Sequential Importance Sampling for Low-Probability and High-Dimensional SRAM Yield Analysis

Kentaro Katayama†, Shiho Hagiwara††, Hiroshi Tsutsui†, Hiroyuki Ochi†, Takashi Sato†
† Department of Communications and Computer Engineering, Kyoto University
Yoshida-hon-machi, Sakyo, Kyoto, 606-8501 Japan
E-mail: paper@easter.kuee.kyoto-u.ac.jp
†† Integrated Research Institute, Tokyo Institute of Technology
4259-R2-17 Nagatsuta, Midori-ku, Yokohama, 226-8503 Japan

Abstract*—In this paper, a significant acceleration of estimating low-failure rate in a high-dimensional SRAM yield analysis is achieved using sequential importance sampling. The proposed method systematically, autonomously, and adaptively explores failure region of interest, whereas all previous works needed to resort to brute-force search. Elimination of brute-force search and adaptive trial distribution significantly improves the efficiency of failure-rate estimation of hitherto unsolved high-dimensional cases wherein a lot of variation sources including threshold voltages, channel-length, carrier mobility, etc. are simultaneously considered. The proposed method is applicable to wide range of Monte Carlo simulation analyses dealing with high-dimensional problem of rare events. In SRAM yield estimation example, we achieved 10^6 times acceleration compared to a standard Monte Carlo simulation for a failure probability of 3×10^{-9} in a six-dimensional problem. The example of 24-dimensional analysis on which other methods are ineffective is also presented.

I. Introduction

In sub-100nm technology, within-chip variation has unignorable impact on circuit performance. The random variation caused by the fluctuations of number and placement of dopant atoms makes circuit performance less predictable. The smaller the gate area becomes, the more critical the random variation becomes [1]. The worst-case analysis [2] usually leads to too pessimistic result, thus statistical method is required to make yield estimations reasonably realistic.

Monte Carlo (MC) method is one of the most robust approaches to estimate the failure probability of SRAM cells on which random variations have a serious influence. The MC method is general, easy to implement, and achieves a good accuracy with enough number of samples. However, the number of required samples increases greatly when analyzing a rare event accurately. In general, SRAM cells have a good yield, so a method to accelerate the MC simulation with no loss of accuracy is strongly required.

Importance sampling (IS) [3] is an efficient method for rare event simulation. The IS is a variance reduction technique in which alternative distribution is used to obtain a greater number of rare event samples. Finding a good alternative function is a key for successful IS. To determine such good alternative distributions, several approaches have been proposed. In [4], the norm minimization technique is proposed. This is based on the theory of large deviations. They propose to shift the original distribution to the coordinate of minimum norm failure-sample found in separate preliminary analysis (Fig. 1), since that coordinate has the largest contribution to the yield estimation.

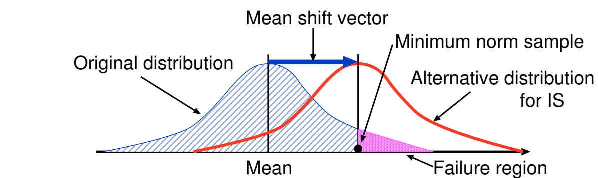


Fig. 1. Mean shift IS.

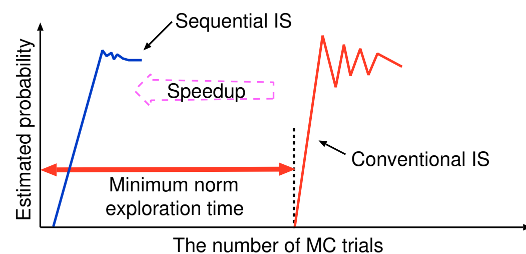


Fig. 2. Comparison of proposed and conventional Monte Carlo analyses.

The preliminary analysis to find the minimum norm is non-trivial and tends to require very large number of simulations particularly in high dimensional problems. Here, the dimension is the number of variables considered in the MC analysis. Because probability estimation can be started only after the minimum norm sample is determined, SRAM designers have to wait long until the shift vector is determined as well as IS is converged.

