

# A Fast Analog Circuit Yield Estimation Method for Medium and High Dimensional Problems

Bo Liu, Jarir Messaoudi, Georges Gielen

ESAT-MICAS, Katholieke Universiteit Leuven, Leuven, Belgium. E-mail: {Bo.Liu, Jarir.Messaoudi, Georges.Gielen}@esat.kuleuven.be

## ABSTRACT

Yield estimation for analog integrated circuits remains a time-consuming operation in variation-aware sizing. State-of-the-art statistical methods such as ranking-integrated Quasi-Monte-Carlo (QMC), suffer from performance degradation if the number of effective variables is large (as typically is the case for realistic analog circuits). To address this problem, a new method, called AYLeSS, is proposed to estimate the yield of analog circuits by introducing Latin Supercube Sampling (LSS) technique from the computational statistics field. Firstly, a partitioning method is proposed for analog circuits, whose purpose is to appropriately partition the process variation variables into low-dimensional sub-groups fitting for LSS sampling. Then, randomized QMC is used in each sub-group. In addition, the way to randomize the run order of samples in Latin Hypercube Sampling (LHS) is used for the QMC sub-groups. AYLeSS is tested on 4 designs of 2 example circuits in 0.35  $\mu\text{m}$  and 90nm technologies with yield from about 50% to 90%. Experimental results show that AYLeSS has approximately a 2 times speed enhancement compared with the best state-of-the-art method.

## Keywords

Yield estimation, analog circuits, Latin Supercube Sampling (LSS)

## 1. INTRODUCTION

In modern technologies, random and systematic process variations have a large influence on the quality and yield of manufactured analog circuits. Currently, Monte-Carlo (MC) simulation-based yield estimation is the most popular method because of its advantages of generality and high accuracy [1]. For a single yield estimation of an analog cell with a reasonable accuracy requirement, even primitive MC (PMC) simulation often does not cost much computational effort, with a CPU time of several minutes. However, when it comes to variation-aware analog sizing or yield optimization [2-4], which requires many yield estimations, the inefficiency of MC-based yield estimation becomes the bottleneck. [2,4] present the first a few efficient variation-aware analog sizing methods based on MC simulation by introducing computational intelligence techniques to increase the efficiency, which makes the computation time practical. On the other hand, another possible way, compatible with the computational intelligence techniques is to contribute to the efficiency of variation-aware analog sizing by developing more efficient MC-based yield estimation techniques. Therefore, this paper focuses on a fast, accurate and general MC-based yield estimation technique for analog circuits with a reasonable accuracy requirement (high-sigma yield estimation is out of the

scope of this work, as in variation-aware analog sizing we often require a reasonably good accuracy). This paper is organized as follows. Section 2 reviews the available methods and provides the motivations for the new method, AYLeSS (analog yield estimation using Latin Supercube Sampling). Section 3 introduces the AYLeSS method. Section 4 tests AYLeSS on practical examples and compares AYLeSS with the best state-of-the-art method. The concluding remarks are presented in Section 5.

## 2. OVERVIEW OF THE STATE-OF-THE-ART

The estimation of yield is an approximation of the integral of the function that determines if the design specifications are met with the statistical process parameters varying over the unit cube [3]. The integration error can be separated into the factor related to the function itself and the factor related to the generated random point set according to the Koksma-Hlawka theorem [5], which is shown as:

$$|\hat{I} - I| \leq D_n^*(x_1, x_2, \dots, x_n) V_{HK}(f) \quad (1)$$

where  $\hat{I}$  and  $I$  are the estimated value and the real value of the integration.  $D_n^*$  is the star discrepancy which measures the uniformity of the generated points: more uniformly distributed samples have lower  $D_n^*$ .  $V_{HK}(f)$  is the total variation of  $f$  [6], which is determined by the problem itself.  $n$  is the number of samples ( $x_i$  is a  $d$ -dimensional vector). This separation (equation 1) leads to 2 categories of advanced MC simulation methods and their usage in circuit yield estimation:

- Variance-reduction methods (e.g. importance sampling (IS) [3,7], Latin Hypercube Sampling (LHS) [8]), which focus on decreasing  $V_{HK}(f)$ . In IS the good shifted distribution function is often circuit specific, which poses the challenge of generality. LHS [9,10] first creates equal slices in each dimension of the stochastic variable vector, and then selects random values within each slice for every coordinate. At last, by randomly matching up the coordinate values, a bunch of LHS samplings are constructed. Because of this stratification technique, the variance is reduced. LHS is a general method, but the performance is not always good enough, especially for some problems that are difficult to be decomposed into a sum of univariate functions [11]. LHS is shown to be worse than Quasi-Monte-Carlo (QMC) [1,12,13] in some circuit examples.
- Low-discrepancy sequence-based methods (e.g. QMC

[1,12,13]), which focus on decreasing  $D_n^*$ . QMC uses low-discrepancy sequences (LDS) to generate more uniformly distributed samples, which is a technique general to different circuits and showing good performance.

However, its major drawback is that for high-dimensional problems, the asymptotic advantage of the QMC point set appears to require an impractically large number of samples to set [11]. For instance, a 50-stochastic-variables case is typical in yield estimation of analog circuits, but using the Sobol' set with base 2 (standard), the advantage on the asymptotic rate of convergence can be expected after  $2^{127}$  samples with the available standard setting. If we use a lower number of samples, the first few dimensions are sampled uniformly, but higher dimensions are not, which degrades the performance. To relieve this problem, [1] presents a method, whose core idea is to sacrifice the non-uniformity in higher dimensions, but making the loss as small as possible. The way is by first ranking the sensitivities of the process variation variables and by selecting the important variables that mainly dominate the variance. They are mapped to the first few dimensions of LDS, which are more uniformly distributed. The theoretical background of this method is ANOVA decomposition and the concept of effective dimensions [14]. Experiments show a significant progress compared with PMC and LHS for medium-scaled (i.e. tens of stochastic variables) yield estimation problems [1]. However, even with this method, some important variation variables also have to use non-uniformly distributed samples in many cases. The degradation will be shown by the examples in section 4. The reason is that the upper limit on dimensions for LDS to keep the uniformity is typically 10-12 [15] if a reasonable number of samples can be used to maintain the efficiency in QMC. But the number of important stochastic variables (or effective dimensions) larger than this threshold can be frequently seen in analog circuits. Hence, a question still remains: What is the better solution when the number of effective dimensions (or important variables that dominate the variance) of the yield estimation is larger than the upper limit on dimensions for LDS (e.g. 12)?

To address this problem, our goal is to propose a LDS-based method for analog circuit yield estimation, called AYLeSS, which aims to:

- the method extends the limitation on the size of the yield estimation problems (i.e. large effective dimensions);
- achieve a 2 times speedup compared with ranking-integrated QMC (the best state-of-the-art method) for analog circuits;
- the method is easy to implement and is robust enough for different settings.

### 3. THE AYLeSS METHOD

#### 3.1 Basics of QMC

Because QMC is a part of the proposed AYLeSS method, QMC is briefly introduced first. QMC aims at generating more uniformly distributed samples, so as to decrease the estimation error of the yield integral. In PMC, the used uniformly distributed 'random numbers' generated by the computer are not truly uniform, and

the gaps arising among the samples adversely affect the uniformity. QMC, on the other hand, constructs a deterministic infinite sequence of  $d$ -dimensional points and selects a certain number of them when performing sampling. The goal of the constructed sequences is to fill the space as geometrically equidistant as possible. Such sequences are called low-discrepancy sequences (LDS). If the integrand has a bounded variation in the sense of Hardy and Krause [11], it is possible to construct a LDS along with

$$D_n^* = O(n^{-1}(\log n)^d) \quad (2)$$

which proves that one can achieve an asymptotic rate better than that of PMC which has a rate of  $n^{-1/2}$  [1]. There are different methods to generate a LDS, e.g. Halton set, Sobol' set, etc. In AYLeSS, we use the Sobol' set with a skip of  $2^{\lfloor \frac{\log n}{\log 2} \rfloor}$  points, because a LDS often has better performance when skipping the first few points [16]. The details of constructing the Sobol' set are in [6].

