Worst-Case Analysis and Optimization of VLSI Circuit Performances

A. Dharchoudhury, Student Member, IEEE and S. M. Kang, Fellow, IEEE

Abstract

In this paper, we present a new approach for realistic worst-case analysis of VLSI circuit performances and a novel methodology for circuit performance optimization. Circuit performance measures are modeled as response surfaces of the designable and uncontrollable (noise) parameters. Worst-case analysis proceeds by first computing the worst-case circuit performance value and then determining the worst-case noise parameter values by solving a nonlinear programming problem. A new circuit optimization technique is developed to find an optimal design point at which all of the circuit specifications are met under worst-case conditions. This worst-case design optimization method is formulated as a constrained multi-criteria optimization. The methodologies described in this paper are applied to several VLSI circuits to demonstrate their accuracy and efficiency.

Keywords

Worst-case analysis, worst-case design optimization.

I. Introduction

INEVITABLE fluctuations in the manufacturing processes and environmental operating conditions of integrated circuits cause circuit parameters to vary randomly about their nominal target values. In order to distinguish the controllable circuit parameter values from the uncontrollable statistical variations, a circuit parameter can be described as the sum of a *designable component* and a *noise component*. The designable component of a parameter can be set to a particular value by the circuit designer and is considered to be a deterministic quantity. On the other hand, the noise component represents the statistical variation about the designable component and is treated as a random variable. The designable components of all the circuit parameters are henceforth called *designable parameters* and all the noise components are called *noise parameters*. Examples of designable parameters include nominal values of channel lengths and widths of MOS devices, nominal resistor and capacitor values, etc. Some typical examples of noise parameters for MOS circuits are the gate-oxide thickness, threshold voltage, channel length reduction, and channel width reduction of MOS devices.

Due to the random nature of noise parameters, circuit performances (such as gain or delay) vary randomly about their nominal values. A circuit design should therefore be optimized for parametric yield so that the majority of manufactured circuits meet the performance specifications. The computational cost and complexity of yield estimation, coupled with the iterative nature of the design process makes yield maximization computationally prohibitive. As a result, circuit designs are usually verified using worst-case models corresponding to a set of *worst-case conditions* of the noise parameters [1],[2]. If the circuit performance is acceptable under these worst-case conditions, then the circuit will have high parametric yield. *Worst-case analysis* refers to the process of determining the values of the noise parameters in these worst-case conditions and the corresponding worst-case circuit performance values. Worst-case analysis is very efficient in terms of computational time and designer effort, and thus has become the most widely-practised technique for statistical analysis and verification. However, it has been long recognized that conventional worst-case analysis approaches suffer from a number of shortcomings. Most notably, these methods are too pessimistic and lead to extremely conservative designs. In fact, design bottlenecks have been caused by too pessimistic simulation results due to inaccurate worst-case models. In the first part of this paper, we present a new approach for realistic worst-case analysis of IC performances.

In the second part of this paper, we present a novel performance optimization technique using the worst-case circuit performances. Most performance optimization approaches are based on nominal values of the performance objectives [1]. Since statistical variations are not considered, the final design obtained from such nominal design optimization is often not well centered and has low parametric yield. The *worst-case design optimization* technique presented in this paper optimizes the worst-case values of the performance objectives. At the final point obtained from worst-case design optimization, the parametric yield of the circuit is guaranteed to be high since the specifications are met under worst-case conditions.

The rest of this paper is organized as follows. In Section II, we describe the response surface modeling of circuit performances. In Section III, we present the formulation and solution of the worst-case analysis problem, and provide a few demonstrative examples. Section IV describes the worst-case design optimization technique, along with application examples. Concluding remarks and suggestions for future work are provided in Section V.

The authors are with the Department of Electrical and Computer Engineering, University of Illinois, Urbana, IL 61801.

II. Circuit Performance Modeling

The functional dependence of circuit performances on the designable and noise parameters are known only implicitly through the circuit equations. Thus, for particular values of the designable and noise parameters, the circuit performance values are computed by using a circuit simulator. In this work, each circuit performance is explicitly modeled as a *response surface* of the designable and noise parameters. This *response surface model* (RSM) is then used as a surrogate to the circuit simulator to provide inexpensive yet accurate estimates of the circuit performance value. This section defines the notation used throughout the paper and describes the procedure used to construct these response surface models.

A. Circuit Parameter Transformations

The designable and noise parameters usually have different units and their values differ by orders of magnitude. In order to avoid potential numerical inaccuracies, and to perform the statistical analyses with simpler variables, the designable and noise parameters are transformed as explained below. The original deterministic designable parameters d_i, $i = 1, \ldots, n_d$, are normalized such that all of the transformed parameters are of the same order. The transformed designable parameters are denoted by the vector $\mathbf{d}' \in \mathcal{D} \subset \mathcal{R}^{n_d}$, where $\mathcal{D}$ is the designable parameter space representing the allowed values of the transformed designable parameters. The noise parameters are treated as *correlated Gaussian* random variables whose means, standard deviations and correlation coefficients can be obtained from process measurements. The n_u noise parameters are denoted by the Gaussian random vector $\mathbf{u}$. These are transformed into a set of *independent zero-mean Gaussian* random variables denoted by the vector $\mathbf{v}$. This transformation, $\mathbf{v} = \mathbf{T}(\mathbf{u})$, is a linear transformation based on the eigen-decomposition of the positive definite covariance matrix of $\mathbf{u}$. The *joint probability density function* (*jpdf*) of the transformed noise parameters is given by

$$f(\mathbf{v}) = \frac{1}{(\sqrt{2\pi})^{n_u}} \exp(-\frac{1}{2}\mathbf{v}^T \Omega^{-1} \mathbf{v}), \tag{1}$$

where Ω is the (diagonal) covariance matrix of $\mathbf{v}$. The contours of constant probability density of $\mathbf{v}$ are hyperspheres centered at the origin.

The assumption that the noise parameters have a Gaussian distribution is frequently used both in literature as well as in practice and it has two advantages: the independence transformation is simple and most sampling (experimental design) schemes favor this distribution. If the noise parameters are known to be non-Gaussian, several transformations (such as square root, log, etc.) can be tried to make the distribution approximately Gaussian. As will be shown later, the techniques presented in this paper can be applied without any changes even if the Gaussian assumption is not valid and the above transformations fail. Under such circumstances, however, the sampling scheme chosen for response surface modeling should be able to handle non-normal and correlated random variables. Latin hypercube sampling, used in this paper, is one such scheme.

B. Experimental Design and Response Surface Modeling

As noted earlier, a circuit performance measure is a function of the designable parameters $\mathbf{d}$ and the noise parameters $\mathbf{u}$. Since the functional dependence is generally implicit, the evaluation of performance values for a given set of designable and noise parameter values entails at least one circuit simulation. The primary objective of circuit performance modeling is to obtain inexpensive and accurate estimates of the performance values in lieu of expensive circuit simulations. The savings provided by these models can, however, be offset if the construction of these performance models from simulation data is itself very expensive. *Experimental designs* [3] and *response surface modeling* [4] provide a systematic and computationally attractive means of obtaining the circuit performance models.