As illustrated in Fig. 2, a large number of MC trials are required for exploring the minimum norm sample. In case the preliminary analysis fails to find the failure sample of sufficiently small norm, convergence rate of the IS becomes very slow. In addition, there is another difficulty in finding the minimum norm sample in high dimensional cases. The parameter-variable space that needs to be explored enlarges exponentially to the number of dimensions. What makes the situation worse is that conventional approaches use uniform

978-1-4244-8192-7/10/$26.00 ©2010 IEEE

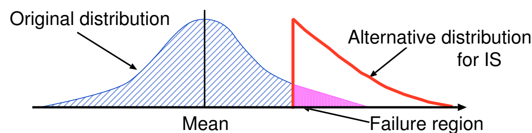


Fig. 3. IS with the optimal alternative distribution.

sampling in a hypercube region, which is intractable in high dimensional cases. In high dimensional problems, the number of samples (and thus time) required for finding minimum norm sample is significantly large. Reducing number of samples may shorten exploration time, but at the risk of poorly selected shift vector (i.e. the minimum norm sample of failure) candidate for IS, which significantly slows down IS convergence.

For the above reasons, existing methods are limited to problems has a small number of variables. [4], [5] can deal with only six variables, the threshold voltages of six SRAM transistors. [6] deals with 12-d analysis. [7] considers 403-dimensional case, but it reduces the number of variables to 11 using primary components analysis. Higher dimensional problems that simultaneously consider other sources of variability, such as channel-length, carrier mobility, and oxide thickness variations, were impossible to solve because of the difficulty in effectively determining failure regions and the minimum norm samples. With our proposed method using sequential importance sampling (SIS), the samples appropriately weighted according to the contribution to the total yield can be autonomously and efficiently generated. Thus, it eliminates lengthy pre-analysis and accelerates convergence as shown in Fig. 2.

Another issue of the existing IS is that the samples generated by an alternative distribution do not match well with the shape of the failure region. Theoretically, the optimal alternative distribution is the one that follows the failure distribution, as shown in Fig. 3 using one-dimensional example. Such a distribution accelerates the IS simulation most effectively [3]. This will never be satisfied by just shifting the original sample distribution, which is usually a Gaussian distribution. The proposed method, on the other hand, generates samples from the optimal distribution autonomously regardless of the shape of the failure region and the number of dimensions.

Furthermore, the existing work [4] determines the sampling regions based on the prior knowledge of circuit structure. In SRAM analysis example, circuit symmetry is utilized. Our method does not rely on any knowledge of circuit structure, thus can be used for truly general problems.

In this paper, we propose an application of sequential importance sampling (SIS) technique called *particle filter*, which resolves all the above problems. The proposed method is applicable to efficiently determining the shape of failure distribution and simultaneously estimates the failure probability. It also eliminates the pre-analysis so users are free from tweaking parameters required to obtain failure region samples having a small norm. In the proposed method, samples autonomously approximate unknown failure distributions. With obtained optimum samples as shown in Fig. 3, our method achieves several orders of acceleration and is applicable to high dimensional problems.

II. BACKGROUND

Our method, the sequential importance sampling (SIS), is based on IS. Thus, in this section, we briefly review standard MC and IS.

A. Monte Carlo method

The MC method is one of the most popular statistical methods to estimate a probability. With a large number of samples, estimation of the target probability becomes accurate. The principle of the MC method is explained as follows. Let us consider a target probability p which is the probability that random variable X falls into a certain (failure) region A. The probability p can be expressed as follows,

$$p = Pr(X \in A). \tag{1}$$

The MC method generates N samples of X, that is $X_i (i = 1, ..., N)$, to obtain $p_{\rm MC}$ which is the estimation of p, the true target probability. The estimated probability $p_{\rm MC}$ can be obtained by

$$p_{\rm MC} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{I}(X_i \in A), \tag{2}$$

where $\mathbf{I}(X_i \in A)$ is an indicator function defined by

$$\mathbf{I}(X_i \in A) = \begin{cases} 1 & (X_i \in A), \\ 0 & (X_i \notin A). \end{cases} \tag{3}$$

