μ , the scale parameter σ and the shape parameter ξ , CDF of the failure region can be formulated by GPD fitting.

$$F_t(y) = G_{(\xi, \mu, \sigma)}(y) = \begin{cases} 1 - \left(1 + \frac{\xi(y - \mu)}{\sigma}\right)^{-1/\xi} & \text{for } \xi \neq 0 \\ 1 - e^{-\frac{-(y - \mu)}{\sigma}} & \text{for } \xi = 0 \end{cases} \quad (3)$$

The location parameter μ means a starting point of GPD and it is corresponded to the threshold t of the tail distribution. Consequently, the failure probability with given threshold t and failure bound y can be computed as follows.

$$P(Y \geq y) = [1 - F_Y(t)] \cdot [1 - G_{(\xi, t, \sigma)}(y)] \quad (4)$$

To approximate the rest of parameters for GPD fitting, we use the maximum likelihood estimation [14].

3. Proposed Failure Region Diagnosis Method

The overall flow of the proposed method, which is called "IFRD" (Iterative Failure Region Diagnosis), is illustrated in Fig.3. Our algorithm starts with given data, such as process variations and some parameters for failure region determination. The failure criteria t_c denotes the reference value of failure and the percentile bound p to calculate threshold in each i_{th} iteration. The first step is to perform initial MC sampling and simulation to capture overall circuit performance metrics. After this, the relaxed threshold t_i can be obtained to separate a failure region from main PDF and the probability of this region is $P(Y > t_i) = p$. A GRBF nonlinear classifier is built with the simulation result of the initial samples to perform more accurate sample filtering. With the simulation result and the classifier, the new method determines the in-spec conditions of process parameters to achieve targeted yield in i_{th} iteration. At the same time, the algorithm generates $m^i * n$ MC samples, which will be filtered by the classifier C_i to likely-to-fail samples based on t_i . Then, the smart sample selection can be employed to further reduce the number of samples for actual simulation. After the simulation, failure probability $P(Y > t_i)$ is updated by GPD fitting. Our approach iterates the above whole procedure with updated threshold bound t_i by percentile bound p , and increasing number of MC samples to calculate the failure probability $P(Y > t_i)$. The algorithm terminates when the threshold bound meets the given failure criterion t_c .

The rest of this section explains in detail the two major contributions of the proposed method: (1) Smart sample selection (2) In-spec guidance of process parameters. We also present the iterative formulation of failure probabilities.

A. Smart sample selection scheme

The simulation cost is a major bottleneck in the statistical analysis of the circuit. The proposed iterative failure diagnosis method can lead to an extra simulation cost in each iteration. To mitigate this problem, we propose the smart sampling scheme, which significantly reduce the number of samples required.

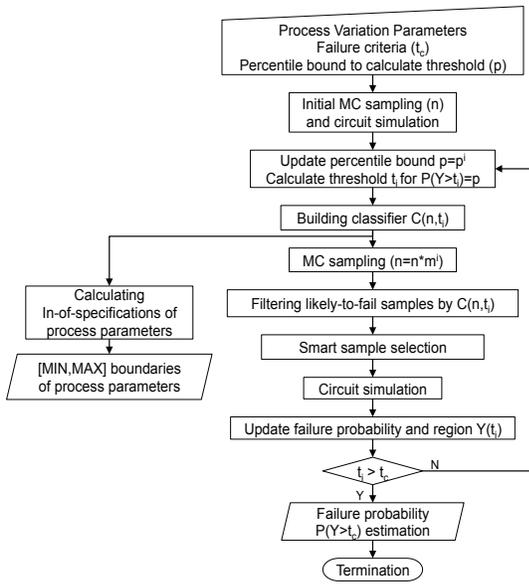


Fig. 3: Proposed iterative failure region diagnosis flow

Two factors should be considered in the sample selection process. The effectiveness of the sample group is the first factor. Each sample consists of the combination of process parameters, which affect differently on simulation results. Therefore, the sensitivity of each parameter should be considered in the sample selection. We first calculate correlation coefficients between process parameters and simulation results for the sensitivity analysis as follows:

$$x_n \in R^m, y_n \in R$$

$$\rho_{x_n, y_n} = \frac{cov(x_n, y_n)}{\sigma_{x_n} \sigma_{y_n}} \in R^m \quad (5)$$

where x_n, y_n are n process parameter vectors with m dimension and n simulation results, respectively.