For statistical circuit design, various performance modeling approaches have been proposed in the past. Yu et al. [5] adopted the assumption that all MOSFET model parameters can be obtained in terms of four critical parameters (channel length reduction, channel width reduction, flat-band voltage, and gate-oxide thickness) [6], and only four noise parameters were considered in their work. An average mean-squared error criterion was used to select the experimental design. In [7], central composite designs are used in conjunction with regression analysis to build macromodels for IC fabrication process design. In central composite design, the input parameters are set to a few quantized levels only. Bernardo et al. [8] model the outputs of a circuit simulator as realizations of a stochastic process and use a sequential circuit design strategy. The model is required to be accurate over a large region of the parameter space. Moreover, this approach is very expensive since the model coefficients are estimated using the maximum likelihood principle.

In order to approximate a circuit performance using a pre-assumed function of the *transformed circuit parameters* $x_1, x_2, \ldots, x_n$, the circuit is simulated at some *training points* in the $\mathbf{x}$-space (here $\mathbf{x}$ denotes $\mathbf{d}'$ and/or $\mathbf{v}$). Experimental designs provide a systematic means of selecting these training points. Latin Hypercube Sampling (LHS) [9] is used as the experimental design technique in this work. LHS provides a much more "uniform" coverage of the input parameter space (*space filling*) than other experimental design techniques such as fractional factorial or central composite designs.

Moreover, a sample of any size can be easily generated, and all types of probability densities are handled. A desired sample rank correlation structure can also be built into the sample.

The circuit is simulated at the training points generated by LHS, and the values of the circuit performance measures are extracted automatically from the simulation outputs. In this paper, two methods are used to fit polynomial response surfaces to the extracted data: (i) linear least squares regression is used to fit linear or quadratic RSMs [4], and (ii) the Maximally Flat Quadratic Interpolation (MFQI) technique [10] is used to fit quadratic RSMs. The regression method requires at least as many runs as the number of unknown coefficients in the model, whereas the MFQI method can be used to construct a quadratic RSM with fewer runs than the number of unknown coefficients. The role of a RSM is as a performance evaluator only; the techniques presented in this paper can be used in conjunction with any other modeling technique or model structure. Further, no assumption is made regarding the criticality of certain MOSFET parameters.

III. Worst-Case Analysis

At any design point, uncontrollable fluctuations in the noise parameters cause circuit performances to deviate from their nominal target values. *Worst-case circuit simulation* is used to verify whether circuit performances are acceptable under worst-case conditions of the noise parameters. The goal of *worst-case analysis* is to determine these worst-case circuit performance values and the corresponding noise parameter conditions. The reliability of the worst-case design optimization method, described in Section IV, greatly depends on the accuracy of the results of the worst-case analysis procedure. Hence, it is extremely important that the worst-case analysis method yield *realistic* estimates of the worst-case noise parameter sets and the worst-case performance values.

A. Previous Approaches

The "best"- and "worst"-case analyses approach have been used heavily in the past, since they are least expensive in terms of computational cost and designer effort, and provide high parametric yields. The traditional worst-case analyses methods, however, suffer from a number of shortcomings. The most common approach is the so-called "corners" or the *one-at-a-time* technique. In this approach, each noise parameter value u_i that leads to the worst performance is chosen independently. Typically, the chosen value is the mean of $u_i \pm 2$ or 3 standard deviations. This method ignores the correlations among the noise parameters, and the simultaneous setting of each noise parameter to its extreme value results in simulation at the tails of the joint probability density of the noise parameters. Thus, the worst-case performance values obtained are extremely pessimistic [11]. Another approach to worst-case analysis is to consider the problem at the process level [2]. This method uses statistically independent process parameters as inputs to a process and circuit simulation system to estimate the worst-case performance. The application of this method to VLSI design is difficult due to the high dimensionality of the process parameter space and the high cost associated with process simulation. Müller [12] proposed the *limit parameters* approach to worst-case analysis, in which the worst-case vector of noise parameters is defined as that which causes the performance to be the worst and has the smallest probabilistic distance from the nominal noise parameter vector. A major shortcoming of this approach is that the circuit performances are assumed to be linear with respect to the noise parameters. Moreover, the sensitivities computed at the nominal noise parameter vector to obtain the coefficients of the linear model may not be accurate away from the nominal values where the worst-case conditions are expected to be. Further, the interaction between the noise parameters and designable parameters is not considered.

In our proposed method, the response surface of each circuit performance is used to obtain realistic worst-case noise parameter values for the particular performance. Circuit performances are functions of both the designable and noise parameters, and thus the worst-case noise parameter values should depend on the values of the designable parameters. Moreover, the worst-case conditions are usually different for different performance measures of a circuit. Both of these statements will be corroborated by experimental evidence in Section III-C.

B. Worst-Case Analysis Formulation

Since worst-case analysis is done one performance at a time, in this section, we will use r to denote any of the performance measures of interest. In order to quantify the concept of "worst", we note that r is a random variable. Therefore, there exists a *cumulative distribution function* $F(a) = \text{Prob}(r \leq a)$, which gives the probability that the performance values are less than or equal to a. Next, we define the *worst-case direction* w for the performance r as follows: w is equal to +1 if larger values are worse than smaller values of r, and is equal to -1 for the opposite case. For example, the worst-case direction for a performance such as delay will be +1, while that for gain will be -1. Given a *worst-case probability* of p, the worst-case value of r, r^{wc}, is defined as the value for which

$$p = \begin{cases} \Pr(r \geq r^{wc}) = 1 - F(r^{wc}), & \text{if } w = +1 \\ \Pr(r \leq r^{wc}) = F(r^{wc}), & \text{if } w = -1 \end{cases} . \tag{2}$$

In other words, the probability that the performance values will be worse than r^{wc} is equal to p.

At a given design point, a circuit performance is an implicit function of the noise parameters alone. As explained in the previous section, this function is replaced by the response surface model (RSM) of r in terms of the transformed noise parameters, and is denoted by $\hat{h}(\mathbf{v})$. Since the noise parameters are multivariate Gaussian, the distribution of r will also be Gaussian if the above function is linear, but this is not true in the nonlinear case. The computation of the worst-case value of the circuit performance for the Gaussian and non-Gaussian cases are discussed below.

Gaussian Case

If r is Gaussian, its cumulative distribution function $F()$ can be characterized by estimating its mean and variance only. This is done as follows: (i) a Monte Carlo (MC) [13] sample of the transformed noise parameters is drawn from the *jpdf* $f(\mathbf{v})$ and the RSM $\hat{h}()$ is used to compute the values of r, (ii) the sample mean and sample variance are used as estimates for the mean μ_r and the variance σ_r^2, respectively. Using the definition in (2), the worst-case value r^{wc} can be obtained for the Gaussian case as

$$r^{wc} = \mu_r + kw\sigma_r. \tag{3}$$

In the above equation, the constant k is related to the worst-case probability p as $k = \Phi^{-1}(1-p)$, where Φ^{-1} is the inverse standard Gaussian cumulative distribution function. For example, a typical value of k is 3, which corresponds to $p = 0.002$. Note that w, the worst-case direction of r, is chosen based solely on the nature of the performance (i.e. whether smaller or larger values are more desirable) and does not depend on the its relationship with the noise parameters. w is used to compute r^{wc} only, and is not used to determine the worst-case values of the noise parameters. In particular, if $w = +1$ results in the worst-case performance value, $w = -1$ will result in the best value for that performance and correspondingly "best-case" analysis.