Figure of merit $\rho(p_{\rm MC})$ is given by

$$\rho(p_{\rm MC}) = \frac{\sqrt{{\rm Var}(p_{\rm MC})}}{p_{\rm MC}}, \tag{4}$$

where ${\rm Var}(p_{\rm MC})$ is variance of $p_{\rm MC}$ at the end of N-sample simulations. This is a barometer for the reliability of $p_{\rm MC}$. If $\rho(p_{\rm MC}) \leq \varepsilon\sqrt{\log_{10} 1/\delta}$, we can declare that the estimate of p has $(1-\varepsilon)100\,\%$ accuracy with confidence at least $(1-\delta)100\,\%$.

The required number of samples $N(\varepsilon, \delta)$ to obtain $(1-\varepsilon)100\,\%$ accuracy at $(1-\delta)100\,\%$ confidence is given by

$$N(\varepsilon, \delta) \approx \frac{\log_{10}(1/\delta)}{p\varepsilon^2}. \tag{5}$$

For example, to obtain accuracy of $90\,\%$ ($\varepsilon = 0.1$) and confidence of $90\,\%$ ($\delta = 0.1$), $\rho(p_{\rm MC}) \leq 0.1$ should be satisfied, and about $100/p_{\rm MC}$ samples are required. The standard MC requires a large number of samples when estimating a rare event.

B. Importance Sampling

The IS is a method to reduce the estimation variance of standard MC. Replacing the original sample distribution $X \sim f(X)$ by an alternative one $\tilde{X} \sim g(\tilde{X})$ in which $p(\tilde{X} \in A)$ is high, more samples fall in the failure region.

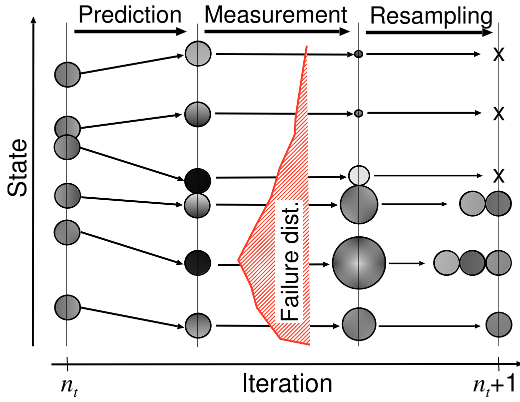


Fig. 4. Particle filter.

Reliable estimation can thus be obtained without increasing N. p_{IS}, the probability obtained by IS, is calculated as

$$p_{\mathrm{IS}} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{I}(\tilde{X}_i \in A)\omega(\tilde{X}_i), \tag{6}$$

where $\omega(\tilde{X}_i) = f(\tilde{X}_i)/g(\tilde{X}_i)$ is a weight function. The IS is a powerful method to accelerate the MC simulation as described in [4]. As stated earlier, to apply IS, determination of the alternative distribution is important.

III. Sequential Importance Sampling

In our method, the particle filter is applied to explore the failure region, and to simultaneously calculate the target probability using IS. After a few iterations of the procedure described in this section, failure probability can be obtained. With the use of autonomously determined samples that follow the failure region, significant speedup becomes possible.

A. Particle filter

The particle filter is one of SIS methods applicable to any state-space model [8]. It is based on a set of samples (called "particles") which explore the failure region. The particles can adaptively track the failure region regardless of the shape and probability distribution of the failure region. The procedure of the particle filter consists of three stages, (1) prediction stage, (2) measurement stage, and (3) resampling stage, as shown in Fig. 4. Each stage of the particle filter is summarized as follows.

(1) At the prediction stage, the next position of each particle is predicted by using its states in the past.
(2) At the measurement stage, the likelihood of each particle is calculated, and the weight of the particle is updated based on its likelihood.
(3) At the resampling stage, particles with higher weight are replicated and the ones with lower weight are eliminated. This is achieved by sampling with replacement considering the weights of the particles.

By iterating these steps, particles explore a given state-space to find regions of importance following the target probability distribution.