The second factor is the coverage ratio of parameters search space by selected samples. The diversity of samples can be calculated by Euclidean distances with the reference sample, which is the median from simulation results. Here we highlight two sets of samples which are of our interest. One set includes the samples around the median. Since the median is located on the highest probability region in the distribution of simulation results, samples around it will do most contribution to the knowledge of the body part of the distribution. Another set includes samples found in the boundary region of the search space as these samples represent the maximum and minimum conditions of parameters that help constructing the right or left tail failure region from the resulting distribution.

Thus, the proposed sampling method calculates two distance factors to cover both central and boundary regions of the search space as follows:

$$\tilde{y} = \text{median}(y_n)$$

$$x_{ref} = \{x | \tilde{y} = f(x), x \in R^m\}$$

$$D_{central}(x) = \frac{1}{\left| \frac{x - x_{ref}}{\max(x_n) - \min(x_n)} \right|} \in R^m \quad (6)$$

$$D_{boundary}(x) = \left| \frac{x - x_{ref}}{\max(x_n) - \min(x_n)} \right| \in R^m$$

where $\max(x_n) - \min(x_n)$ is the normalization term. In (6), $D_{central}$ increases ($D_{boundary}$ decreases) which implies the sample is more close to the reference sample.

By calculating the inner product of correlation coefficient in (5) and distance factors in (6), we obtain the samples selection weight with respect to central region $W_{central}(x)$ or boundary region $W_{boundary}(x)$ as follows:

$$W_{central}(x) = \rho_{x_n, y_n}^T \cdot D_{central}(x) \quad (7)$$

$$W_{boundary}(x) = \rho_{x_n, y_n}^T \cdot D_{boundary}(x)$$

According to the selection ratio r , which determines the number of selected samples,

the final set with nr samples $E(n, r)$ can be chosen as following manner:

$$E(n, r) = S\left(\frac{nr}{2}, W_{central}(x_n)\right) \cup S\left(\frac{nr}{2}, W_{boundary}(x_n)\right) \quad (8)$$

here $S(n, W(x_n))$ is the set consists of samples that with first n element of $W(x_n)$ in descent order.

B. Parameters guidance for performance targeting

In order to improve the yield of a circuit, designers require the good knowledge of ranges of process parameters with regards to the circuit performance specification. However,

simulating all possible combinations of parameters is impossible due to exponential possibilities with a large number of parameters.

The proposed method ranks priorities of process parameters based on their variances. Since the parameters with huge variance mainly lead to spread samples in search spaces, these parameters must be handled to avoid certain failure regions. In Fig. 3, the variances of parameters can be calculated from simulation data of updated failure region in our iterative framework. Given n samples with m process parameters, the variance of samples can be written mathematically as follows:

$$Var(x_m) = \frac{1}{n} \sum_{i=1}^n (x_{m(i)} - \mu_m)^2 \quad (9)$$

where x_m is a single sample and μ_m is the mean of n samples. Next, our method redraws samples with only considering the distributions of high ranked parameters. Nominal values are assigned for non-chosen parameters. We use SOBOL sequence [15] to redraw these samples. It uses a quasi-random low-discrepancy sequence, so these samples can cover the search spaces of parameters more uniformly than the previous samples for simulation. Suppose l is the number of redrawn samples and k high ranked parameters are chosen. Redrawn samples can be formed as

