Variability-Aware Parametric Yield Estimation for Analog/Mixed-Signal Circuits: Concepts, Algorithms and Challenges

Fang Gong
Cadence Design Systems, Inc.
2655 Seely Ave.,
San Jose, CA
fgong@cadence.com

Yiyu Shi
Department of ECE
Missouri University of Science and Technology
Missouri, Rolla
yshi@mst.edu

Hao Yu
School of Electrical and Electronic Engineering
Nanyang Technological Univeristy
Singapore
haoyu@ntu.edu.sg

Lei He
Department of EE
University of California at Los Angeles
Los Angeles, CA
lhe@ee.ucla.edu

***Abstract*—with technology scaling down to 90nm and below, process variation has become a primary challenge for both design and fabrication of analog/mixed-signal circuits due to significantly increased circuit failures and yield loss. As a result, it is urgently required to estimate the yield of one design efficiently in the presence of process variation. In this paper, we present the recent advance for yield estimation for analog/mixed-signal circuits with a number of critical topics and techniques discussed and classified into two categories. The first is performance domain method, which requires extensive Monte Carlo simulations; and the second is parameter domain method, which requires the characterization of yield boundary defined by performance constraints without using Monte Carlo. We review the pros and cons of these methods, which are evaluated by a number of circuit examples with quantitative comparison.**

Keywords- Process Variation, Yield Analysis, Circuit Simulation, Monte Carlo.

I. INTRODUCTION

Process variation has become the dominant challenge for both analog circuit design and fabrication at nano-scale. Many uncertainties can be introduced during manufacturing process such as lithography, chemical mechanical polishing (CMP), etching, etc. Consequently, circuit parameters such as effective channel length L_{eff} and threshold voltage V_{th} can deviate significantly from their nominal values. This in turn will cause circuit performance merits (such as delay, period, maximum clock frequency, leakage power and etc.) to differ from design specifications.

The undesired uncertainties in circuit performance can lead to analog/mixed-signal circuit failures and yield loss at nano-scale. As such, it has become extremely critical for high-precision analog/RF circuits such as phase locked loop (PLL) and custom/mixed-signal circuits such as SRAM arrays, which both have tight operating margin due to lower power supply and higher operating frequency. Thereby, the parametric yield, defined as the percentage of fabricated circuits that can function correctly, has become one important criterion to evaluate design robustness under process variation at nano-scale.

In general, the parametric yield can be estimated in either the *performance domain* or the *parameter domain* as shown in Figure (1). The *performance domain* is defined as a space consisting of all possible performance merits (i.e. voltage, gain, bandwidth and etc.), while the *parameter domain* is a space spanned by all circuit parameters (i.e. channel width, threshold voltage and etc.) and bounded by their minimum and maximum values. In both domains, the successful samplings form a bounded region called the “yield body” or “success region”, and the nonlinear boundary that separates the success and failure regions is called the “yield boundary”. As a result, the percentage of satisfied samplings in both domains can be used to estimate the yield rate equivalently.

Figure 1: Yield estimation in (a) parameter domain and (b) performance domain.

Many performance-domain techniques have become available in past few decades: Monte Carlo method repeatedly draws samples, runs simulations and evaluates the yield rate, which can be easily applied to high-dimensional problems. However, it is extremely time-consuming. To relieve the problem, Quasi Monte Carlo [1][2] has been proposed to generate quasi-random numbers rather than purely-random samplings, which can save a large number of samples. Similarly, Latin Hypercube Sampling (LHS) [3] has been developed to generate multivariate samples by dividing the range of each variable into equally probable intervals and placing only one sample in each grid cube containing the same position (or called Latin hypercube). Moreover, Importance Sampling [4] [5][6] has been established to efficiently handle the rare event (i.e. millions samples can capture only one failure) which changes the sampling density to draw more failed samples.

Meanwhile, a large number of parameter-domain methodologies have been developed as well: nonlinear surface sampling [7] locates the points on the “yield boundary” by searching along the nonlinear performance surface in the parameter domain. Global search method [8] can find one point on the “yield boundary” with one-time simulation and thus save more runtime. Response surface modeling method [9] tries to model the performance as a polynomial function of all variable parameters and then evaluate the yield estimation. In addition, design centering method [10], performance modeling method [11] and other advanced techniques can also be used for parametric yield estimation.