B. Proposed method

In our method, we initially place particles on an n-dimensional sphere having a large radius to find failure samples. Here, the central point stands for nominal state of parameters. We propose to use a uniform distribution on a spherical surface although any distributions can be used for initialization The objective of the initial particle placement is to comprehensively find failure regions. Once the failure region is found, the particles are gradually moved to the regions of more importance while repeating the above procedure as long as the failure region is contiguous. Thus, the initial particles does not have to have small norm. In case no initial particle falls into the failure region, we set the radius of sphere larger and explore again so that the particles are also moved to the failure region.

At the prediction stage, particles are moved based on the Gaussian distribution whose mean is the present position of each particle. We use the Gaussian distribution whose mean and sigma are X_i and σ_{pred}, respectively. The optimal variance of the prediction distribution will be discussed in the next section. When particles are at coordinates $X_i (i = 1, ..., N)$, the probability of a particle to lie at x by the prediction stage is given by the following,

$$m(x) = \frac{1}{N}\sum_{i=1}^{N} p(x|X_i). \tag{7}$$

where $p(x|X_i)$ is a conditional probability that X_i moves to x. Weight function $\omega(x)$ is calculated by

$$\omega(x) = \frac{p(x)}{m(x)}, \tag{8}$$

where the target probability $p(x)$ is estimated by Eq. (6).

At the measurement stage, the weights of particles are updated by the following,

$$W_i = \mathbf{I}(X_i \in \text{failure})p(X_i), \tag{9}$$

where $W_i (i = 1, ..., N)$ is the weight of a particle i, and $p(X_i)$ is the original probability. Based on these weights of particles, the resampling stage processes the particles.

In our method, we use running-average of estimated probability obtained in each predictions for overall probability estimation.

$$p_{\mathrm{ave}}(n_t) = \frac{1}{n_t}\sum_{j=1}^{n_t} p_{\mathrm{IS}}^{j}. \tag{10}$$

Here, n_t is the number of iterations and p_{IS}^{i} is obtained probability in j-th iteration step. We stop our algorithm when $\rho(p_{\mathrm{ave}}(n_t))$ becomes smaller than 0.1. At the beginning of analysis when particles are still moving to more important regions, which we call *burn-in* period, estimation tends to be far from true value. We start evaluating the convergence of running-average $\rho(p_{\mathrm{ave}}(n_t))$ after

$$0.5 \leq \frac{p_{\mathrm{ave}}(n_t + 1)}{p_{\mathrm{ave}}(n_t)} \leq 2.0 \tag{11}$$

is satisfied.

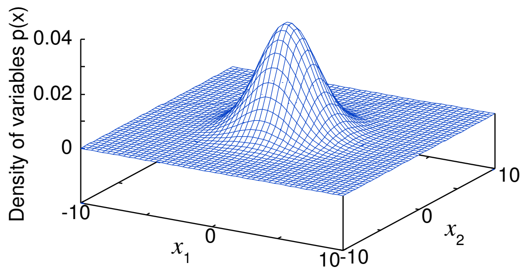


Fig. 5. Density function of variable $\mathbf{x}$.

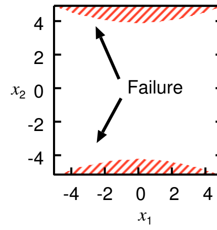


Fig. 6. Failure region.

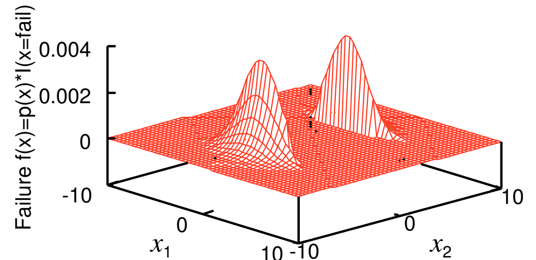


Fig. 7. Failure function $f(\mathbf{x})$.

