

# Remarks on Statistical Design Centering

Leszek J. Opalski

**Abstract**—The paper overviews optimization based statistical design centering techniques for analog circuits. Emphasis is placed on dependence between formulation of quality indices, problem formulation, and computational complexity of design centering algorithms, executed in single- or multiple-processor environments. For characterization of solution techniques a standard CMOS op-amp design case and a simplified computational complexity analysis are used.

**Keywords**—statistical design centering, design yield, realistic worst-case design, stochastic approximation, propagation of variance, Monte Carlo simulation.

## I. INTRODUCTION

THIS paper overviews most successful/promising design centering techniques for analog circuits, emphasizing relationships that exist between formulation of a design problem, computational complexity and user-perceived quality of its solution. The overview concentrates on three groups of design centering problems: nominal design, yield optimization, realistic worst-case design. Two groups of optimization-based centering methods (design centering solvers) are considered – purely deterministic, and stochastic. Fundamental properties of presented problem formulations and solvers are exemplified and uniformly compared with advent of calculations, made for one standard CMOS op-amp example case. Brief theoretical support is given for making rational prediction of solver’s behavior for other circuits, that differ in the number of designable parameters and/or number of variability sources.

## II. CIRCUIT MODEL FOR STATISTICAL DESIGN CENTERING

Automatic synthesis techniques require, that circuit responses of interest,  $y_i$ ,  $i = 1, \dots, m$ , can be evaluated numerically with sufficient accuracy by a circuit simulator. Typically a circuit under design is represented with a *parametrized* net-list of connected primitives, as RLC elements, voltage/current sources, diodes, transistors. As such, the circuit simulation implements the mapping of net-list parameters  $\mathbf{e} \in \mathcal{R}^{n_e}$  into responses:  $\mathbf{y} = \varphi^e(\mathbf{e})$ . Usually the circuit parameters  $\mathbf{e}$  are not directly designable, depending on a set of designer controllable (designable) parameters  $\mathbf{x} \in \mathcal{R}^n$ , and a set of (uncontrollable) disturbances  $\theta \in \mathcal{R}^\theta$  – representing manufacturing inaccuracy and variability of environment (temperature, power supply, variability of input signal or load):  $\mathbf{e} = \phi(\mathbf{x}, \theta)$ . Dependence of response on designable parameters and variability sources will be denoted as  $\mathbf{y} = \varphi(\mathbf{x}, \theta)$ .

We will not limit consideration, to the frequently assumed model:  $e_i = x_i + \theta_i$  – which is insufficient for integrated

circuit modeling, and will assume, that the transformation  $\phi$  can be non-linear and perhaps implicit, as implemented with a statistical macro-model of parameter variability [1], [2], [3], [4] or a statistical fabrication process simulator [5], [6].

The disturbances  $\theta$  can have the following nature.

- Manufacturing variations make each circuit element slightly different from another one, nominally identical. Two consequences of these variations are important.
  - Two nominally identical circuits demonstrate (more or less) different responses/behavior – which might be undesirable for cooperating circuits in a hierarchical design, or for circuit quality perception of the final user.
  - Since analog designers tend to use symmetry (to rely on more accurate ratios of parameter values and not on absolute values) – mismatch of the nominally identical elements can significantly degrade design performance of each circuit instance.
- Variability of environmental parameters: ambient temperature, power supply, input signal, load. Statistical description is very difficult to construct (e.g. weather conditions), so typically design is expected to be resistant to *any* environmental variability within a finite interval (e.g. ambient temperature: 0-30°C).

In this overview we will concentrate on variability sources  $\theta$ , that can be characterized statistically with joint probability density function (p.d.f.), denoted as  $f_\theta(\theta)$ . We will also assume, that elements of the  $\theta$  vector are Gaussian random variables with mean  $\bar{\theta}$  and covariance matrix  $\mathbf{C}_\theta$ . Thus the  $\theta$  vector can be obtained from a vector  $\eta$  of standard independent Gaussian variables via a linear transformation:

$$\theta = \bar{\theta} + \mathbf{L}_\theta \eta, \text{ where } \mathbf{L}_\theta \mathbf{L}_\theta^t = \mathbf{C}_\theta \quad (1)$$

**Example 1.** The internally compensated CMOS op-amp shown in Fig. 1 will be used for illustration of properties of different design centering problem formulations and solution methods. Four circuit responses are considered: gain bandwidth product (GBW), low frequency gain (A0), phase margin (PM), and the slew-rate (SR).

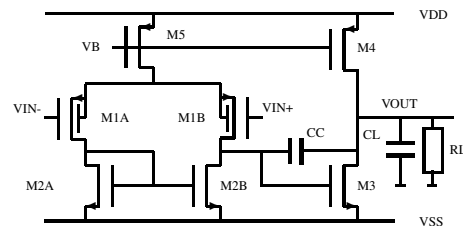


Fig. 1. Internally compensated CMOS op-amp.  $RL=1T\Omega$ ,  $CL=10pF$ .