In this article, we present several above-mentioned techniques, including the Monte Carlo method and its variants (e.g. Quasi Monte Carlo [1][2], Importance Sampling[4][5][6]) in the performance domain, and two recent-established methods in the parameter domain (e.g. nonlinear surface sampling method [7] and global search method [8]). We have compared all these approaches quantitatively by several circuit examples from the perspectives of both accuracy and efficiency. Moreover, we discuss the advantage and limitations of each method and further analyze the remaining challenges for parametric yield estimation of analog/mixed-signal circuits.

It is worthwhile to point out that this article only covers a subset of well-established techniques in this article due to the limited space, and interested readers are encouraged to refer to the resources in the reference section.

II. Performance Domain Yield Estimation

In this section, we discuss the yield estimation methods, which are in the performance domain and include Monte Carlo method and its derivations. For illustration purpose, we first briefly present the fundamental idea behind each method and then have a quantitative comparison using one 6-T SRAM cell example at the end of this section.

A. *Monte Carlo Method*

The most straight-forward method in the performance domain is Monte Carlo method (MC), which first generates tens of thousands of samples with the probability distribution of variable parameters. Then, it performs circuit simulation with each sample to evaluate the performance merit of interest. With the given performance constraints, Monte Carlo method can identify the successful or acceptable samples. In this way, the percentage of successful samples can be used to estimate the yield rate.

The advantage of Monte Carlo method is its simplicity and generality; it can be applied to arbitrary distributions of parameters and performance merits without any *priori* knowledge. However, it is very time-consuming to achieve high accuracy since its convergence rate to the exact value is only $O(1/\sqrt{N})$, where $\boldsymbol{N}$ is the total number of samples. Therefore, it is not suitable for practical applications.

B. *Quasi Monte Carlo Method*

An alternative to the Monte Carlo approach is called Quasi Monte Carlo (QMC) method [1][2], which uses quasi-random sequences rather than purely-random samplings. QMC starts with the generation of quasi-random numbers (or representative samples), such as Faure (1982), Neiderreiter (1987), Sobol (1967) or Halton (1960) sequences. It then converts the samples following those specific distributions to ones following the desired distribution.

For example, the Sobol sequence follows a uniform distribution $\boldsymbol{u}\sim\boldsymbol{U}(0,1)$ and can be converted to Gaussian distribution using $\boldsymbol{x}=\boldsymbol{F}^{-1}(\boldsymbol{u})$, where $\boldsymbol{F}^{-1}$ is the *inverse cumulative distribution function* (inverse CDF) of the Gaussian distribution. The resultant sequence $\boldsymbol{x}$ follows the Gaussian distribution. As such, circuit simulation can be performed with $\boldsymbol{x}$ sequence to obtain performance merits, and the yield is estimated as the percentage of successful samples.

The key idea behind the Quasi Monte Carlo is try to cover the entire parameter space evenly with *deterministic* samples rather than pure random numbers. Therefore, QMC can use fewer samples and potentially improve both accuracy and efficiency when compared with Monte Carlo approach.

In addition, the convergence rate of Quasi Monte Carlo can be reduced to $O(1/N)$ in the optimal cases, much faster than Monte Carlo approach. However, the convergence rate becomes $O(ln(N^d)/N)$ for multiple dimensions in the worst case, where d is the number of dimensions. As such, the performance of QMC can degrade with the dimensions.

C. *Importance Sampling*

To improve the efficiency of MC and QMC, the Importance Sampling (IS) method has been proposed [4][5][6], which shifts the sampling distribution towards the failure region and makes each sample more useful.

For example, we consider the original probability density function (PDF) of a variable parameter shown in Figure (2), and assume the failure region locates at right tail. As such, IS tries to shift the sampling function towards the failure region which can increase the probability of sampling within the failure region.

Figure 2: Shifting sampling distribution in Importance Sampling [5]

Importance sampling can reduce the number of samples required to achieve a desired accuracy, especially in the case where the failure region is small for rare failure events [4]. However, it is always challenging to obtain an optimal sampling distribution or shift vector efficiently (e.g. mean shift vector for one parameter in Figure (2)), since this depends on the actual distribution of the performance merit and is unknown beforehand. Therefore, many techniques have been developed to find the optimal sampling distribution or shift vector in recent literatures [5][6], which will not be discussed in this article due to limited page space.