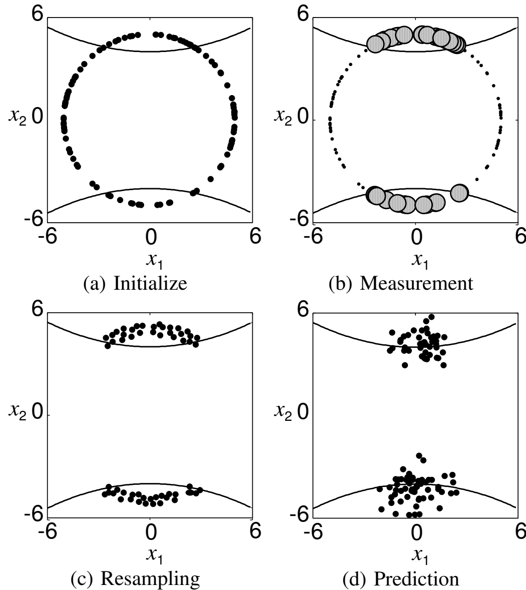


Fig. 8. Particle filter in 2-d analysis.

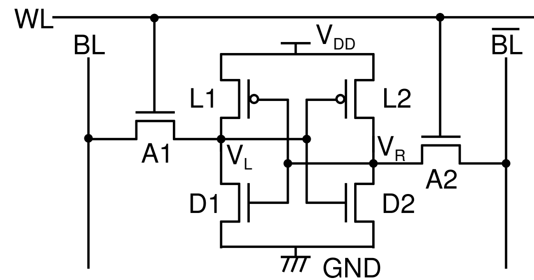


Fig. 9. SRAM circuit.

TABLE I
STANDARD DEVIATIONS (σ) OF DEVICE PARAMETERS.

	Load	Driver	Access
V_{th}	30 mV	22 mV	18 mV
L_{g}	1 nm	1 nm	1 nm
T_{ox}	0.042 nm	0.040 nm	0.040 nm
μ_0	5.4 mm^2/Vs	5.0 mm^2/Vs	5.0 mm^2/Vs

C. Example

A graphical example of how the proposed method using particle filter works in a two-dimensional problem is presented in Figs. 5 to 8. $\mathbf{x} = (x_1, x_2) \sim p(\mathbf{x})$ (illustrated in Fig. 5) is a two-dimensional variable and the failure region defined here is

$$x_2 \geq 0.04x_1^2 + 4 \quad \text{or} \quad x_2 \leq -0.04x_1^2 - 4, \tag{12}$$

as shown in Fig. 6. Combining Figs. 5 and 6, the failure function $f(\mathbf{x})$ is defined as follows,

$$f(\mathbf{x}) = \mathbf{I}(\mathbf{x} \in \text{failure})p(\mathbf{x}). \tag{13}$$

In this case, $f(\mathbf{x})$ looks like Fig. 7. This is the distribution that we want to obtain the probability. In other words, the failure probability is the integral (volume) of this function. In this example, $f(\mathbf{x})$ is also a weight function (Eq. (9)). Initially, we place particles on the n-dimensional sphere (circle in two-dimensional case) to seek the failure sample as shown in Fig. 8(a). At the measurement stage, we update the weight of particles by Eq. (9). The weights of particles in the failure region have non-zero value and those in the pass region drop to zero. In Fig. 8(b), the size of particles corresponds to their weight. At the resampling stage, we update the position of particles in proportion to the weights of the particles. More number of particles are distributed around the particle whose weight is large, and the particles whose weight is small eventually disappear as shown in Fig. 8(c). At the prediction stage, we move particles randomly as shown in Fig. 8(d) and calculate the failure probability by using Eqs. (7), (8), and (6). By iterating measurement, resampling, and prediction, we can successfully estimate failure probability.

IV. EXPERIMENTAL RESULTS

The proposed method is applied to estimate yield of a six-transistor SRAM cell shown in Fig. 9. The SRAM yield is defined by the read noise margin (RNM) [9]. Transistors L1 and L2 are loads, D1 and D2 are drivers, and A1 and A2 are access transistors, respectively. The 65-nm PTM model [10] is employed as a transistor model. A circuit simulator, SPICE, is used to calculate the RNM considering device-parameter variations.

In this experiment, we consider the following two cases.

1) The threshold voltages (V_{th}) of all transistors are the variables: the six-dimensional problem (6-d).
2) In addition to the above case, the channel-lengths (L_{g}), the oxide thicknesses (T_{ox}), and the carrier mobilities (μ_0) are varied (24-d).