The vector of decision variables  $\mathbf{x}$  consists of gate widths and lengths of all 7 transistors, the bias voltage  $VB$ , and the compensation capacitance  $CC$ . Nominal values of designable parameters and response specifications are shown in Table I. The vector  $\theta$  consists of 9 independent global Gaussian random variability sources, with mean values and standard deviations as shown in Table II. A non-linear (quadratic) statistical model, described in [1] is used, to transform samples of the vector  $\theta$  into transistor parameter values. Symbolic macro-model is used then, to evaluate respective response values fast.

### III. NOMINAL DESIGN CENTERING

Even though we are interested in statistical design centering – yet considering nominal design is justified (if not necessary), as it gives some estimate of minimum computational effort needed by design centering. Besides it can give a useful starting point for subsequent statistical centering.

Nominal design assumes, that variability sources attain fixed nominal values  $\bar{\theta}$ . To avoid unimplementable designs and limit search space (for efficiency reasons) – the designable parameters are subject of feasibility constraints  $\mathbf{x} \in \mathbb{X} \subset \mathcal{R}^n$ . Typically box constraints are imposed  $x_i^L \leq x_i \leq x_i^U$ ,  $i = 1, \dots, n$ , but a more general setting might also be desired:

$$\mathbb{X} = \{ \mathbf{x} \in \mathcal{R}^n \mid \begin{aligned} h_i(\mathbf{x}) &= 0, \quad i = 1, \dots, n_h; \\ g_i(\mathbf{x}) &\leq 0, \quad i = 1, \dots, n_g \end{aligned} \} \quad (2)$$

Equality conditions can, e.g., impose tracking of some parameters (to provide circuit symmetry). Inequalities can prevent use of unimplementable parameter values and/or use of improper regimes of transistor operation (saturation, breakdown, etc.). Generally, linear constraint functions  $h_i, g_i$  do not increase computational complexity of modern optimization processes, but non-linear do, since feasibility is usually enforced by optimizers iteratively. If  $\mathbf{x}$  components are additionally restricted to attain only a finite (discrete) set of values (e.g. [7]) – a substantial combinatorial factor appears in complexity, but this case will not be discussed here.

TABLE I  
NOMINAL VALUES OF CMOS OP-AMP DESIGNABLE PARAMETERS AND PERFORMANCE SPECIFICATIONS

Variable	$\mathbf{x}^{(0)}$	#1	#2	#3
W1A, W1B	26 $\mu\text{m}$	V,V	T1	T1
W2A, W2B	10 $\mu\text{m}$	V,V	T2	T2
W3	115 $\mu\text{m}$	V	V	V
W4	55 $\mu\text{m}$	V	V	V
W5	13 $\mu\text{m}$	V	V	V
L1A, L1B	16 $\mu\text{m}$	V,V	T3	
L2A, L2B	10 $\mu\text{m}$	V,V	T4	
L3, L4	5 $\mu\text{m}$	V,V	V,V	
L5	10 $\mu\text{m}$	V	V	
VB	1.07 V	V	V	V
CCNOM	1 pF	V	V	V

Response	$y_i^L$	$y_i^T$	$y_i^U$	nom.value
GBW [MHz]	2	2.5	5	2.056
A0 [V/mV]	5	10	35	9.468
PM [deg]	55	60	65	55.741
SR [V/ $\mu\text{s}$ ]	2	5	10	2.9826

Columns '#1', '#2' and '#3' display variables used for the three optimization runs, V – independent variable, T# – tracking variable group of given #.

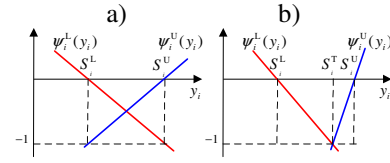


Fig. 2. Illustration of linear scaling of responses with (3) and weights: a) (4), b) (5).

A generic nominal design problem formulation can be based on a concept of “designer’s satisfaction” [8]. It is assumed, that a designer can measure progress towards his/her (perhaps unreachable) goals by specifying for each response some finite upper  $y_i^U$  or lower  $y_i^L$  satisfactory levels, respectively. Reaching by the  $j$ -th response its respective satisfactory level should give the same “designer’s satisfaction” as reaching by the  $i$ -th response its respective satisfactory level. Based on these premises partial quality indices can be defined:

$$\psi_i^L(y_i) = w_i^L (y_i^L - y_i) \quad \psi_i^U(y_i) = w_i^U (y_i - y_i^U) \quad (3)$$