#### 3.2 The partitioning method in AYLeSS

To be able to handle high-dimensional problems, different from ranking-integrated QMC, our core idea is not to grant the degradation of QMC at high dimensions and minimize the loss, but to decrease the number of dimensions to avoid such loss. The method is partitioning the high dimensional problem into sub-groups with lower dimensions. In this way, generating a high-dimensional LDS is transformed to generate some groups of low-dimensional LDS, and in each group the uniformity can be kept with a reasonable number of samplings if the dimensionality of each group is not large.

However, the partition neglects the interactions between different sub-groups. Hence, the way of partitioning may affect the result. The best partitioning is to arrange variables that interact more strongly into a sub-group, and the interactions between different sub-groups should be as small as possible. The reason is that the interactions are considered within a sub-group, but not between separate sub-groups. This rule is quite intuitive. Suppose we divide the  $d$  dimensions into  $k$  groups, then each group has  $s$  dimensions, where  $d = k \times s$ . In the extreme case, different sub-groups have no interaction to each other, then the error is the sum of the errors in each sub-group. Hence, we can expect that the variance converges at the rate of a  $s$ -dimensional problem. Now, the problem becomes how to construct a set of easy to implement rules to obtain a good partitioning. In AYLeSS, the rules are constructed based both on the aspects of analog circuit design and statistics. The following rules are recommended by us:

- (1) The dimension of each sub-group should not be larger than 12 [15], because if the dimension of the sub-group is too large, the uniformity of LDS will also be sacrificed.
- (2) It is not wise to use too many sub-groups, because the fewer the number of sub-groups, the less interactions between different sub-groups need to be considered.
- (3) The number of dimensions of each sub-group is better as similar as possible, because the convergence rate is often determined by the sub-group with the highest dimension.
- (4) Devices whose widths and lengths have symmetry correlations (e.g. differential pairs) or have clear design relations interacting strongly (e.g. current mirror) when considering process variations

should be clustered to one sub-group.

(5) The intra-die variables of one transistor interacting with each other should be clustered to one sub-group.

According to these rules, the following partitioning method can be constructed:

Algorithm 1: the partitioning method

**Step 0:** Put the statistical inter-die variables into one sub-group. If the dimension of the sub-group is larger than 12 [15], partition them into 2 or more groups. The dimensions of each group should be as similar as possible.

**Step 1:** Find differential pairs and current mirrors in the circuit.

**Step 2:** Put the intra-die variables of each differential pair and current mirror into one cluster.

**Step 3:** Combine the clusters with small dimensions from Step 2 to sub-groups, whose dimensions should not be larger than 12. If the dimension of any cluster is larger than 12, split it into 2 or more sub-groups with dimensions less than or equal to 12, maintaining the intra-die variables of one transistor in a new sub-group. The dimensions of each group should be as similar as possible.

**Step 4:** For all other transistors that are not in differential pairs and current mirrors, partition them to different sub-groups. The dimensions should be as similar as the sub-groups in previous steps. The intra-die variables of one transistor should be in the same sub-group.

We now provide an example using the circuit in section 4.3 (Fig. 4). In this circuit, the differential pairs are: M1-M2, M3-M5, M7-M8, M9-M10, and M11-M12. The current mirrors are Mbn-M4-M11-M12 and Mbp-M5-M3. In the 0.35  $\mu m$  CMOS technology we used, there are 15 inter-die variables, and each transistor has 4 stochastic intra-die variables ( $V_{th}, T_{ox}, W_{eff}, L_{eff}$ , which are common to many technologies). According to Algorithm 1, the process variables can be divided as follows: group 1: inter-die variables (8 variables); group 2: inter-die variables (7 variables); group 3: intra-die variables of M1-M2 (8 variables); group 4: intra-die variables of M3-M5-Mbp (12 variables); group 5: intra-die variables of M7-M8 (8 variables); ...; group 8: intra-die variables of Mbn-M4 (8 variables). Through experiments, we found that if obeying these partitioning rules, AYLeSS performs up to 2 times speed up than ranking-integrated QMC.