The variables are considered to be all independent, and follow Gaussian distributions, although interactions between variables

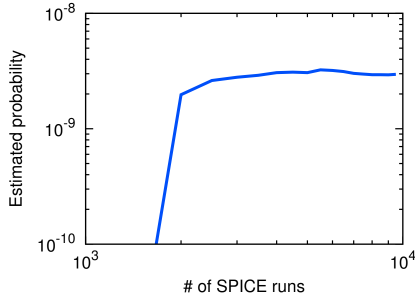


Fig. 10. Failure probability estimation for 6-d.

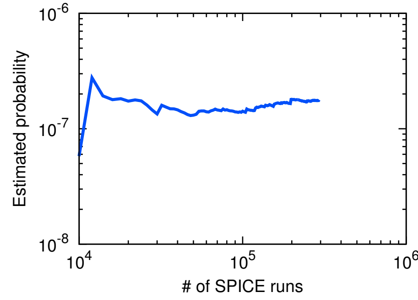


Fig. 11. Failure probability estimation for 24-d.

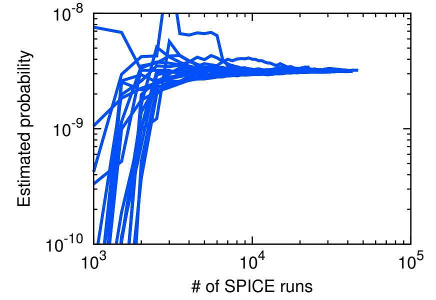


Fig. 12. Estimated probabilities of 20 trials.

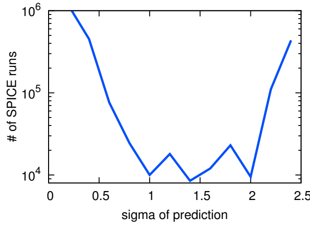


Fig. 13. $\sigma_{\rm pred}$ and the number of SPICE runs (6-d).

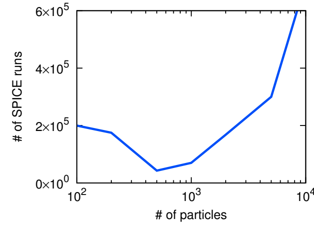
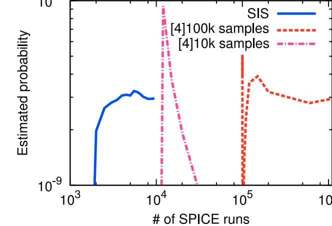
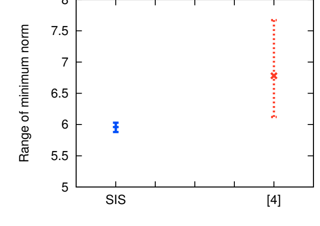


Fig. 14. The number of particles and the number of SPICE runs (6-d).



(a) Estimated probabilities.



(b) Minimum norm samples.

Fig. 15. Comparison with existing method [4].

(such as channel-length and threshold voltages) are automatically considered in the circuit simulation.

Standard deviations (σ) of device parameters used in our experiments are shown in Table I. The gate length is 65 nm for all transistors. The oxide thickness is 1.95 and 1.85 nm for load and the other transistors, respectively. The carrier mobility is 57.4 and 49.1 mm^2/Vs for the load transistors and the others, respectively. The supply voltage $V_{\rm DD}$ is 1.0 V.

A. Read failure probability in various dimensions

The read noise margin (RNM) is a measure for evaluating read stabilities of an SRAM cell [9]. The RNM is the maximum noise level under which the read operation succeeds. In sub-100 nm process, small RNM due to various device variations is a serious concern for optimizing an SRAM cell. The RNM is calculated by measuring the side of the largest square that can be inscribed between voltage transfer curves of $V_{\rm R}$ and $V_{\rm L}$ using DC analysis. When the RNM of a cell is negative, the cell is considered as read-failure. The proposed method is applied to estimate failure rate of the SRAM cell under device parameter variations of different dimensions.