$$\begin{aligned} x_{m(i)} &= [x_1, x_2, \dots, x_k, x_{k+1}, \dots, x_m] \in R^m, i = 1, 2, \dots, l \\ [x_1, x_2, \dots, x_k] &= \mathbf{SOBOL}(x_m, \min(x_m), \max(x_m)) \\ [x_{k+1}, x_{k+2}, \dots, x_m] &= \mathbf{NOMINAL}(x_m) \end{aligned} \quad (10)$$

where $MIN(x_m)$ and $MAX(x_m)$ mean the minimum and maximum values of $x_{m(1,2,\dots,l)}$. As a result, we can generate samples with reduced dimensions while maintaining well-coverage of the failure region. The classifier with the updated threshold can filter out these samples to determine pass or fail condition of process parameters. With the classification result of samples, the proposed method induces *if-then* rules from the highest-ranked parameters so that all failure conditions of parameters can be filtered. The overall steps of the new in-spec guidance method are explained in Fig. 4.

C. Iterative calculation of failure probabilities

In the proposed iterative diagnosis method, the probability of updated failure region should be repeatedly calculated. In this subsection, we derive the complete formula, which was not given explicitly in existing works [3], [5].

The failure probability with given threshold t in (2) (which denotes $[1 - P(Y \leq t)]$) can be replaced by the PDF from simulation result with the threshold t_i in each iteration. So, the equation can be rewritten with $P_{MC}(Y \geq t_1)$, which denotes

the PDF of initial MC sampling simulation with the threshold t_1 .

$$\begin{aligned} P(Y \geq t_c) &= P_{MC}(Y \geq t_1) \cdot P(Y \geq t_c | Y \geq t_1) \\ P(Y \geq t_c | Y \geq t_1) &= \frac{P(Y \geq t_c, Y \geq t_1)}{P(Y \geq t_1)} \end{aligned} \quad (11)$$

The conditional probability part in (11) can be estimated by GPD fitting using simulated failure samples. Therefore, we can reformulate (11) as

$$P(Y > t_c) = P_{MC}(Y \geq t_1) \cdot P_{IS}(Y \geq t_c | Y \geq t_1) \quad (12)$$

where P_{IS} represents the conditional probability in the updated distribution by GPD fitting. If the proposed method iterates twice with t_1 and t_2 as threshold bounds, failure samples for the second simulation were filtered out again by the updated classifier based on t_2 . The failure probability with t_2 can be computed by multiplying the first failure probability with t_1 . Thus, the failure probability in each step can be calculated as

$$\begin{aligned} P(Y \geq t_c) &= P_{(2)}(Y \geq t_c) \\ P_{(1)}(Y \geq t_2) &= P_{MC}(Y \geq t_1) \cdot P_{IS(1)}(Y \geq t_2 | Y \geq t_1) \\ P_{(2)}(Y \geq t_c) &= P_{MC}(Y \geq t_1) \cdot P_{IS(1)}(Y \geq t_2) \\ &\quad \cdot P_{IS(2)}(Y \geq t_c | Y \geq t_2) \\ P_{IS(i)}(Y \geq t_c | Y \geq t_i) &= \frac{P_{IS(i)}(Y \geq t_c)}{P_{IS(i)}(Y \geq t_i)} \end{aligned} \quad (13)$$

Without the loss of generality, we can formulate the iterative failure probability calculation as follows. Suppose that the iteration loops k times,

$$\begin{aligned} P_{(i)}(Y \geq t_c) &= \begin{cases} P_{MC}(Y \geq t_i) \cdot P_{IS(i)}(Y \geq t_c | Y \geq t_i) & \text{for } i = 1 \\ P_{MC}(Y \geq t_i) \cdot \prod_{i=1}^{k-1} (P_{IS(i)}(Y \geq t_{i+1})) \\ \quad \cdot P_{IS(k)}(Y \geq t_c | Y \geq t_k) & \text{for } i > 1 \end{cases} \end{aligned} \quad (14)$$

The final failure probability can be obtained by recursively applying (14) in each iteration.