Non-Gaussian Case

If r is not Gaussian and the Gaussian approximation is not accurate, then (3) cannot be used to compute r^{wc}. In this case, we first numerically estimate the cumulative distribution function $F()$ using classical Monte Carlo techniques [13]. Since a large number of performance evaluations are required in this step, the RSM is used instead of actual circuit simulations. Next, given the value of p and the estimated $F()$, we solve (2) for r^{wc} using the Newton-Raphson method. This approach is computationally quite efficient since RSMs are used in the Monte Carlo estimation, and the Newton Raphson procedure requires very few iterations since $F()$ is a monotonically non-decreasing function.

Next, to determine the worst-case vector of transformed noise parameters, $\mathbf{v}^{wc}$, we note that it is the solution to the following equation:

$$\hat{h}(\mathbf{v}) = r^{wc}. \tag{4}$$

The above equation, however, defines a hypersurface of possible values of $\mathbf{v}^{wc}$. The worst-case noise vector is chosen as the *most probable* solution of (4), since this vector is the most frequently occurring combination of noise parameter values that causes the performance to be the worst. In other words, $\mathbf{v}^{wc}$ is the solution to the following nonlinear programming problem where the objective is to maximize the joint probability density function $f(\mathbf{v})$ subject to the constraint that the noise parameters belong to the hypersurface (4):

$$\mathbf{NLP1}: \quad \max_{\mathbf{v}} \ f(\mathbf{v}) \ \text{ s.t. } \ \hat{h}(\mathbf{v}) = r^{wc}. \tag{5}$$

Since the noise parameters have been assumed to be multivariate Gaussian, (1) and (5) imply that the above nonlinear programming problem is equivalent to

$$\mathbf{NLP2}: \quad \min_{\mathbf{v}} \ \mathbf{v}^T\Omega^{-1}\mathbf{v} \ \text{ s.t. } \ \hat{h}(\mathbf{v}) = r^{wc}. \tag{6}$$

In NLP2, the objective function $\mathbf{v}^T\Omega^{-1}\mathbf{v}$ can be interpreted as the *probabilistic distance* of $\mathbf{v}$ from the origin (which is the mean vector for $\mathbf{v}$). This definition of the worst-case noise parameter vector is illustrated in Fig. 1 for $n_u = 2$. The level curves of the objective function are shown as concentric circles and the response surface is tangential to a level curve at the worst-case point.

If $\hat{h}(.)$ is a linear response surface given by

$$\hat{h}(\mathbf{v}) = a + \mathbf{b}^T\mathbf{v}, \tag{7}$$

then NLP2 becomes a quadratic programming problem, for which an analytical solution can be derived as

$$\mathbf{v}^{wc} = \frac{r^{wc} - a}{\mathbf{b}^T\Omega\mathbf{b}}\Omega\mathbf{b}. \tag{8}$$

For the case of nonlinear response surfaces, however, no analytical solution can be found and a nonlinear program solver, such as NPSOL [14], may be used to solve NLP2. To summarize, our worst-case analysis method consists of first computing the worst-case value of the performances and then determining the corresponding worst-case noise parameter vectors. This sequence of steps is different from that reported in [2]. Also, in [2], each noise parameter is perturbed an

equal probabilistic distance from its mean to obtain the worst-case noise vector. From the discussion above, it is clear that the noise vector found as such may not correspond to the most probable combination of noise parameters. Since our approach focuses on finding the most probable worst-case noise vector, it will inherently be more realistic than the approach in [2].

Some remarks regarding the worst-case analysis procedure are in order. Firstly, note that worst-case analysis is performed at a fixed design point, and that the procedure results in different worst-case noise vectors for different performances. Secondly, if the *jpdf* of the noise parameters is not Gaussian, then the nonlinear program NLP1 should be solved instead of NLP2. Thirdly, the procedure outlined above assumes that a single RSM is adequate to represent the circuit performance over the entire noise parameter range. If this is not true, a regionwise model should to be used. A regionwise modeling strategy deserves further investigation, although preliminary results of its application to worst-case analysis are encouraging.

C. Worst-Case Analysis Examples

In this section, we provide a few examples of the application of the worst-case analysis procedure outlined above, starting with a simple illustrative one. Each example uniquely illustrates a particular observation developed in this paper.

C.1 Example 1: A Low-Pass Filter Example

In this example, we use the low-pass filter circuit of Fig. 2(a). The performance measure of interest here is the time-constant $T = RC$. For this example, all R values will be reported in Kohm, all C values in picoFarad, and all T values in nanoseconds. The circuit parameters R and C are considered to be independent Gaussian random variables, with $R \sim \mathrm{N}(\mu_R = 10, \sigma_R = 1)$ and $C \sim \mathrm{N}(\mu_C = 10, \sigma_C = 1)$. The worst-case direction w for T is equal to $+1$. This example is simple enough that exact solutions can be obtained analytically to verify our worst-case analysis procedure.

Analytical

It can be easily shown that the mean value of T is $\mu_T = \mu_R\mu_C = 100$, and the standard deviation is $\sigma_T = [(\sigma_R^2 + \mu_R^2)(\sigma_C^2 + \mu_C^2) - \mu_T^2]^{1/2} = 14.2$. T is not a Gaussian random variable. Moreover, a comparison between the probability density of T and that of a Gaussian random variable with mean μ_T and standard deviation σ_T (Fig. 2(b)) shows that a Gaussian approximation will not be accurate. Hence, we use the procedure of Section III-B for the non-Gaussian case to compute the worst-case value of T, T^{wc}. A 10000-run MC sample of the noise parameters R and C is drawn, and the formula $T = RC$ is used to compute the values of T for this sample. The cumulative distribution function $F()$ is estimated from the sample (Fig. 2(c)), and (2) is solved for T^{wc} using the Newton-Raphson procedure. For $p = 0.002$, we obtain $T^{wc} = 145.48$. Now, the worst-case noise parameter vector satisfies the equation $RC = T^{wc}$, and the set of all possible solutions is the hyperbola shown in Fig. 2(d). In order to obtain the most probable of these solutions, we solve NLP2 which, in this case, is

$$\min \ \left(\left(\frac{R-\mu_R}{\sigma_R}\right)^2 + \left(\frac{C-\mu_C}{\sigma_C}\right)^2\right) \ \ s.t. \ \ RC = T^{wc}. \tag{9}$$

It can be shown that at the minima, the following condition holds:

$$\frac{R}{\sigma_R}\frac{R-\mu_R}{\sigma_R} = \frac{C}{\sigma_C}\frac{C-\mu_C}{\sigma_C}. \tag{10}$$

Since $\mu_R = \mu_C$ and $\sigma_R = \sigma_C$, we get $R^{wc} = C^{wc} = \sqrt{T^{wc}}$. Thus, the analytical solutions are $R^{wc} = 12.06$ and $C^{wc} = 12.06$.

Linear Response Surface

Let R' and C' denote the transformed independent noise parameters; they are given by $R' = (R - 10)/3$ and $C' = (C - 10)/3$. A linear RSM is constructed for T in terms of R' and C' using 10 circuit simulations. The response surface is $\hat{T} = 99.8 + 26.1R' + 32.6C'$. In general (unlike this example), the actual relationship between the circuit performance and the noise parameters is not known, and the decision as to whether the performance is Gaussian or not is based on the nature of the RSM. Since the RSM in this case is linear, T is a Gaussian random variable, and we use the procedure of Section III-B for the Gaussian case. With $k = 3$ (corresponding to $p = 0.002$) in (3), we obtain $T^{wc} = 142.04$. To obtain the worst-case values of the noise parameters, we use (8) which gives $R^{wc} = 11.86$ and $C^{wc} = 12.33$. The inaccuracy of the results in this case compared to the analytical results can be attributed to the fact that we are using a linear RSM to approximate an essentially nonlinear relationship between T and the noise parameters R and C.