Figs. 10 and 11 show estimated failure probabilities using the proposed method for 6-d and 24-d cases. The number of particles is 500 and 2,000 for 6-d and 24-d cases, respectively. The x-coordinate of the left-most point in the graph shows the number of initial particles. We stop our algorithm when $\rho(p_{\rm ave}(t)) < 0.1$. This means that the algorithm stops when accuracy of 90 % ($\varepsilon = 0.1$) and confidence of 90 % ($\delta = 0.1$) is achieved.

At the beginning of the analysis, short burn-in period is observed. After a few iterations when the particles are settled, a good estimation of the very low failure probabilities is obtained. In particular, in the 6-d case, the probability is as low as the order of 10^{-9}. With our method, very good estimation is obtained by 10^4 SPICE-runs. This is a significant improvement compared with standard MC, whose analysis requires more than 10^{10} SPICE-runs. Our analysis is already 6 orders magnitude faster than the standard MC in the 6-d case. The speedup is even larger for higher dimensional cases, while not even a run-time estimation is possible for standard MC.

As expected, the failure probability increases as we increase the number of variables, as shown in Fig. 11. This is quite understandable because adding variation sources deteriorates the noise margin. Even in the high-dimensional cases, the proposed method achieves very stable and thus successful estimation with much less than 10^6 SPICE-runs.

B. Accuracy of the proposed method

In order to evaluate the stability and accuracy of our method, we repeated the estimation in 6-d case for 20 times. The result is shown in Fig. 12. In this experiment, the number of particles N and the sigma used in the prediction stage $\sigma_{\rm pred}$ is set 500 and 1.0σ, respectively, where σ is the device parameter variation. We stopped each trial when $\rho(p_{\rm ave}(t))$ is less than 0.1. In each trial, estimated probability is about 3.16×10^{-9}. When the number of SPICE-runs is not enough large, estimated probability has a variation. However, it can be seen that the accuracy could be improved by increasing the number of SPICE-runs. Note that the number of SPICE-runs is in proportion of the number of iterations and particles. In this case, obtained probability differs by at most 10 % from the average. This result corresponds with the stop criterion.

C. Optimization of our method

In the proposed method, the number of samples for estimation relies on the parameters of the particle filter — particularly on the choice of prediction function. In Fig. 13, the relationship between $\sigma_{\rm pred}$, sigma of Gaussian distribution

used in the prediction stage, and the number of SPICE-runs is shown. The number of particles is again 500. If $\sigma_{\rm pred}$ is too low, particles move slowly. In other words, particles spend a lot of simulations for exploring the failure region. If $\sigma_{\rm pred}$ is too large, $m(x)$ in Eq. (7) greatly varies so failure probability obtained in each prediction stage fluctuates. According to Fig. 13, the choice of $\sigma_{\rm pred}$ is relatively insensitive to the total number of circuit simulations at around 1.5σ, The automatic determination of $\sigma_{\rm pred}$ is one of our future works.

The number of particles also affects the estimation efficiency. The relationship between the number of particles and the number of SPICE-runs when $\sigma_{\rm pred} = 1.0\sigma$ is shown in Fig. 14. When the number of particles is small, IS returns unreliable values because of insufficient number of random samples. As a result, running-average of estimated probability requires many samples for convergence. When the number of particles is large, IS returns reliable values so the number of iterations for convergence can be small. Regardless of it, however, the number of required samples increases when particles are too many. So there is an optimal point in number of particles. Compared with the effect of $\sigma_{\rm pred}$, the number of particles is less insensitive to the total number of circuit simulations. The automatic control of the number of particles is another future work.

D. Comparison with existing method

In Fig. 15(a), the number of SPICE simulations to estimate probability in 6-d case is compared with the existing method. The dashed-dotted line and dashed line are the results of the existing norm minimization method [4], where 10,000 and 100,000 samples are used in the pre-analysis, respectively. In the case of 10,000 samples (dashed-dotted line), due to very slow convergence caused by the sub-optimal shift-vector selection, the failure probability lies mostly out of the plotted range.