4. Experimental Results and Discussions

The proposed method (IFRD) was implemented in Python 2 and tested on a Linux workstation with 32 CPUs (2.6GHz Xeon processors) and 64GB RAM. The performance and accuracy of proposed method have been evaluated by verifying the failure rate of 6T-SRAM single-bit cell and charge pump circuit in PLL, which are highly replicated instances for system-on-chip (SOC) designs. Both of circuits were designed with BSIM4 transistor model and simulated in NGSPICE [16]. Table. 1 shows 9 major process parameters of MOSFET. We performed four different methods (MC, REscope [7], RSB [5] and IFRD) to compare their accuracy and performance.

A. 6T-SRAM failure rate diagnosis

The schematic design of the single-bit 6T-SRAM cell is shown in Fig. 5. The read operation of 6T-SRAM fails when the voltage gap between BL and \overline{BL} is not enough to be determined by sense amplifiers in a certain period. We measure the delay of discharging \overline{BL} as the failure criterion. Our experimental setup for the initial conditions are: $\overline{Q}='L'$, $\overline{Q}='H'$, BL and $\overline{BL} = 'H'$. When WL turns on, \overline{BL} is discharged by MN2 and MN1 and BL charged by MP2. So, the number of the process parameters is 27(3*9) in this reading

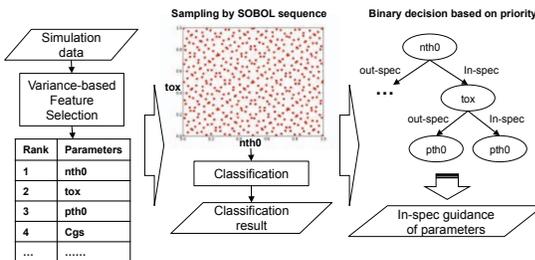


Fig. 4: In-spec guidance of parameters calculation flow 1) Feature selection 2) Sampling and Classification 3) Calculate the boundaries for in-spec conditions

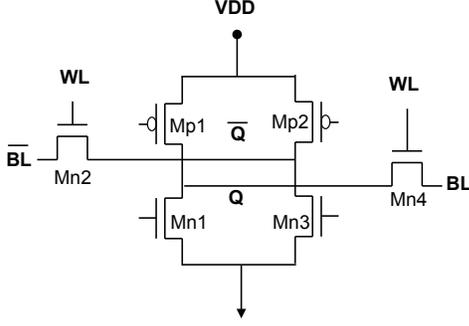


Fig. 5: The schematic of 6T-SRAM single-bit cell

operation. The initial number of sampling for capturing the circuit behavior is 2,000. We set the failure criterion t_c as $P(Y \geq t_c) = 0.00023$, which means 3-sigma in terms of the yield level. The proposed method is iterated twice with 97% percentile bound as a slope guard to separate the failure region from initial distribution. Hence, threshold bounds t_1 and t_2 are calculated as $P(Y \geq t_1) = 0.03$ and $P(Y \geq t_2) = 0.0009$, respectively. Table II shows the accuracy and performance of failure analysis performed by different approaches. Comparing the traditional MC method, IFRD shows 63x speedup gain with 24.00% relative error. Smart sampling scheme is applied with a selection ratio of 0.5. That is to say, we only keep half of original likely-to-fail samples for real simulation. RSB costs 2x more simulations while receiving a 3.35% increase in accuracy. For the estimated specification guidance for parameters, we find that only 1.2% of samples, which meet the in-spec guidance, are determined as the failure by the classifier in Table III

Fig. 6 shows that our proposed method is more accurate than previous methods since the tail of CDF depicting the 3-sigma failure region is more correlated to the golden reference (MC).