Quadratic Response Surface

Next, we construct a quadratic RSM for T in terms of R' and C' from the same 10 simulations as above. The response surface is $\hat{T} = 99.7 + 30.3R' + 29.9C' + 0.19R'^2 + 18.4R'C' - 0.42C'^2$. In this case, T will not be Gaussian because of the nonlinear nature of the RSM. A procedure similar to that in the analytical case above is used to compute T^{wc}, with the exception that the quadratic RSM above is used to compute the values of T in the MC sample. The worst-case value is computed to be $T^{wc} = 145.77$. To determine the worst-case noise parameter values, we solve NLP2 in (6) with $\hat{h}()$ replaced by the RSM above. The worst-case values are computed to be $R^{wc} = 12.09$ and $C^{wc} = 12.06$. This result is very accurate compared to the analytical result, and demonstrates the correctness of the worst-case analysis procedure.

C.2 Example 2: MOSFET Realization of Resistor Pair

The circuit for this example is shown in Fig. 3 and consists of four identical NMOS transistors which simulate a pair of identical resistors R [15]. The performance measure of interest is the conductance $G = 1/R$ (in units of 10^{-6} mho). The noise parameters selected for this example are assumed to be independent Gaussian and their means and standard deviations are as follows: (i) gate oxide thickness u_1 (A°), $\mu_1 = 400$ and $\sigma_1 = 13.3$, (ii) electron mobility u_2 (cm^2/V.s), $\mu_2 = 700$ and $\sigma_2 = 23.3$, (iii) channel width reduction u_3 (μm), $\mu_3 = 0.0$ and $\sigma_3 = 0.067$, and (iv) channel length reduction u_4 (μm), $\mu_4 = 0.0$ and $\sigma_4 = 0.067$. Two response surfaces are used: a linear response surface (LRS) and a quadratic response surface (QRS), both fitted from 30 runs of the circuit simulator SPICE using linear regression. The worst-case analysis procedure for the RSMs is the same as that described in the previous example. The results using the RSMs are verified using a *confirmation experiment*. For a circuit performance r, the various steps in the confirmation experiment are: (i) a MC sample (size 2000) of the noise parameters is drawn, (ii) the circuit is *simulated* at the sample points and the values of r are extracted, (iii) the cumulative distribution function $F()$ is estimated from the r values in the MC sample, (iv) r^{wc} is computed by solving (2) using the estimated $F()$, and (v) the worst-case noise vector is chosen from the sample as the one that has the smallest probabilistic distance and for which the r value is within a small tolerance about r^{wc}. The results are shown in Table I, where the last column indicates the probabilistic distance at the worst-case vector. As seen from Table I, the results using the quadratic response surface are more accurate. Moreover, the QRS gives a worst-case vector with a much smaller probabilistic distance (and hence a much larger probability) than the LRS. This example demonstrates that *the nonlinear response surface approach coupled with the nonlinear programming formulation of the worst-case analysis problem yields realistic results.*

C.3 Example 3: CMOS Clock Driver Circuit

This example demonstrates the accuracy and the efficiency of the worst-case analysis procedure and the dependence of the worst-case conditions on the values of the designable parameters. The CMOS clock driver circuit, shown in Fig. 4, consists of an upper branch with three inverters and a lower branch with two inverters. The performance measure of interest is the clock skew ΔS (in units of nsecs), which is defined as the larger of the rising skew S_r and the falling skew S_f. The six noise parameters and their means and standard deviations are: (i) nMOS threshold voltage u_1 (V), $\mu_1 = 0.8$ and and $\sigma_1 = 0.04$, (ii) pMOS threshold voltage u_2 (V), μ_2 = -0.8 and $\sigma_2 = 0.04$, (iii) nMOS channel length reduction u_3 (μm), $\mu_3 = 0.05$ and $\sigma_3 = 0.005$, (iv) pMOS channel length reduction u_4 (μm), $\mu_4 = 0.05$ and $\sigma_4 = 0.005$, (v) nMOS channel width reduction u_5 (μm), $\mu_5 = 1.0$ and $\sigma_5 = 0.04$, and (vi) pMOS channel width reduction u_6 (μm), $\mu_6 = 1.1$ and $\sigma_6 = 0.04$. The correlation coefficients among the noise parameters are: $\rho(u_3, u_4) = 0.8$ and $\rho(u_5, u_6) = 0.6$. The six designable parameters for this circuit are the nominal widths (in μm) of the transistors MN2, MP2, MN3, MP3, MN5 and MP5. A quadratic response surface (QRS) is constructed for the clock skew ΔS in terms of the transformed noise parameters. This QRS is constructed from 20 runs of the simulator ILLIADS [16] (chosen according to LHS) using the MFQI approach. The worst-case analysis and the 2000-run Monte Carlo confirmation experiment are conducted at two design points $\mathbf{d}_1 = [3, 6, 3, 6, 3, 6]$ and $\mathbf{d}_2 = [2.8, 9, 2.7, 9, 1.8, 3.6]$. The results are shown in Table II. In the table, the last two columns display the probabilistic distance and the simulated values at the corresponding worst-case noise vectors (QRS or MC). The results of Table II demonstrate the accuracy and efficiency of the worst-case analysis procedure. Moreover, it also demonstrates that *the worst-case values of the circuit performances and the corresponding worst-case noise parameter vectors depend on the values of the designable parameters.*

C.4 Example 4: CMOS Operational Amplifier

This example demonstrates that the worst-case noise parameter vectors are different for different performance measures of a given circuit. The CMOS operational amplifier circuit used in this example is shown in Fig. 5. The performance measures of interest are the unity-gain frequency f_t (MHz), the low-frequency gain A_v (dB), and the power dissipation P_d (mW). The $[\mu - 3\sigma, \mu + 3\sigma]$ ranges of the 7 noise parameters are as follows: (i) nMOS and pMOS gate oxide thickness u_1 (A°), $360 \leq u_1 \leq 440$, (ii) nMOS threshold voltage u_2 (V), $0.7 \leq u_2 \leq 0.9$, (iii) pMOS threshold voltage u_3 (V), $-0.9 \leq u_3 \leq -0.7$, (iv) nMOS length reduction u_4 (μm), $-0.2 \leq u_4 \leq 0.2$, (v) pMOS length reduction u_5 (μm), $-0.2 \leq u_5 \leq 0.2$, (vi) nMOS width reduction u_6 (μm), $-0.2 \leq u_6 \leq 0.2$, and (vii) pMOS width reduction u_7 (μm), $-0.2 \leq u_7 \leq 0.2$. The correlation coefficients among the noise parameters are as follows: $\rho(u_1, u_2) = 0.8$, $\rho(u_1, u_3) =$

0.4, $\rho(u_2, u_3) = -0.5$, $\rho(u_4, u_5) = 0.8$, and $\rho(u_6, u_7) = 0.6$. A quadratic RSM is constructed for each of the performance measures in terms of the transformed noise parameters. 30 SPICE runs (chosen according to LHS) are used to construct these response surfaces using the MFQI approach. The worst-case analysis results are shown in Table III. A 5000-run Monte Carlo experiment similar to that in the previous examples is used to confirm the results obtained from the quadratic RSMs. Besides reiterating the accuracy of the worst-case analysis methodology, Table III shows that *different worst-case noise parameter vectors are required for different performance measures of the same circuit*.