D. *Experimental Results*

To evaluate the discussed methods, we apply them to estimate yield of a six-transistor SRAM cell shown in Figure (3), which is particularly vulnerable to process variation due to ever-decreasing supply voltage and reduced noise margin. One SRAM cell is used to store one memory bit, and a typical implementation involves six transistors: the four transistors Mn1, Mn3, Mp5 and Mp6 have two stable states, i.e., either a logic ‘0’ or ‘1’, and the two additional access transistors Mn2 and Mn4 serve to control the access to the cell during read and write operations. The word line is used to determine whether the cell should be accessed (connected to bit line) or not, and the bit line is used to read/write the actual data from/to the cell.

In this paper, the SRAM yield is defined by the *data retention failure* due to static noise margin (SNM) variations, and the threshold voltages of all transistors are modeled as independent random variables. In general, the SNM is used to measure the amount of noise voltage that is needed to flip the stored data in one SRAM cell. Typically SNM can be measured by the length of the maximum embedded square in the butterfly curves which consists of the voltage transfer curves (VTC) of the two inverters in SRAM cell. When SNM is smaller than zero, the butterfly curves collapse and data retention failure happens.

Figure 3: Schematic of a 6-transistor SRAM Cell.

To investigate how the various methods might behave in a more realistic context, we apply the Monte Carlo, Mixture Importance Sampling [4], Spherical Sampling [6] to estimate the yield rate due to data retention failures, where both Hspice and BSIM4 transistor model are used. Note that the failures have become *rare events* and the yield rate is highly close to 99.99%. As such, we first compare the evolution of failure rate estimation from different methods in Fig.(4), which shows MC converge very slowly and Quasi Monte Carlo can significantly improve the convergence. In addition, both Mixture Importance Sampling and Spherical Sampling can reach the similar estimation results with much fewer samples.

Moreover, Table 1 further shows the number of samples and circuit simulations required by different methods when the target accuracy level and confidence interval are set to 95%. From this table, it can be observed that Quasi Monte Carlo method can achieve around 5.3X speedup over Monte Carlo method, while Importance Sampling based methods can be tens or even hundreds times faster than MC method with the same accuracy. Also, it is worthwhile to point out that the performance of Importance Sampling based methods is significantly sensitive to the shifted sampling distribution, which can be demonstrated by the different speedup obtained from Mixture Importance Sampling and Spherical Sampling.

Figure 4: Evolution comparison of failure rate estimation for different methods (*VDD*=290mV).

	Monte Carlo	**Quasi-MC**	**Mixture Importance Sampling**	**Spherical Sampling Method**
Prob.(failure)	8.66E-4	8.64E-4	8.69E-4	8.23E-4
Accuracy	95%	95%	95%	95%
# simulation runs	2.5E+6	4.72E+5	1.26E+5	5951
Speedup	1X	5.3X	20X	420X

Table 1: Number of simulations runs required by performance-domain methods to achieve the same accuracy for SRAM cell failure estimation.

We can see that previous conclusions can still hold: Monte Carlo method tries to randomly select samplings to cover the entire space evenly, so it needs a huge number of samplings. Quasi Monte Carlo can reduce the number of samplings by covering the entire space with deterministic sequences. Moreover, Importance Sampling method can shift the sampling distribution towards the failure regions so that to pick more failed samplings for improved accuracy and efficiency.

III. PARAMETER DOMAIN YIELD ESTIMATION

In order to avoid the large number of samples and simulations for estimating yield at the performance domain, several approaches have been proposed to estimate the yield in the parameter domain. Generally, the variable circuit parameters with performance constraint can define one *hyper-volume* in the parameter domain, and the points within this hyper-volume denote those samples that provide satisfied performance merits. Therefore, the parametric yield estimation is to calculate the ratio of the volume of the hyper-volume to that of the entire parameter domain.

One most straight-forward method is to find all the points in the hyper-volume with massive samples, which is expensive and unnecessary because the points on the hyper-surface boundary can provide adequate information of the hyper-volume. Therefore, the parametric yield estimation methods in the parameter domain aim to locate points on the hyper-surface boundary (or called “yield boundary”) efficiently and accurately.

In this section, we will present two yield estimation approaches (i.e., Nonlinear Surface Sampling [7] and QuickYield [8] in the parameter domain, which can efficiently estimate the yield with very high accuracy along with hundreds times speedup over Monte-Carlo. Also, we use a ring-oscillator as an example to compare these methods and the reasons are two-fold: 1) QuickYield [8] can only be applied to DC and PSS (periodical steady state) analyses and the evaluation of oscillator period needs PSS analysis. 2) The methods in parameter domain calculate the yield rate with geometrical computation (i.e., area, volume), therefore, they become very inaccurate for circuits with “rare failure events” (i.e., the failure probability of SRAM circuit is very small) because the failure region becomes very tiny and the yield rate is very close to 100%.