This partitioning method is often applicable because: (1) To the best of our knowledge, most technologies use 4 intra-die variables ( $V_{th}, T_{ox}, W_{eff}, L_{eff}$ ). Even for a complex technology, the dimension of intra-die variables of a differential pair being larger than 12 can seldom be seen. Hence, the differential pairs, which have strong interaction, can often be clustered to one sub-group. (2) Even if the technology is very complex, we can still cluster the intra-die variables of one transistor in a sub-group. Experiments in section 4 still show much better results than the ranking-integrated QMC method when neglecting the differential pairs and only clustering the intra-die variables of a transistor to a sub-group.

In each sub-group, we use randomized QMC (RQMC) sets instead of QMC sets. QMC uses a determined LDS, and RQMC

adds scrambling to the QMC set by using random permutations to the digits of each coordinate value. The scrambling method used in AYLeSS is described in [17]. The benefit is that the variance of the yield estimation can decrease with the number of independent replications (number of sub-groups). The details are in [11].

### 3.3 Latin Supercube Sampling

Besides the core idea of partitioning, there exists a sampling method in the computational statistical field, called Latin Supercube Sampling (LSS) [11], which is even stronger. Not only does it hold the idea of partitioning to solve the high-dimensional QMC sampling problem, LSS also integrates the LHS method to further enhance the performance. The method to randomize the run order of the stratified samples of LHS [15] is used to the QMC sub-groups. Hence, the points in each group are obtained by randomly permuting the run order of the QMC points (the permutations of different groups are independent). Suppose  $x$  is a  $d$ -dimensional input sample.  $x$  is divided into  $k$  groups, where  $d = k \times s$  (in real applications, the dimensions of the sub-groups are not necessarily equal).  $x_i^j, i = 1, 2, \dots, n$  is an  $s$ -dimensional QMC point set ( $j = 1, 2, \dots, k$ ). The  $i^{\text{th}}$  LSS sample is:

$$x_{lss_i} = (x_{\pi_1(i)}^1, x_{\pi_2(i)}^2, \dots, x_{\pi_k(i)}^k), i = 1, 2, \dots, n \quad (3)$$

where  $\pi_j$  are independent uniform random permutations of  $1, 2, \dots, n$ . The purpose of this random permutation is the same as in LHS, i.e. to make the projection of each coordinate of the samples more uniform so as to reduce the variance (now coordinate refers to the sub-group). It has been proven that even with no partitioning rules, a poor grouping can still be expected to do equally well as LHS [11]. Therefore, the convergence rate of the AYLeSS method can always be better than LHS for complex and high dimensional circuits.

### 3.4 The framework of AYLeSS

In summary, the AYLeSS method works as follows:

Algorithm 2: the AYLeSS method

**Input:**  $d$ -dimensions of the variation variables  $x$ , sample size  $n$ , joint probability of the variation variables  $\Pi(x)$ .

**Step 0:** Skip  $2^{\lfloor \log n / \log 2 \rfloor}$  points of the Sobol' sequence.

**Step 1:** Partition the input variation variables into  $k$  groups with dimensions  $\{s_1, s_2, \dots, s_k\}$  according to algorithm 1.

**For EACH** sub-group:

**Step 2:** Scramble the Sobol' set according to the method in [17].

**Step 3:** Select the RQMC points according to the dimension of the sub-group.

**Step 4:** Perform a random permutation to the run order of the RQMC points according to eqn. (3) to obtain the LSS sample  $X_s$ .

**End**

**Step 5:** Generate the required samples by  $\Pi^{-1}(X_s)$  for circuit simulation.

## 4. EXPERIMENTAL RESULTS AND COMPARISONS

#### 4.1. Experimental Method

In this section, four designs are shown as examples with yield from 50% to 90% for two typical analog circuits in different technologies. In each example, the comparisons to ranking-integrated QMC, pure QMC, LHS and PMC are carried out. The same ranking method and the setting of 1000 initial MC samplings for ranking (counted in the total number of samples), are used as in [1]. For pure QMC, the LDS coordinates are assigned randomly. Experiments to verify the partitioning method are also performed. In data analysis, the confidence interval (e.g. 1% error compared to the true yield value) under a certain confidence level (e.g.  $2\sigma$ ) is used to reflect the estimation error. Two properties, the standard deviation convergence rate and the necessary sample size to obtain a certain accuracy, are selected as the comparison criteria, which is the same as [1]. We perform 10 runs to AYLeSS, QMC-based methods and LHS, 5 runs for the PMC method (because the computational cost is very large for PMC). From these data, the standard deviation for different numbers of samplings for each method can be obtained. The  $\sigma \propto n^{-\alpha}$  relationship is used and by linear fitting, via least square error, the corresponding convergence exponent,  $\alpha$ , can be estimated. By the same fitting method, the sample size needed for each method can also be estimated for a given level of accuracy (or estimation error) using the central limit theorem [18]:

$$\sigma \Phi^{-1}\left(\frac{1+p}{2}\right) \leq Y\left(\frac{\delta}{100}\right) \quad (4)$$

where  $\Phi$  is the standard normal cumulative distributed function,  $Y$  is the exact value of the yield (estimated by 500,000 PMC samples as substitute),  $\delta$  is the confidence interval and  $p$  is the confidence level. We use a  $2\sigma$  confidence level and a 1% confidence interval (the same as [1]) to compute the required  $\sigma$ . With a derived  $\sigma$  from eqn. (4), the required sample size,  $N_{\text{req}}$ , can be estimated by the fitted function.

#### 4.2. Test Example 1

The AYLeSS algorithm is first tested on a two-stage fully differential folded-cascode amplifier with common-mode feedback, shown in Fig. 1. The circuit is designed in a 90nm CMOS process with 1.2V power supply. The specifications are  $A_0 \geq 60\text{dB}$ ,  $\text{GBW} \geq 45\text{MHz}$ ,  $\text{PM} \geq 60^\circ$ , output swing  $\geq 1.9V$ , power  $\leq 2.5\text{mW}$  and  $\sqrt{\text{area}} \leq 250\mu\text{m}$ . The number of design variables is 21 and the variation parameters can be extracted from 52 standard normal-distributed random numbers for the selected circuit. Two designs with a yield of 84.29% and 90.39%, respectively, are shown as examples. Table 1 shows the results. The fitted lines in log10-log10 scale for different methods are shown in Fig. 2 and Fig. 3.

From the necessary number of samples ( $N_{\text{req}}$ ) columns in Table 1, two conclusions can be drawn. (1) It can be seen that the AYLeSS result is the best one compared with the other 4 methods for both designs, and can achieve a 2 to 2.5 times speed enhancement compared with ranking-integrated QMC, which is the current best state-of-the-art method, and more than a 5 times speedup compared with PMC. (2) If we decrease the 1000 samples (used

for ranking) for the ranking-integrated QMC method, we can see that the new  $N_{\text{req}}$  numbers, 1767 and 1187, are still worse than the AYLeSS results. This shows that the large number of effective dimensions degrades the QMC sampling even with ranking. If the number of effective dimensions is larger than the threshold (e.g. 12), some important dimensions cannot obtain uniformly distributed samples. In contrast, AYLeSS, does not have this problem and can often lower the scales by partitioning while receiving good results. From the convergence exponent ( $-\alpha$ ) column in Table 1, two more conclusions can be drawn. (1) The ranking is important for the QMC method, as the convergence rate considerably decreases if without ranking, which will further be shown by the next example. (2) Designs with 85%-99% yield are the most interesting ones to obtain an accurate estimation, because they may become a real product. However, designs with high yield need less MC samples, and the required number of samples for 50% yield is the largest [2,3]. From this relatively large-scale circuit, we can get a rough idea of the necessary number of samples for a yield larger than 85% for typical analog circuits when using LDS-based methods. Hence, the 1000 additional samples used for QMC ranking is expensive for these cases. In contrast, AYLeSS does not need ranking and the partitioning rules are easy to implement.

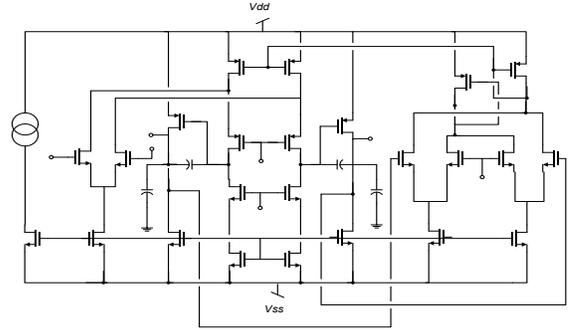


Fig. 1. Two-stage fully differential folded-cascode amplifier

Table 1. Results obtained by different methods (example 1)