IV. Worst-Case Design Optimization

The worst-case analysis technique described in the previous section is extended in this section to develop a new method for *circuit performance optimization*. The generic performance optimization problem involves varying the design point to optimize multiple performance objectives (which are possibly competing against each other) subject to constraints on the values of other secondary performance measures and designable parameter bounds [1]. Mathematically, the multi-criteria circuit performance optimization problem is

$$\begin{aligned} \max_{\mathbf{d}'} \quad & \{f_{oi}(\mathbf{d}'), 1 \le i \le l\} \\ \text{s. t.} \quad & a_j \le r_{cj}(\mathbf{d}') \le b_j, \quad 1 \le j \le m, \quad \mathbf{d}' \in \mathcal{D}. \end{aligned} \tag{11}$$

In the above, f_{oi}, $1 \le i \le l$, denotes the objective functions, r_{cj} denotes the constrained performances with lower bounds a_j and upper bounds b_j for $1 \le j \le m$, and $\mathcal{D}$ denotes the allowed (transformed) designable parameter space. A performance measure with worst-case direction $w = +1$ has to be minimized, whereas one with $w = -1$ has to be maximized. Therefore, we can define the objective functions f_{oi} as

$$f_{oi}(\mathbf{d}') = -w_{oi} r_{oi}(\mathbf{d}'), \tag{12}$$

where r_{oi}, $1 \le i \le l$, denotes the performance measures which are the objectives of the optimization. The specifications for the performance objectives are *one-sided*: upper bounds for performances with $w_{oi} = +1$ (i.e. $r_{oi} \le r_{oi}^{spec}$) and lower bounds for performances with $w_{oi} = -1$ (i.e. $r_{oi} \ge r_{oi}^{spec}$).

Most performance optimization approaches presented in the past [1],[17] are *nominal design optimization* methods since they consider only nominal values of performances (performance values when the noise parameters are at their mean values). If r^0 denotes the nominal value of a performance r, this optimization problem can be obtained from (11) and (12) by substituting r_{oi}^0 for r_{oi} and r_{cj}^0 for r_{cj}. Since noise parameters in these approaches are fixed at their mean values, statistical variations are ignored. As a result, the final design point may not be well-centered and may have low parametric yield, even though the specifications are nominally satisfied. The approach presented in this paper is called *worst-case design optimization* (WCDO). Its goal is to optimize the worst-case values of the performance objectives subject to bounds on the nominal values of the performance constraints and on the designable parameters. In other words, the objective functions f_{oi} in WCDO are defined as

$$f_{oi}(\mathbf{d}') = -w_{oi} r_{oi}^{wc}(\mathbf{d}'). \tag{13}$$

If the final design point of WCDO satisfies performance specifications, then the parametric yield is guaranteed to be very high since the specifications are met under worst-case conditions. A graphical comparison of worst-case and nominal design optimization is given in Fig. 6.

Performing the usual conversion of the multi-criteria optimization problem of (11) into a max-min problem, the WCDO procedure can be formally stated as

$$\begin{aligned} \textbf{WCDO}: \quad & \max_{\mathbf{d}'} \min_{1 \le i \le l} \quad -w_{oi} r_{oi}^{wc}(\mathbf{d}') \\ \text{s. t.} \quad & a_j \le r_{cj}^0(\mathbf{d}') \le b_j, \quad 1 \le j \le m, \quad \mathbf{d}' \in \mathcal{D}. \end{aligned} \tag{14}$$

Note that even though the noise parameters do not appear in the above formulation of WCDO, they are implicitly taken into account by considering the worst-case values of the performance objectives.

The WCDO formulation is similar to that of the design centering problem presented by Low et al. [18]. Despite the obvious similarities in the formulation, the goals of the two approaches are quite different: the WCDO procedure is a performance optimization method whereas design centering is a geometric interpretation of the yield maximization problem. A comparison of the two methods, highlighting their differences, is given in the Appendix. The following two sections discuss the solution of the WCDO problem and the computation of the worst-case performance values.

A. Objective Function for WCDO

For each performance *objective*, r_{oi}, $1 \leq i \leq l$, a "good" or desirable value g_{oi} and a "bad" or undesirable value b_{oi} are assigned by the user. At any design point $\mathbf{d}'$, the *worst-case performance value* is converted into a *performance score* $s_{oi}(\mathbf{d}')$ as follows:

$$s_{oi}(\mathbf{d}') = 100 * (\frac{r_{oi}^{wc}(\mathbf{d}') - b_{oi}}{g_{oi} - b_{oi}}) \tag{15}$$

In other words, g_{oi} is assigned a score of 100, b_{oi} is assigned a score of 0 and all intermediate performance values are scored by linear interpolation between these two values. Various other nonlinear relationships between the worst-case performance value and the performance score may also be used, based on the relative desirability of the values. Since the worst-case directions of the performance objectives may be different, the performance score formulation allows a uniform treatment of all the performance objectives. In terms of the performance scores, the WCDO problem of (14) reduces to

$$\begin{aligned} &\max_{\mathbf{d}'} \ \min_{1 \leq i \leq l} \quad s_{oi}(\mathbf{d}') \\ &\text{s. t. } \ a_j \leq r_{cj}^0(\mathbf{d}') \leq b_j, \quad 1 \leq j \leq m, \quad \mathbf{d}' \in \mathcal{D}. \end{aligned} \tag{16}$$

The individual performance scores at a design point are used to compute an overall index of circuit quality called the *circuit score*. The circuit score, denoted by $s_o(\mathbf{d}')$, is the objective function for worst-case design optimization. Thus, the constrained multi-criteria optimization problem of (16) is converted into an optimization problem with a single objective function. The definition of the circuit score depends on the kind of optimization method used. Two well-known methods have been used.

A.1 Nelder-Mead Simplex Method

This method [19] deals with *unconstrained* optimization problems and does not require gradients. In order to convert the WCDO problem of (16) into an unconstrained problem with a single objective function, we first define the circuit score as

$$s_o(\mathbf{d}') = \begin{cases} \min_{1 \leq i \leq l} \ s_{oi}(\mathbf{d}'), & \text{if } a_j \leq r_{cj}^0(\mathbf{d}') \leq b_j, \\ & 1 \leq j \leq m, \ \mathbf{d}' \in \mathcal{D} \\ -\infty, & \text{otherwise} \end{cases} . \tag{17}$$

Then, the WCDO problem becomes

$$\max_{\mathbf{d}'} \ s_o(\mathbf{d}'). \tag{18}$$

Note that, in this case, the circuit score is not differentiable. Non-feasible design points are penalized by assigning circuit scores that are very small so that the optimization is driven away from such non-feasible regions. Even though this method is much slower than the gradient-based methods, it has been observed to be very robust for circuit design optimization applications [20].

A.2 Sequential Quadratic Programming (SQP)

In this method [21], the objective function and constraints have to be smooth, and therefore, the "min" – operator of the previous method cannot be used. Instead, we define the circuit score as the p-norm ($p < 0$, e.g. $p = -10$) of the individual circuit scores [22]. Mathematically,

$$s_o(\mathbf{d}') = \| \ [s_{o1}(\mathbf{d}'), \ldots, s_{ol}(\mathbf{d}')]^T \ \|_p \tag{19}$$

Then, the WCDO problem becomes

$$\begin{aligned} &\max_{\mathbf{d}'} \quad s_o(\mathbf{d}') \\ &\text{s. t. } \ a_j \leq r_{cj}^0(\mathbf{d}') \leq b_j, \quad 1 \leq j \leq m, \quad \mathbf{d}' \in \mathcal{D}. \end{aligned} \tag{20}$$

In the computation of the circuit score, zero performance scores may occur at the "bad" values specified by the user. These are avoided by slightly modifying the bad value or by assigning small but non-zero scores at those points. Thus, the above scheme maps the worst-case values of the performance objectives into a differentiable index of circuit quality, the circuit score. The nonlinear programming package NPSOL [14] is used as the SQP engine to solve (20).