A. Nonlinear Surface Sampling

Yield Estimation with Nonlinear Surface Sampling [7] (or called “YENSS”) aims to find the yield boundary as shown in Figure (5(a)). For example, the performance surface can be defined with each point on the surface corresponding to a sampling point in the parameter domain. With the given performance constraint, the surface can be divided into success and failure regions, which are separated by the intersection of these two surfaces (called the “yield boundary” shown in Figure (5(b))). To that end, it starts from the nominal design point (P0 in Figure (5(a))) and searches along the tangent direction *locally* on the performance surface. As such, the yield can be estimated by the area (or, volume) ratio of the bounded region to that of the entire parameter domain.

(a) (b)

Figure 5: (a) Performance surface over parameter domain, and (b) the yield boundary in the parameter domain [7]

This method can avoid massive sampling but has two-fold disadvantages: firstly, local search is inefficient to find multiple boundary points, since a large number of expensive simulations are required; secondly, this method can only handle small-scale problems with up to three parameters. In [8], we have shown that a *global search* algorithm can resolve these limitations.

B. Global Search Based Yield Estimation

According to our discussion so far, previous methods in the parameter domain mostly utilize “local search” shown in Figure (6(a)): generating samples of variable parameter, performing simulation at samples, and then comparing with the given performance constraints to find the samples on the yield boundary. Typically the iteration needs to be repeated many times to find one point on the yield boundary.

(a) Local Search (b) Global Search

Figure 6: Comparison of algorithms to find the yield boundary

In order to improve the efficiency, “global search” has been proposed in [8] recently to find each point on the yield boundary with one-time simulation without sampling. It breaks the framework in “local search” by introducing the constraints as an extra equation into the original circuit system and treating the variable parameter as an unknown (as shown in Figure (6(b))).

Specifically, one circuit can be described by a differential algebraic equation (DAE) system as

$$\frac{d}{dt}q(x(t)) + f(x(t)) + b = 0 \tag{1}$$

Here, *x(t)* are the state variables including node voltage and branch current. *q(x(t))* contains active components such as charges and fluxes, and *f(x(t))* describes passive component and *b* denotes input sources. As such, the DAE system can determine a performance surface in the parameter domain shown in Figure (7(a)).

(a) (b)

Figure 7: (a) Rotated view and (b) Top view of the yield boundary

Meanwhile, the performance constraint further locates a constraint plane in Figure (7(a)). It is clear that the yield boundary in the parameter domain shown in Figure (7(b)) is the intersection boundary of these two surfaces in Figure (7(a)), which is equivalent to the solution of an augmented system including DAE and performance constraint. As a result, we can introduce the constraint into the DAE system and add the variable parameter as the extra unknown as:

$$\begin{cases} \frac{d}{dt}q(x(t)) + f(x(t)) + b = 0 \\ H(\gamma_p; f_m) = f_m(\gamma_p) - f_{worst} = 0 \end{cases} \tag{2}$$

The second equation of $H(\gamma_p; f_m)$ is the general expression of performance constraint [7][12], where γ_p is the variable parameter and f_{worst} is the worst performance that can be accepted. It should be clarified that above nonlinear system is a determined system when the number of variable parameters γ_p equals to the number of performance constraints $H(\gamma_p; f_m)$. However, when number of performance constraints is smaller than that of variable parameters, the nonlinear system can only consider part of parameters at one time in order to handle only deterministic system.

Moreover, this nonlinear system can be denoted as $F(x(t), \gamma_p) = 0$ for the ease of notation and be solved with Newton-Raphson method which needs to calculate the Jacobian matrix $J(X) = \partial F / \partial X$ ($X = \left[x^T; \gamma_p\right]$ is a vector of unknowns). For DC analysis, the Jacobian matrix becomes simply as:

$$J(X)=\begin{pmatrix} \partial f/\partial x & \partial f/\partial \gamma_p + b \\ \partial H(\gamma_p; f_m)/\partial x & \partial H(\gamma_p; f_m)/\partial \gamma_p \end{pmatrix} \tag{3}$$

Similarly, one augmented system for PSS analysis can be constructed in discrete time domain with the finite difference method. Interested audience are referred to [8][12] for more details. As such, the resultant solution contains the parameter value of γ_p corresponding to the point on the yield boundary in the parameter space, thereby, one boundary point can be found with only one-time simulation. Since it searches for the boundary point within the entire parameter space, we name this scheme as "global search". In the following, we further show the advantages of the global search based yield estimation.