Design 1	$N_{\text{req}}$	$-\alpha$	Design 2	$N_{\text{req}}$	$-\alpha$
AYLeSS	1265	-0.631	AYLeSS	807	-0.6236
QMC	3681	-0.4842	QMC	2148	-0.5202
Ranking	2767	-0.6816	Ranking	2187	-0.5729
QMC			QMC		
LHS	2448	-0.53	LHS	3687	-0.4963
PMC	6831	-0.5189	PMC	4307	-0.4904

In the following, we show the results obtained by LSS using a simplified partitioning method. In the five partitioning rules described in section 3.2, besides those related to the LSS sampling itself (e.g. requirements on dimension), there are two rules about analog circuits. In this example, we keep the intra-die variables of one transistor in a group (rule 5), but do not consider the symmetry (matching) devices (rule 4). The dimensions of all the sub-groups containing intra-die variables are 8. Three different groupings are used when combining the intra-die variables of transistors to form sub-groups. The results are shown in Table 2.

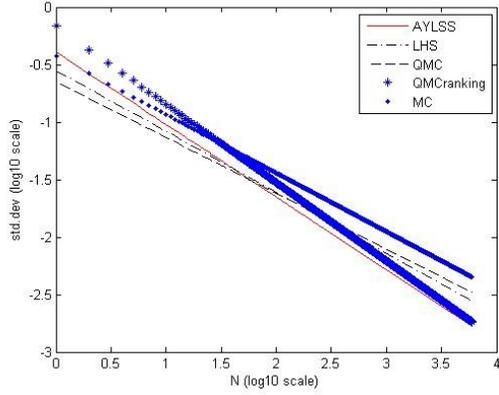


Fig. 2. Fitting of the convergence rates for different methods (design 1 of example 1)

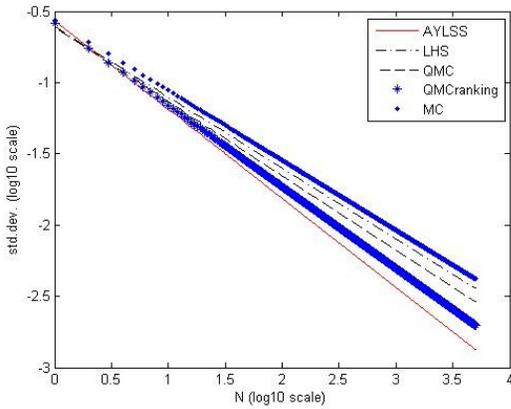


Fig. 3. Fitting of the convergence rates for different methods (design 2 of example 1)

Table 2. LSS results for different groupings

Design 1	$N_{req}$	$-\alpha$	Design 2	$N_{req}$	$-\alpha$
LSS1	1317	-0.6596	LSS1	822	-0.4518
LSS2	1313	-0.6280	LSS2	842	-0.6132
LSS3	1450	-0.6232	LSS3	914	-0.5908

From Table 2, we can see that: (1) The performances in Table 2 are lower than that of AYLeSS in Table 1. Hence, the partitioning rule of clustering the symmetry devices in one sub-group can enhance the performance. (2) Even if only clustering the intra-die variables of one transistor in a group, the result is still better than ranking-integrated QMC and LHS. (3) Different ways of partitioning all provide good results if the rule of clustering the intra-die variables of one transistor in a group is satisfied. Hence, AYLeSS is robust with respect to the partitioning used.

### 4.3 Test Example 2

The AYLeSS method is now tested on a fully differential folded-cascode amplifier, shown in Fig. 4, implemented in a  $0.35 \mu m$

CMOS process with 3.3V power supply. Although the circuit topology is simpler than example 1, the number of process variables is larger. The specifications are gain  $A_0 \geq 70dB$ ,

$GBW \geq 40MHz$ , phase margin  $PM \geq 60^\circ$ , differential output swing  $\geq 4.6V$  and power  $\leq 1mW$ . The number of design variables is 13 and the number of variation parameters is 75 uniformly-distributed random numbers in the normalized interval  $[0,1]$ . Two designs with a yield of 57.86% and 72.65%, respectively, are shown as examples. The results are shown in Table 3. The fitted lines in log10-log10 scale for the different methods are shown in Fig. 5 and Fig. 6.