B. Augmented Response Surface Modeling

The dependence of r_{oi}^{wc} on the designable parameters is due to the dependence of $\mu_{r_{oi}}$ and $\sigma_{r_{oi}}$ on $\mathbf{d}$. The most natural and computationally efficient way to account for this dependence is to construct response surfaces for the performance *objectives* in terms of both the (transformed) designable and noise parameters. These response surfaces are denoted by $\hat{r_{oi}}(\mathbf{d}', \mathbf{v})$, $1 \leq i \leq l$, where l is the number of performance objectives. At a particular design point $\mathbf{d}'_k$, $\hat{r_{oi}}(\mathbf{d}'_k, \mathbf{v})$ is some function of the transformed noise parameters $\mathbf{v}$ alone, say $\hat{t_{oi}}(\mathbf{v})$. $\hat{t_{oi}}()$ represents the variation of the ith performance objective at a particular design point with respect to the noise parameters. The mean and the standard deviation of r_{oi} at $\mathbf{d}'_k$ can be estimated using $\hat{t_{oi}}()$ in the same manner as in Section III-B, and these are denoted by $\mu_{r_{oi}}(\mathbf{d}'_k)$ and $\sigma_{r_{oi}}(\mathbf{d}'_k)$, respectively. Then for each $i = 1, \ldots, l$, the worst-case performance value at $\mathbf{d}'_k$ can be obtained using (3) as

$$r_{oi}^{wc}(\mathbf{d}'_k) = \mu_{r_{oi}}(\mathbf{d}'_k) + k w_i \sigma_{r_{oi}}(\mathbf{d}'_k) \tag{21}$$

If required, the worst-case noise parameter vector at $\mathbf{d}'_k$, $\mathbf{v}_{oi}^{wc}(\mathbf{d}'_k)$, is obtained as the solution of (6) with $\hat{h}_i$ replaced by $\hat{t_{oi}}$, and $\mathbf{u}_{oi}^{wc}(\mathbf{d}'_k) = \mathbf{T}^{-1}(\mathbf{v}_{oi}^{wc}(\mathbf{d}'_k))$.

The l augmented response surface models are fitted in a region in the design space called the *sampling box*. This region is usually much smaller than the entire design space so that a single response surface can accurately approximate the behavior of the circuit performances over the range of the sampling box. The dimensions of the sampling box are passed as designable parameter bounds ($\mathcal{D}$) to the WCDO procedure. Thus, the optimization proceeds in stages, with the final design point of the current stage becoming the initial design point for the next and the new sampling box being an n_d-dimensional box about this design point. Sometimes, however, during one stage, the RSMs may lose accuracy as the design points move away from the center of the current sampling box. In such cases, the worst-case noise parameter vector for each of the l performance objectives is computed as above, and the circuit is simulated l times corresponding to the combination of the current design point and the l worst-case noise vectors. The simulation results are then used to update the response surfaces. These adaptively refitted response surfaces ensure that the worst-case values of the performance objectives are accurate at each design point encountered during the optimization.

Since only the nominal values of the performance *constraints* are required, the corresponding RSMs are in terms of the transformed designable parameters alone. These are denoted by $\hat{r_{cj}}(\mathbf{d}')$, for $1 \leq j \leq m$. The concepts of the sampling box and adaptive fitting as described above for the performance objectives are utilized for the performance constraints also. The overall flow of the WCDO procedure is shown in Fig. 7.

C. Worst-Case Design Optimization Examples

In this section, we demonstrate the application of the WCDO procedure to a CMOS clock driver circuit and a CMOS operational amplifier. In both the examples, the parametric yield estimate at any design point, $\hat{Y}$, is obtained by simulating a MC sample of noise parameters of size $N_{mc} = 5000$ and counting the number of cases for which the specifications are met. Note that the standard error for $\hat{Y}$ [13] is given by

$$\sigma(\hat{Y}) = [\hat{Y}(1 - \hat{Y})/(N_{mc} - 1)]^{1/2}. \tag{22}$$

C.1 Example 1: CMOS Clock Driver

In this example, the circuit of Fig 4 (used in Section III-C.3) is used to demonstrate the WCDO procedure. The designable and noise parameters used in this example are the same as in Section III-C.3. The performance objective is the clock skew ΔS defined earlier and the performance constraint is the nominal circuit area A^0. The performance optimization problem for the clock driver circuit is

$$\min \quad \Delta S \qquad \text{s.t.} \qquad A^0 \leq 180\ \mu\text{m}^2$$

with the specification on the clock skew being $\Delta S \leq 0.5ns$. In this example, the Nelder-Mead Simplex method is used as the optimizer for the WCDO procedure. Hence the WCDO problem is formulated as in (18) and the circuit score is computed using (17). The following quadratic response surface is constructed for ΔS in terms of the transformed designable and noise parameters:

$$\hat{\Delta S} = a + \mathbf{b}^T \mathbf{d}' + \mathbf{c}^T \mathbf{v} + \frac{1}{2} \mathbf{d}'^T \mathbf{Q} \mathbf{d}' + \mathbf{d}'^T \mathbf{R} \mathbf{v} + \frac{1}{2} \mathbf{v}^T \mathbf{S} \mathbf{v}. \tag{23}$$

This RSM, containing 91 coefficients, is fitted at the beginning of each pass of the WCDO procedure using ΔS values from 30 ILLIADS runs. Fig 8 shows the ΔS values predicted from the RSM plotted against simulated values for a set of *checking points* (different from training points used to fit the RSM). Most of the points lie near the 45^0 line indicating that the predicted values are in close agreement with the simulated values. The circuit area is defined as the sum of the products of the widths and lengths of all the MOS transistors in the circuit. A RSM for the area is not required

since simulations are not needed to compute it. The status of the optimization at the beginning and end of each pass is shown in Table IV. ΔS^{wc}_{pred} denotes the value of ΔS^{wc} predicted by the RSM, while ΔS^{wc}_{cal} denotes the value of ΔS^{wc} calculated by worst-case analysis at the current design point. SB denotes the size of the sampling box as a percentage of the current designable parameter values. The nominal value of the area A^0 and the parametric yield estimate $\hat{Y}$ are also shown in the table. In the third pass, one additional simulation is used to adaptively refit the RSM for ΔS. Note that this involves computing the worst-case noise vector for the skew ΔS at the current design point, simulating the circuit at the worst-case noise conditions, and updating the RSM. The close agreement between the ΔS^{wc}_{pred} and ΔS^{wc}_{cal} values indicates that the RSM provides realistic estimates of the worst-case conditions throughout the optimization. The final design point obtained from the WCDO procedure is $\mathbf{d}^* = [2.58, 8.70, 4.69, 9.375, 1.69, 3.375]$ at which ΔS^{wc} and A^0 satisfy their specifications. The parametric yield at $\mathbf{d}^*$ is estimated to be 100%. (Note: $\sigma(\hat{Y}) = 0$ according to formula (22), but if the specification on ΔS is tightened to 0.36ns, then at $\mathbf{d}^*$, $\hat{Y} = 97.6\%$ and $\sigma(\hat{Y}) = 0.69\%$.) The total simulation cost in this example is 61 ILLIADS runs. This example demonstrates the efficiency and accuracy of the WCDO technique.