C. Experimental Results

In this section, we consider a 3-stage ring oscillator as shown in Figure (8) to compare the methods in the parameter domain, where the period is considered as the performance merit. The nominal period of the oscillator T_{norm} is 7.2028ns calculated via periodical steady state (PSS) simulation. The design specification requires that the variation in period δT should be within ±2.5% of T_{norm}. For illustration purpose, we consider the effect of random variations in the width of MOSFET in the first stage with 40% perturbation range from their nominal values. The nominal width of Mp1 is 3*um* and that of Mn1 is 2*um*. Notice that the same methods can be applied to other parameter variations, such as threshold voltage, channel length, and etc.

(a) (b) (c)

Figure 8: (a) 3-stage ring-oscillator; Simulation results from (b) QuickYield; (c) Monte Carlo method.

In fact there exist two boundaries in the parameter space, which correspond to T_{min} and T_{max}. Therefore, we need to solve two augmented systems and each of them considers one performance constraint in the below:

$$\begin{cases} H_1 = T - T_{\min} = 0 \\ H_2 = T - T_{\max} = 0 \end{cases}$$

As a result, we can observe these two nonlinear boundaries from results of QuickYield in Figure (8(b)), which are identical to the result from Monte Carlo method in Figure (8(c)). Also, YENSS in [7] can capture the same boundaries as Figure (8(b)), but it needs more circuit simulations due to local search. Note that we consider two variable parameters (e.g., channel widths of the first stage as *Wp* and *Wn*), which implies that the system in (2) becomes an under-deterministic system. To solve it, we convert it into a deterministic system by fixing the values of *Wp* at different points (shown in Figure (8(b))) and solving for the corresponding values of *Wn*. In other words, the system (2) has one-dimensional $H(\gamma_p; f_m)$ and one-dimensional γ_p parameter.

To evaluate the efficiency with the same accuracy (95% accuracy with 95% confidence interval), we compare the runtime of Monte Carlo, YENSS obtained from [7] by normalizing with respect to its Monte Carlo runtime, and QuickYield in Table 2. From this table, the Monte Carlo method needs 44073.8 seconds, while YENSS can achieve139X speedup over it. Moreover, QuickYield can obtain 519X speedup over Monte Carlo and be 4X faster than YENSS at a similar accuracy.

Method	**Total Simulation Time (second)**	**Speedup**
Monte Carlo	44073.8	1X
YENSS	317	139X
QuickYield	84.9	519X

Table 2: Number of circuit simulations required by parameter-domain methods to achieve the same accuracy for SRAM cell yield estimation.

Note that the runtime comparison in Table 2 only contains the runtime of circuit simulations, and thereby QuickYield can achieve more speedup over YENSS when the expensive sensitivity analysis in YENSS is considered. Furthermore, extensive experiments show that the accuracy and runtime of QuickYield can scale with the number of boundary points [8], which is suitable for analog/RF circuits with the high-precision design requirement.

IV. Discussion and Conclusion

The methods in the performance domain (e.g. Monte Carlo method and its variants) tend to follow the "sampling-and-checking" procedure and thereby make them extremely time-consuming. As for the Importance Sampling, it is a critical but not trivial problem to find one good alternative distribution as shown in Fig.(2) which can provide fast convergence rate of probability estimation so as to reduce the number of required samples and circuit simulations. As a summary, the primary unresolved issue for methods in the performance domain is how to efficiently identify as few samples as possible yet to provide accurate yield estimation.

As for methods in the parameter domain, on the one hand, they can provide extremely efficient yield estimation by searching yield boundary and approximate the yield with the ratio of area or volume of the success region to that of the entire parameter space; on the other hand, they are also facing the same high-dimensional challenge as the Monte-Carlo based sampling approaches: usually a typical process design kit contains more than 100 independent random variables to model the global variations and 5-10 extra independent random variables to model random mismatches for a single transistor. As such, it easily involves 1000+ random variables in the parametric yield analysis of a typical analog/mixed-signal circuit. Therefore, the methods in the parameter domain become quite computationally expensive for high-dimensional problems, since these methods require expensive calculations for the hyper-volume of the success region in the high-dimensional space. Moreover, when "rare failure events" are considered, the area/hyper-volume of the failure region becomes very tiny so that the methods in parameter domain depending on calculation of area/volume ratios cannot provide accurate yield estimation. For a subset of cases - DC and PSS - QuickYield can provide significant additional speed improvements over YENSS.