Table 3. Results obtained by different methods (example 2)

Design 1	$N_{req}$	$-\alpha$	Design 2	$N_{req}$	$-\alpha$
AYLeSS	6251	-0.6053	AYLeSS	2884	-0.6332
QMC	12033	-0.4738	QMC	7320	-0.4921
Ranking	10052	-0.5246	Ranking	5710	-0.5457
QMC			QMC		
LHS	13210	-0.5117	LHS	5782	-0.5038
PMC	33617	-0.5211	PMC	20873	-0.5147

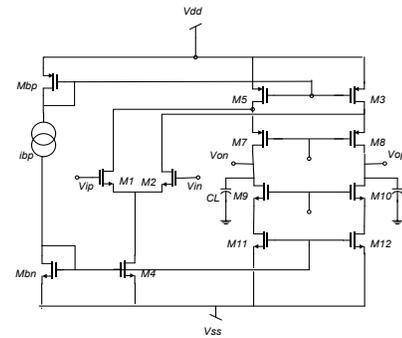


Fig. 4. Fully differential folded-cascode amplifier

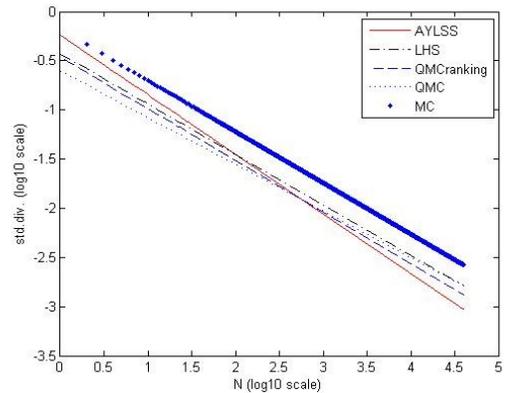


Fig. 5. Fitting of the convergence rates for different methods (design 1 of example 2)

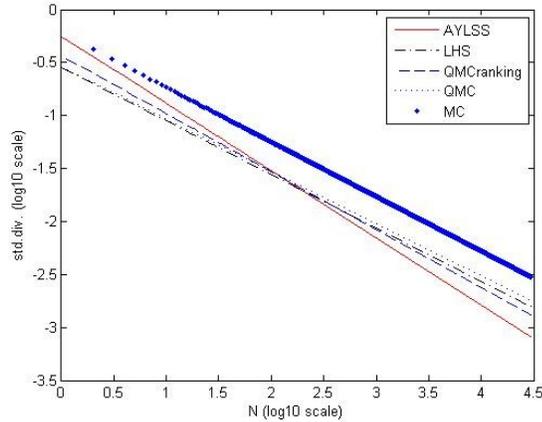


Fig. 6. Fitting of the convergence rates for different methods (design 2 of example 2)

From Table 3, it can be seen that the results of AYLSS achieve the smallest number of necessary samples and also the highest convergence rate. A 1.5 to 2 times speed enhancement can be achieved compared with ranking-integrated QMC and a more than 5 times speed-up compared with PMC.

We now show the results obtained by LSS using two different partitioning methods. The first partitioning keeps the intra-die variables of one transistor in a group (rule 5), but does not consider the symmetry devices (rule 4), which is the same as the last example. The second partitioning does not consider both rule 4 and rule 5. Only LSS is used, and the partitioning is assigned randomly.

Table 4. LSS results for different partitioning rules

Design 1	$N_{req}$	$-\alpha$	Design 2	$N_{req}$	$-\alpha$
LSS 1	6589	-0.6028	LSS 1	3004	-0.6102
LSS rand	7654	-0.5540	LSS rand	3966	-0.4907

From Table 4, besides the conclusions drawn from the last example, it can be seen that if we randomly partition the sub-groups in LSS, the performance is clearly lower than LSS 1, which uses rule 5 but not rule 4. This shows the significance of clustering the intra-die variables of one transistor in a group.