C.2 Example 2: CMOS Operational Amplifier

This example deals with the worst-case design optimization of the CMOS operational amplifier shown in Fig. 5. The 7 correlated noise parameters, their means, variances and correlation coefficients are given in Section III-C.4. The 5 designable parameters are: (i) d_1 = width of M1 and M2 (in μm), (ii) d_2 = width of M3 and M4 (in μm), (iii) d_3 = width of M5 and M8 (in μm), (iv) d_4 = width of M6 (in μm), and (v) d_5 = value of the compensation capacitor CC (in pF). There are three performance measures of interest; two are objectives, unity-gain frequency f_t and low-frequency gain A_v, and the third is a constraint, power dissipation P_d. The performance optimization problem for the opamp is to

$$\max \quad f_t, \ A_v \quad \text{s.t.} \qquad P_d^0 \leq 0.6 \ \text{mW}$$

The specifications on the performance objectives are: $f_t \geq 7.5$MHz and $A_v \geq 60$dB. The NPSOL package is used as the optimizer for the WCDO procedure in this example. The circuit score is given by (19) with $p = -10$ and the optimization problem is formulated as in (20). Note that the nominal power dissipation is constrained. If it was desired that the worst-case value of power dissipation satisfy some specification, then it would become an objective. The resultant optimization problem would be different, but could still be handled by the WCDO procedure.

The worst-case design optimization proceeds in stages as before. At each stage, a sampling box is defined and the designable parameter values are chosen from this sampling box by LHS. A quadratic response surface similar to (23) is constructed for f_t and A_v. For P_d, a quadratic response surface is constructed in terms of the designable parameters alone. The dimensions of the sampling box are passed as bounds to the optimizer and the response surfaces are refitted, if required, to maintain accuracy. Only the initial and final design points of the optimization are shown in Table V. The optimization is stopped as soon as the specifications are met under worst-case conditions. The parametric yield estimate $\hat{Y}$ at the two design points are obtained as explained earlier. At the initial design point $\mathbf{d}_0$, $\hat{Y}$ is zero since the specifications on f_t and A_v are not met in any of the simulated cases. The final design point $\mathbf{d}_f$ satisfies the specifications under worst-case conditions; the parametric yield estimate at $\mathbf{d}_f$ is 100%. In this example, the total cost of the WCDO procedure is 172 SPICE simulations.

V. Conclusions and Future Work

In this paper, we have presented a new approach for worst-case analysis of VLSI circuit performances. For realistic worst-case analysis, we have formulated a nonlinear programming problem in which the objective is to maximize the joint probability density of the noise parameters. The worst-case analysis approach has been demonstrated, through illustrative examples, to be accurate and efficient. A new methodology for circuit performance optimization has also been presented. This method optimizes the worst-case values of multiple performance objectives subject to constraints on the nominal values of other performance measures and designable parameter bounds. Response surface models of the circuit performances in terms of the designable and noise parameters have been used. The circuit design examples presented demonstrate the efficiency, robustness and effectiveness of the worst-case design optimization method.

A number of areas remain open for future research. Firstly, as mentioned earlier, the worst-case analysis method needs to be extended to work with regionwise response surface models of the circuit performances. Preliminary work on this subject has been encouraging. Secondly, much work remains to be done in the area of performance modeling in the context of statistical design: sampling techniques that are more space filling than LHS, better diagnostic and validation tools for the maximally flat quadratic interpolation technique, and graphical tools for the visualization of models are some of the possible areas for future work. Thirdly, the adaptive fitting of response surface models in the WCDO procedure is currently done heuristically and should be improved. Also, the process of switching between gradient-based and non-gradient-based optimizers when the latter stops at local minima is currently dependent on user-intervention; these steps should be automated.

Appendix

Comparison of WCDO and Design Centering

As pointed out by a reviewer of this paper, the WCDO methodology presented in this paper is similar to the design centering methodology of Low et al. [18]. There are obvious similarities: both approaches use response surface modeling, the formulations for the two approaches are alike, and both methods have used NPSOL. Despite these similarities, the two approaches are quite different. Design centering (DC) is a yield maximization technique: all circuit performances appear as constraints in the optimization, the objective function of which is the yield. WCDO is a performance optimization technique; yield maximization is not the goal and there is a clear distinction between the performances that have to be optimized and those that have to be constrained. High parametric yield at the final design point is a useful by-product of the WCDO procedure (and indeed its motivation) since the performance specifications are met under worst-case conditions. Both methodologies have been formulated as a max-min problem; in design centering the objective function is the smallest geometric distance of a design point from the boundary of the acceptable region, whereas in WCDO, the objective function is the worst-case performance value. Another difference between the two approaches is in the way the performance specifications appear in the definition of the yield. In design centering, the performances are constraints and their specifications are *two-sided*. In WCDO, since the performance objectives have to be maximized or minimized, their specifications are *one-sided*. Thus, WCDO is much different from the design centering approach of Low et al. The following simple example demonstrates the difference between the two approaches.

Consider a hypothetical circuit with one circuit parameter x which has a designable component d and a noise component v, $v \sim N(0,1)$, and one performance r whose dependence on the circuit parameter is known apriori to be $r = x = d + v$. Let the design space be defined by $d \geq 4$. Suppose we first want to maximize the yield by design centering and the yield specification (2-sided) on the performance is $4 \leq r \leq 9$. Then the optimal design point according to DC is $d^{\star} = 6.5$ and the yield is given by

$$\begin{aligned} Y(d^{\star}) &= Prob[4 \leq r \leq 9] = Prob[-2.5 \leq v \leq 2.5] \\ &= \Phi(2.5) - \Phi(-2.5) = 2\Phi(2.5) - 1, \end{aligned}$$

where Φ is the standard normal distribution function. Suppose we now want to minimize the worst-case value of r subject to design space restrictions. Let the yield specification (1-sided) be $r \leq 9$. In this case, it can be easily verified that the optimal design point according to WCDO is $d^{\star\star} = 4$ and the yield is

$$\begin{aligned} Y(d^{\star\star}) &= Prob[r \leq 9] = Prob[v \leq 5] \\ &= \Phi(5) \end{aligned}$$

Thus, it can be seen that the two approaches address different problems and therefore their answers are quite different.

References

[1] R. K. Brayton, G. D. Hachtel, A. L. Sangiovanni-Vincentelli, "A survey of optimization techniques for integrated circuit design," *Proc. IEEE*, vol. 69, pp. 1334-1362, 1981.

[2] S. R. Nassif, A. J. Strojwas, and S. W. Director, "A methodology for worst-case analysis of integrated circuits," *IEEE Trans. Computer-Aided Design*, vol. CAD-5, no. 1, pp. 104-113, Jan. 1986.

[3] G. E. P. Box, W. J. Hunter, and J. S. Hunter, *Statistics for Experimenters: An Introduction to Design, Data Analysis, and Model Building*. New York:Wiley, 1978.

[4] G. E. P. Box and N. R. Draper, *Empirical Model-Building and Response Surfaces*. New York:Wiley, 1987.