B. Charge pump failure rate diagnosis

In a large logic circuit, a clock is frequently distributed to several sub-clocks, so frequencies of sub-clocks are easy to be inaccurate due to propagation delays. A PLL is frequently

TABLE I: Process parameters of MOSFET

Variable name	Std(σ)	Unit
Flat-band voltage(V_{fb})	0.1	V
Gate oxide thickness(t_{ox})	0.05	m
Mobility (μ_0)	0.1	m^2/Vs
Doping concentration at depletion (N_{dep})	0.1	cm^{-3}
Channel-length offset (ΔL)	0.05	m
Channel-width offset (ΔQ)	0.05	m
Source/drain sheet resistance(R_{sh})	0.1	$\Omega m/mm^2$
Source-gate overlap unit capacitance(C_{gso})	0.1	F/m
Drain-gate overlap unit capacitance(C_{gdo})	0.1	F/m

TABLE II: Comparison of the accuracy and efficiency on 6T-SRAM circuit

	Failure probability	# Sim. runs	Speed-up(x)	Error%
Monte Carlo(MC)	2.300E-04	1M	-	-
Rare Event Microscope (REscope)	3.786E-04	5009	199.6	64.61
Recursive Statistical Blockade (RSB)	2.775E-04	29260	34.2	20.65
Proposed method (IFRD)	2.852E-04	15730	63.6	24.00

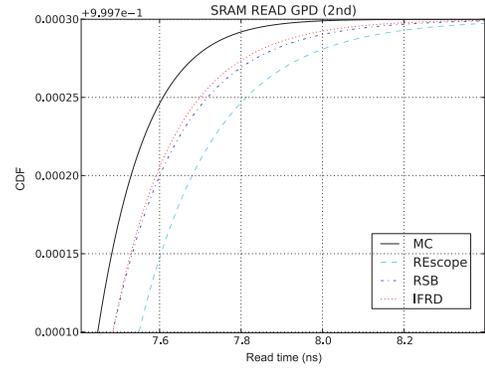


Fig. 6: Estimating the CDF of 6T-SRAM readtime around 3-sigma region

TABLE III: Estimated in-spec guidance of parameters on 6T-SRAM circuit

Rank	Parameter @MOSFET	Initial condition (μ, σ)	In-spec Guidance [MIN,MAX]
1	$v_{fb}@MN1$	(-5.5E-01,0.1)	[-7.31E-01,-3.78E-01]
2	$v_{fb}@MP2$	(5.5E-01,0.1)	[3.54E-01,7.15E-01]
3	$ndep@MN1$	(2.8E+18,0.1)	[1.84E+18,3.76E+18]
4	$ndep@MN2$	(2.8E+18,0.1)	[1.70E+18,3.84E+18]
4	$ndep@MP2$	(2.8E+18,0.1)	[1.89E+18,3.78E+18]
Failure probability	0.0009(= t_2)	Estimation Error (%)	1.21

used to adjust the phase of the clock. The functional block diagram of PLL is shown in fig. 7. After comparing the output clock (CLK_{out}) with the reference clock (CLK_{ref}) by phase detector, a charge pump circuit actually adjusts the frequency of clock signal by charging and discharging capacitors rely on input signals (UP and DN). The mismatch of MOSFETS in a charge pump can cause the unbalanced timing and phase jitter between two different operation modes. Hence, we measure the timing ratio of charging and discharging operations, which can be formulated mathematically as $r_{min} \leq \frac{t_{discharge}}{t_{charge}} \leq r_{max}$ ($r_{min,max}$ represents the minimum and maximum ratio to determine failures). A charge pump circuit consists of 9 MOSFETS as shown in fig. 8. The total number of process parameter is 81 ($9*9$), so the dimensionality of parameters is much higher than 6T-SRAM case. We perform 2,000 initial sampling and simulation to model the initial performance distribution accurately. Similar to the 6T-SRAM case, we perform our algorithm twice with 97% percentile bound ($P(Y \geq t_1) = 0.03$, $P(Y \geq t_2) = 0.0009$). The result is summarized in Table IV. Our approach only required 6,263 sampling for estimating the failure probability of 3-sigma region with 10.78% error, while MC needs 1 million samples.