In this article, the problem of variability-aware parametric yield estimation for analog/mixed-signal circuits is introduced: the first is the existing approaches in the performance domain, such as Monte Carlo method and its variants; and the second is the newly developed approaches in the parameter domain, such as YENSS and QuickYield. The pros and cons of all these methods are evaluated and compared using different circuit examples. Moreover, the potential challenges facing the variability-aware parametric yield estimation are also elaborated to facilitate the future research.

REFERENCE

[1] H. Niederreiter, Random Number Generation and Quasi-Monte Carlo Methods, Society for Industrial and Applied Mathematics, 1992.

[2] Amith Singhee and Rob A. Rutenbar, "From Finance to Flip Flops: A Study of Fast Quasi-Monte Carlo Methods from Computational Finance Applied to Statistical Circuit Analysis," Proc. IEEE 8th International Symposium on Quality Electronic Design (ISQED), March 2007.

[3] R. L. Iman (1999). "Latin Hypercube Sampling," Encyclopedia of Statistical Sciences, Update Volume 3, Wiley, NY, 408-411.

[4] R. Kanj, R. Joshi and S. Nassif, "Mixture Importance Sampling and Its Application to the Analysis of SRAM Designs in the Presence of Rare Failure Events", Proc. DAC, 2006, pp. 69-72.

[5] Kentaro Katayama, Shiho Hagiwara, Hiroshi Tsutsui, Hiroyuki Ochi, and Takashi Sato. "Sequential Importance Sampling for Low-Probability and High-Dimensional SRAM Yield Analysis," in Proc. of ACM/IEEE International conference on computer-aided design (ICCAD), pp.703-708, 2010.

[6] M. Qazi, M. Tikekar, L. Dolecek, D. Shah, and A.P. Chandrakasan, "Loop Flattening & Spherical Sampling: Highly Efficient Model Reduction Techniques for SRAM Yield Analysis" in Proc. Design, Automation, and Test in Europe Conference (DATE), March 2010, pp. 801-806.

[7] C. Gu and J. Roychowdhury. "An Efficient, Fully Nonlinear, Variability-Aware Non-Monte-Carlo Yield Estimation Procedure with Applications to SRAM Cells and Ring Oscillators", in Proc. of Design Automation Conference (ASP-DAC), Asia and South Pacific, 2008, pp. 754-761.

[8] Fang Gong, Hao Yu, Yiyu Shi, Daesoo Kim, Junyan Ren and Lei He, "QuickYield: An Efficient Global-Search Based Parametric Yield Estimation with Performance Constraints," in Proc. of Design Automation Conference, 2010, pp. 392-397.

[9] Soner Yaldiz, Umut Arslan, Xin Li, Larry Pileggi, "Efficient statistical analysis of read timing failures in SRAM circuits," 10th International Symposium on Quality of Electronic Design, 2009, pp.617-621,

[10] Robert Schwencker, Frank Schenkel, Helmut E. Graeb, Kurt Antreich, "The Generalized Boundary Curve-A Common Method for Automatic Nominal Design and Design Centering of Analog Circuits", DATE 2000: 42-47

[11] Trent McConaghy, Georges G. E. Gielen, "Analysis of simulation-driven numerical performance modeling techniques for application to analog circuit optimization", ISCAS 2005, pp.1298-1301

[12] I. Vytyaz, D. C. Lee, S. Lu, A. Mehrotra, U.-K. Moon, and K. Mayaram, "Parameter finding methods for oscillators with a specified oscillation frequency," in Proc. of Design Automation Conference, 2007, pp. 424-429.

Author Bios:

Fang Gong received his Ph.D. degree in the Electrical Engineering at University of California, Los Angeles in 2012. His research interests mainly focus on numerical computing and stochastic techniques for CAD, including fast circuit simulation, yield estimation and optimization. Currently he was working as a senior member of technical staff in Cadence Design Systems, Inc.

Yiyu Shi received his Ph.D. degree in Electrical Engineering from the University of California, Los Angeles 2009. He joined the faculty of Electrical and Computer Engineering Department at Missouri University of Science and Technology from Sept. 2010. His current research interest include advanced design and test technologies for 3D ICs, and renewable energy applications.