The existing method [4] has difficulty in finding a good minimum norm sample in their pre-analysis when the number of samples are small. In this method, the hypercube is sampled by uniform distribution, which becomes significantly inefficient when estimating low probability cases. Increasing the dimensions also exponentially enlarges the exploration space, thus the pre-analysis of the norm-minimization technique is inapplicable to the cases larger than 6-d. The 6-d is the only case that we could compare convergence rate. As can be seen in the figure, our proposed method finishes much earlier than the norm minimization starts IS. The time required for the pre-analysis in norm minimization technique is more than enough to obtain good estimation in our method.

This fact can be observed from a different angle. In Fig. 15(b), we compare the minimum norm obtained in our method, and the existing method. The particles (number of total SPICE-runs) in our method and the samples for pre-analysis of norm minimization are equivalent (500) for fair comparison. The graph shows the maximum, average, and minimum of 30 trials of the minimum norm sample found in the respective analyses. Our method always finds a good minimum norm sample, which shows the particles appropriately track important failure region, although finding the minimum norm is not the primary objective of our method. In the case of the existing method, it is impossible to judge whether a good alternative distribution for IS is chosen. In our method, alternative distribution changes in each iteration and we calculate the running-average of the probability estimated by each distribution. Appropriateness of the alternative distribution is indirectly observed by the convergence of failure probability.

V. Conclusions

We proposed an SIS method for accelerating the MC simulations. This method is applicable for yield estimation to low-probability and high-dimensional analyses. The particle filter autonomously detects the failure region and particles track the shape of failure distribution, so that IS becomes most efficient. Our method is verified through experimental analysis of SRAM yield. In the low-probability failure analysis of 10^{-9} orders, over $10^{6}\times$ speedup compared to standard MC method has been achieved. Higher dimension problems up to 24 dimensional problems are also efficiently solved.

Acknowledgement

This work has been partly supported by Semiconductor Technology Academic Research Center (STARC), and by KAKENHI Grant-in-Aid for Scientific Research (B) 22360143 from JSPS. This work is also partly supported by VDEC in collaboration with Synopsys, Inc.

References

[1] K. Takeuchi, T. Tatsumi, and A. Furukawa, “Channel engineering for the reduction of random-dopant-placement-induced threshold voltage fluctuation,” in *Tech. Dig. of IEDM ’97*, Dec. 1997, pp. 841–844.

[2] Y. Tsukamoto, K. Nii, S. Imaoka, Y. Oda, S. Ohbayashi, T. Yoshizawa, H. Makino, K. Ishibashi, and H. Shinohara, “Worst-case analysis to obtain stable read/write DC margin of high density 6T-SRAM-array with local Vth variability,” in *Proc. of ICCAD 2005*, Nov. 2005, pp. 398–405.

[3] P. H. Borcherds, “Importance sampling: an illustrative introduction,” *European Journal of Physics*, vol. 21, no. 5, pp. 405–411, 2000.

[4] L. Dolecek, M. Qazi, D. Shah, and A. Chandrakasan, “Breaking the simulation barrier: SRAM evaluation through norm minimization,” in *Proc. of ICCAD 2008*, Nov. 2008, pp. 322–329.

[5] 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,” in *Proc. of DAC 2006*, Jul. 2006, pp. 69–72.

[6] 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. of DATE 2010*, Mar. 2010, pp. 801–806.

[7] A. Singhee and R. A. Rutenbar, “Statistical blockade: A novel method for very fast monte carlo simulation of rare circuit events, and its application,” in *Proc. of DATE 2007*, Apr. 2007, pp. 1–6.

[8] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, “A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking,” *IEEE Trans. on Signal Processing*, vol. 50, no. 2, pp. 174–188, Feb. 2002.

[9] K. Agarwal and S. Nassif, “The impact of random device variation on SRAM cell stability in sub-90-nm CMOS technologies,” *IEEE Trans. on Very Large Scale Integration Systems*, vol. 16, no. 1, pp. 86–97, Jan. 2008.

[10] W. Zhao and Y. Cao, “New generation of predictive technology model for sub-45nm design exploration,” in *Proc. of ISQED’06*, Mar. 2006, pp. 585–590.