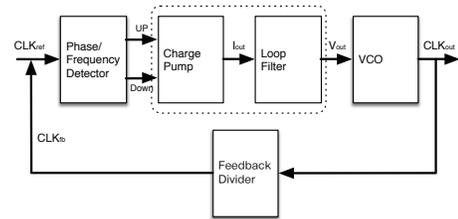


Fig. 7: A functional diagram of PLL

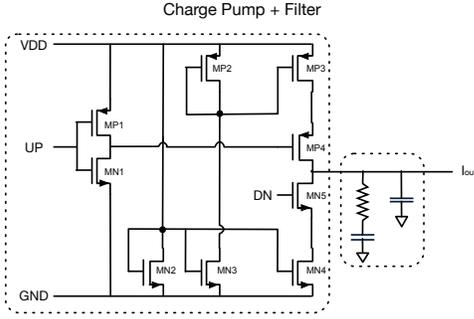


Fig. 8: Schematic representations of charge pump and filter

REscope gains best speedups with 1388 less simulation runs, however, is far from accurate in failure region estimation. RSB here achieves a best accuracy by consuming double simulation runs compared to IFRD. Table V shows that the proposed method makes the decision for in-spec conditions of process parameters with 98% confidence level by managing only the first 5 ranked parameters of 81. The tail distribution of IFRD in 3-sigma failure region is much close to MC than REscope as we can see in Fig. 9.

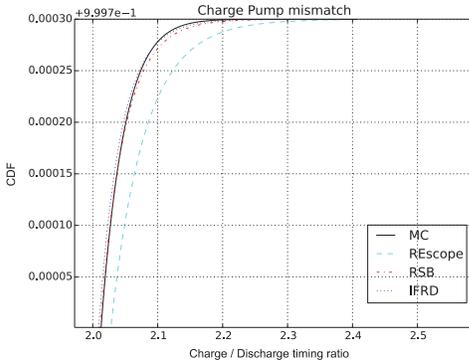


Fig. 9: Estimating the CDF of charge pump mismatch around 3-sigma region

5. Conclusion

In this paper, we presented the novel statistical diagnosis method for rare failure events. The proposed method introduced two new techniques to speed up the failure analysis while providing the in-spec guidance of process parameters to facilitate the design and optimization process. First, the proposed method applied the smart sample selection method to reduce the additional simulation cost during iterative failure region locating process. Second, the new approach can provide

TABLE IV: Comparison of the accuracy and efficiency on charge pump circuit

	Failure probability	# Sim. runs	Speed-up(x)	Error(%)
Monte Carlo(MC)	2.300E-04	1M	-	-
Rare Event Microscope (REscope)	3.337E-04	4875	205.1	45.09
Recursive Statistical Blockade (RSB)	2.245-04	10432	95.9	2.39
Proposed method (IFRD)	2.052-04	6263	159.7	10.78

TABLE V: Estimated in-spec guidance of parameters on charge pump circuit

Rank	Parameter @MOSFET	Initial condition (μ, σ)	In-spec Guidance [MIN,MAX]
1	$ndep@MN1$	(2.8E+18,0.1)	[1.74E+18,3.78E+18]
2	$ndep@MP2$	(2.8E+18,0.1)	[1.68E+18,3.73E+18]
3	$ndep@MP4$	(2.8E+18,0.1)	[1.78E+18,3.81E+18]
4	$ndep@MN5$	(2.8E+18,0.1)	[1.78E+18,3.77E+18]
4	$ndep@MP3$	(2.8E+18,0.1)	[1.88E+18,3.80E+18]
Failure probability	0.0009(= t_2)	Estimation Error (%)	2.00

safe design space of parameters, which can help design to improve the yield and meet the target performance of the design. Experimental results show that the new method can be 4X more accurate than the recently proposed REscope method. Furthermore, the new method reduces the simulation cost by 2X than the recursive statistical blockage (RSB) method with same accuracy level. Our approach also provides the precise guidance of diverse parameters with 1.21% estimation error.