[5] T. K. Yu, S. M. Kang, I. N. Hajj, and T. N. Trick, "Statistical performance modeling and parametric yield estimation of MOS VLSI," *IEEE Trans. Computer-Aided Design*, vol. CAD-6, pp. 1013-1022, Nov. 1987.

[6] P. Yang and P. Chatterjee, "Statistical Modeling of small geometry MOSFETs," *IEEE Int. Electron Device Meeting*, pp. 286-289, Dec. 1982.

[7] K. K. Low and S. W. Director, "An efficient methodology for building macromodels of IC fabrication processes," *IEEE Trans. Computer-Aided Design*, vol. CAD-8, no. 12, pp. 1299-1313, Dec. 1989.

[8] M. C. Bernardo, R. Buck, L. Liu, W. A. Nazaret, J. Sacks, and W. J. Welch, "Integrated circuit design optimization using a sequential strategy," *IEEE Trans. Computer-Aided Design*, vol. CAD-11, no. 3, pp. 361-372, March 1992.

[9] M. D. McKay, R. J. Beckman, and W. J. Conover, "A comparison of three methods for selecting values of input variables in the analysis of output from a computer code," *Technometrics*, vol. 21(2), pp. 239-245, May 1979.

[10] R. M. Biernacki and M. A. Styblinski, "Statistical circuit design with a dynamic constraint approximation scheme," *Proc. IEEE Int. Symp. Circuits and Systems*, pp. 976-979, 1986.

[11] P. Tuohy, A. Gribben, A. J. Walton, and J. M. Robertson, "Realistic worst-case parameters for circuit simulation," *IEE Proceedings*, vol. 134, no. 5, pp. 137-140, Oct. 1987.

[12] G. E. Müller-L., "Limit-parameters: the general solution of the worst-case problem for the linearized case," *Proc. IEEE Int. Symp. Circuits and Systems*, pp. 2256-2259, 1990.

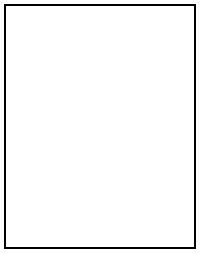
[13] J. M. Hammersley and D. C. Handscomb, *Monte Carlo Methods*. London: Methuen and Co. Ltd., 1964.

[14] P. E. Gill, W. Murray, M. A. Saunders, and M. H. Wright, "User's guide for NPSOL, version 4.0," Report SOL86-2, Department of Operations Research, Stanford University, 1986.

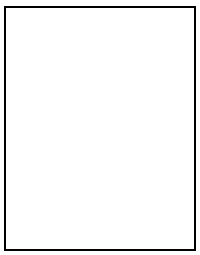
[15] Z. Czarnul, "Modification of the Banu-Tsividis continuous-time integrator structure," *IEEE Trans. Circuits Syst.*, vol. CAS-33, pp. 714-716, July 1986.

[16] Y. H. Shih, Y. Leblebici, and S. M. Kang, "ILLIADS: a fast timing and reliability simulator for digital MOS circuits," *IEEE Trans. Computer-Aided Design*, vol. CAD-12, no. 9, pp. 1387-1402, Sept. 1993.

[17] W. Nye, D. C. Riley, A. L. Sangiovanni-Vincentelli, and A. L. Tits, "DELIGHT.SPICE: an optimization-based system for the design of integrated circuits," *IEEE Trans. Computer-Aided Design*, vol. CAD-7, no. 4, pp. 501-519, April 1988.
[18] K. K. Low and S. W. Director, "A new methodology for the design centering of IC fabrication processes," *IEEE Trans. Computer-Aided Design*, vol. CAD-10, no. 7, pp. 895-903, July 1991.
[19] J. A. Nelder and R. Mead, "A simplex method for function mimization," *Comput. J.*, vol. 7, pp. 308-313, 1965.
[20] A. Dharchoudhury and S. M. Kang, "An integrated approach to realistic worst-case design optimization of MOS analog circuits," *Proc. 29th ACM/IEEE Design Automation Conf.*, pp. 704-709, 1992.
[21] D. G. Luenberger, *Linear and Nonlinear Programming*, Reading, MA: Addison-Wesley, 1984.
[22] J. W. Bandler and C. Charalambous, "Theory of generalized least pth approximation," *IEEE Trans. Circuit Theory*, vol. CT-19, pp. 287-289, May 1972.



Abhijit Dharchoudhury received the B. Tech. degree in electrical engineering from the Indian Institute of Technology, Kanpur, in 1989. In 1992, he received the M. S. degree in electrical engineering from the University of Illinois at Urbana-Champaign, where he is currently pursuing his Ph. D. degree. Since 1989, he has been a research assistant at the Coordinated Science Laboratory and at the Beckman Institute of Advanced Science and Technology. In the summer of 1990, he worked at the Semiconductor Process and Design Center of Texas Instruments Inc., Dallas, TX in the area of high performance analog circuit optimization. His research interests include statistical design and analysis, circuit optimization techniques, fast timing simulation, and mixed-mode simulation.



S. M. Kang (S'73-M'75-SM'80-F'90) received the Ph. D. degree in electrical engineering from the University of California at Berkeley in 1975. Until 1985, he was with AT&T Bell Laboratories at Murray Hill and Holmdel, and had also served as faculty member of Rutgers University. In 1985, he joined the University of Illinois at Urbana-Champaign where he is Professor of Electrical and Computer Engineering, Computer Science and Research Professor of the Coordinated Science Laboratory and the Beckman Institute of Advanced Science and Technology. He is also an associate in the Center for Advanced Study, University of Illinois at Urbana-Champaign. Currently, he serves as Associate Director of NSF Engineering Research Center for Compound Semiconductor Microelectronics and Director of University of Illinois Center for ASIC Research and Development. In 1989, he was a Visiting Professor at the Swiss Federal Institute of Technology at Lausanne. His research interests include VLSI design methodologies and optimization for performance, reliability and manufacturability, modeling and simulation of semiconductor devices and circuits, high-speed optoelectronic circuits, and fully optical network systems.

He has served as AdCom member, Secretary and Treasurer, Administrative Vice President, and 1991 President of IEEE Circuits and Systems Society. He has served on the program committees and technical committees of major international conferences which include DAC, ICCAD, ICCD, ISCAS, MCMC, International Conference on VLSI and CAD (ICVC), Asia-Pacific Conference on Circuits and Systems, LEOS Topical Meeting, SPIE OE/LASE Meeting, and on the editorial boards of IEEE Transactions on Circuits and Systems, International Journal of Circuit Theory and Applications, and Circuits, Signals, and Systems. He is the Founding Editor-in-Chief of the IEEE Transactions on Very Large Scale Integration (VLSI) Systems.

Dr. Kang is listed in Who's Who in the World, Who's Who in America, Who's Who in Technology, and Who's Who in Engineering. He has received the IEEE Circuits and Systems Society Meritorious Service Award (1994), SRC Inventor Recognition Awards (1993), IEEE CAS Darlington Prize Paper Award (1993) and other best paper awards (1979, 1987) and co-authored five books, *Design Automation for Timing-Driven Layout Synthesis* (1992), *Hot-Carrier Reliability of MOS VLSI Circuits* (1993), *Physical Design for Multichip Modules* (1994), and *Modeling of Electrical Overstress in Integrated Circuits* (1994) from Kluwer Academic Publishers, and *CMOS Digital Integrated Circuits: Analysis and Design* to be published by McGraw-Hill.