From Fig. 1 to Fig. 4, three of them show when the number of samples is small (the variance is large), the corresponding variance of QMC is better than that of AYLSS. However, when more samples are used and the estimation variance is reduced for practical use, AYLSS performs much better. This fact also shows the potential of AYLSS in high-sigma yield estimation.

## 5. CONCLUSIONS

In this paper, a new method, called AYLSS, has been proposed for the yield estimation of analog circuits. The method solves the limitation on the number of effective dimensions suffered by ranking-integrated QMC, which is the best state-of-the-art method. The key idea of AYLSS is the partitioning of the

process variation variables into sub-groups and using RQMC in each sub-group. The partitioning method for analog circuits has been proposed. It is effective, easy to implement and robust. Latin Supercube Sampling is used, which integrates the partitioning and the random permutation of LHS. Experimental results show that AYLSS can achieve a 1.5 to 2.5 times speed enhancement compared with ranking-integrated QMC and can handle circuits with many process variation parameters.

## 6. ACKNOWLEDGMENTS

This research was supported by a special bilateral agreement scholarship of Katholieke Universiteit Leuven, Belgium and Tsinghua University, P. R. China.

## 7. REFERENCES

- [1] A. Singhee et al., "Why Quasi-Monte Carlo is better than Monte Carlo or Latin Hypercube Sampling for Statistical Circuit Analysis", *IEEE TCAD*, pp. 1763-1776. 2010.
- [2] B. Liu et al., "Efficient and Accurate Statistical Analog Yield Optimization and Variation-Aware Circuit Sizing based on Computational Intelligence Techniques", *IEEE TCAD*, pp. 793-805. 2011.
- [3] H. Graeb, "Analog Design Centering and Sizing", Springer. 2007.
- [4] T. McConaghy et al., "Globally Reliable Variation-Aware Sizing of Analog Integrated Circuits via Response Surfaces and Structural Homotopy", *IEEE TCAD*, pp. 1627-1640. 2009.
- [5] E. Hlawka, "Functionen von beschränkter variation in der theori der gleichverteilung", *Annali di Matematica Pura ed Applicata*, pp. 325-333. 1961.
- [6] F. Hickernell, "A generalized discrepancy and quadrature error bound", *Mathematics of Computation*, pp. 299-322. 1998.
- [7] T. Doorn et al., "Importance sampling Monte Carlo simulations for accurate estimation of SRAM yield", *Proc. of ESSCIRC*, pp. 230-233. 2008.
- [8] M. Stein, "Large Sample Properties of Simulations Using Latin Hypercube Sampling," *Technometrics*, pp. 143-151. 1987.
- [9] M. Keramat et al., "Latin Hypercube Sampling Monte Carlo Estimation of Average Quality Index for Integrated Circuits", *Analog Integrated Circuits and Signal Processing*, pp.131-142. 1997.
- [10] A. Dharchoudhury et al., "Performance-constrained worst-case variability minimization of VLSI circuits", *Proc. of DAC*, pp. 154-158. 1993.
- [11] A. Owen et al., "Latin supercube sampling for very high dimensional simulations", *ACM Transactions on Modeling and Computer Simulation*, 8(1). 1998.
- [12] A. Singhee, et al., "Practical, fast Monte Carlo statistical static timing analysis: Why and how", *Proc. of ICCAD*, pp. 190-195. 2008.
- [13] A. Singhee, et al., *Novel Algorithms for Fast Statistical Analysis of Scaled Circuits*, Springer. 2009.
- [14] R. Caflisch et al., "Valuation of mortgage backed securities using Brownian bridges to reduce effective dimension", Technical Report of UCLA. 1997.
- [15] M. McKay et al., "A comparison of three methods for selecting values of input variables in the analysis of output from a computer code", *Technometrics* 21, pp. 239-245. 1979.
- [16] P. Acworth et al., "A comparison of some Monte Carlo and quasi-Monte Carlo techniques for option pricing", *Proc. of Monte Carlo and Quasi-Monte Carlo methods*, pp. 127. 1996.
- [17] J. Matousek, "On the L2 discrepancy for anchored boxes", *Journal of Complexity*, pp. 527-556. 1998.
- [18] H. Fischer, "A History of the Central Limit Theorem: From Classical to Modern Probability Theory", Springer. 2010.