References

- [1] B. Calhoun, Y. Cao, X. Li, K. Mai, L. Pileggi, R. Rutenbar, and K. L. Shepard, "Digital circuit design challenges and opportunities in the era of nanoscale cmos," *Proceedings of the IEEE*, vol. 96, pp. 343–365, Feb 2008.
- [2] H. Masuda, S. Ohkawa, A. Kurokawa, and M. Aoki, "Challenge: variability characterization and modeling for 65- to 90-nm processes," in *Proc. IEEE Custom Integrated Circuits Conference (CICC)*, pp. 593–599, Sept 2005.
- [3] A. Singhee and R. Rutenbar, "Statistical blockade: Very fast statistical simulation and modeling of rare circuit events and its application to memory design," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 28, pp. 1176–1189, Aug 2009.
- [4] D. Hocevar, M. Lightner, and T. N. Trick, "A study of variance reduction techniques for estimating circuit yields," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 2, pp. 180–192, July 1983.
- [5] A. Singhee, J. Wang, B. Calhoun, and R. Rutenbar, "Recursive statistical blockade: An enhanced technique for rare event simulation with application to sram circuit design," in *International Conference on VLSI Design*, pp. 131–136, Jan 2008.
- [6] W. Wu, F. Gong, G. Chen, and L. He, "A fast and provably bounded failure analysis of memory circuits in high dimensions," in *Proc. Asia South Pacific Design Automation Conf. (ASPDAC)*, pp. 424–429, Jan 2014.
- [7] W. Wu, W. Xu, R. Krishnan, Y.-L. Chen, and L. He, "Rescope: High-dimensional statistical circuit simulation towards full failure region coverage," in *Proc. IEEE/ACM Design Automation Conference (DAC)*, pp. 1–6, June 2014.
- [8] S. Mukhopadhyay, H. Mahmoodi, and K. Roy, "Statistical design and optimization of sram cell for yield enhancement," in *Proc. Int. Conf. on Computer Aided Design (ICCAD)*, pp. 10–13, Nov 2004.
- [9] K. Agarwal and S. Nassif, "Statistical analysis of sram cell stability," in *Proc. IEEE/ACM Design Automation Conference (DAC)*, pp. 57–62, 2006.
- [10] M. Qazi, M. Tikekar, L. Dolecek, D. Shah, and A. Chandrakasan, "Loop flattening & spherical sampling: Highly efficient model reduction techniques for sram yield analysis," in *Proc. Design, Automation and Test in Europe. (DATE)*, pp. 801–806, March 2010.
- [11] S. N. R. Kanj, R. Joshi, "Mixture important sampling and its application to the analysis of sram designs in the presence of rare failure events," in *Proc. IEEE/ACM Design Automation Conference (DAC)*, pp. 69–72, 2006.
- [12] T. Hastie, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, 2009.
- [13] J. J. R. Hosking, "Parameter and quantile estimation for the generalized pareto distribution," *Technometrics*, vol. 29, no. 3, pp. 339–349, 1987.
- [14] J. R. Hosking, "Algorithm as 215: Maximum-likelihood estimation of the parameters of the generalized extreme-value distribution," *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, vol. 34, no. 3, pp. 301–310, 1985.
- [15] S. Joe and F. Y. Kuo, "Remark on algorithm 659: Implementing sobol's quasirandom sequence generator," *ACM Trans. Math. Softw.*, vol. 29, pp. 49–57, Mar. 2003.
- [16] P. Nenzi and V. Holger, "Ngspice users manual," *Version 25plus*, 2010.