and then minimized – in the course of design centering. Weighting coefficients  $w_i^L, w_i^U$  are selected such, that specified increments of response values ( $y_i^L - y_i$  or  $y_i - y_i^U$ ) are scaled into the same change of “designer’s satisfaction”. In [8] the weighting coefficients are calculated as:

$$w_i^L = w_i^U = 1/(y_i^U - y_i^L) \quad (4)$$

This scaling is mostly useful, when the goal of design is to make response close to  $y_i^U$  or  $y_i^L$  – transforming the range  $[y_i^L, y_i^U]$  into  $[-1, 0]$  (see Fig. 2a). If the goal is to place response value inside  $[y_i^L, y_i^U]$ , as far as possible from the interval endpoints the following weighting coefficients are advantageous [9]:

$$w_i^L = 1/(y_i^T - y_i^L), \quad w_i^U = 1/(y_i^U - y_i^T) \quad (5)$$

with additional target values  $y_i^T \in (y_i^L, y_i^U)$  (see Fig. 2b for illustration). One can notice, that when  $y_i^T \searrow y_i^L$  the coefficient  $w_i^L$  dominates  $w_i^U$ , and when  $y_i^T \nearrow y_i^U$  the opposite holds.

Overall, the goal of design might be stated as a uniform increase of “designer’s satisfaction”, over all responses of interest – as measured by partial satisfaction indices (3). In mathematical terms we have the min-max problem, i.e. minimization of the largest satisfaction index:

TABLE II  
MEANS AND STANDARD DEVIATIONS OF INDEPENDENT NORMAL VARIABILITY SOURCES

Name	Description	Mean	Std. dev.
TOX	Oxide thickness [nm]	31.8	1.83
LDN	NMOS lateral diff. [nm]	348	6.04
WDN	NMOS width modulation [nm]	400	40
NSUBN	NMOS substrate doping [ $10^{14} \cdot \text{cm}^{-3}$ ]	144	8.69
LDP	PMOS lateral diff. [nm]	285	10
WDP	PMOS width modulation [nm]	400	40
XJP	Metallurgical junction depth [nm]	678	24.7
NSUBP	PMOS substrate doping [ $10^{14} \cdot \text{cm}^{-3}$ ]	602	41.9
RCC	CC variability [fF]	0	50

$$\min_{\mathbf{x} \in \mathbb{X}} \bar{\psi}(\mathbf{x}), \quad \text{where } \bar{\psi}(\mathbf{x}) \equiv \psi(\varphi(\mathbf{x}, \bar{\theta})) \quad (6)$$

$$\psi(\mathbf{y}) = \max_i (\max(\psi_i^L(\mathbf{y}), \psi_i^U(\mathbf{y}))) \quad (7)$$

Looking at Figure 2 we can see, that negative value of  $\bar{\psi}(\mathbf{x})$  assures, that all responses of a design with parameters  $\mathbf{x}$  are satisfying respective bounds:  $y_i \in (S_i^L, S_i^U)$ .

**Example 2.** In this example the nominal design problem (6) was solved with the mini-max solver of MATLAB (with estimation of derivatives) for the op-amp from Example 1. The objective function  $\bar{\psi}(\mathbf{x})$  was decreased from initial -0.115 to the final value around the global minimum of -1, with the cost of some 90 circuit simulations. Progress of intermediate solutions  $\bar{\psi}(\mathbf{x}^{(k)})$  is depicted in Fig. 3 with the distance to global objective function value:  $\Delta^{(k)} \equiv \bar{\psi}(\mathbf{x}^{(k)}) + 1$ . Design centering improved not only nominal circuit behavior (as measured with  $\bar{\psi}(\mathbf{x})$  value), but also statistical properties of the design. Fig. 4 depicts histograms of  $\psi(\mathbf{x}, \theta)$  obtained from Monte Carlo simulation with 2000 samples for the initial and centered design. It is seen, that nominal centering shifted distribution of quality index  $\psi$  such, that on average 89.4% of samples satisfied specification ( $\psi(\mathbf{x}, \theta) \leq 0$ ), while initially – only 36.25%. That kind of improvement of yield (average percentage of circuits satisfying specs) is frequent with nominal design. One should remember though, that the statistical improvement is only a “side-effect” of nominal design – and so cannot be guaranteed to occur; special design problem formulations, that specifically target yield or variability improvement (which are overviewed next), can be expected to find better solution.

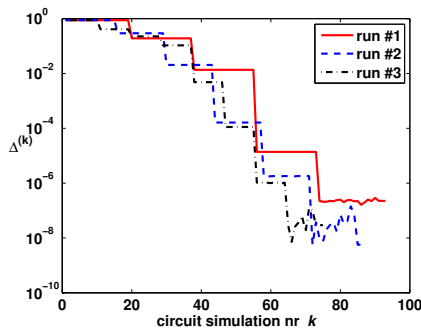


Fig. 3. Progress of several mini-max optimization runs for the example CMOS circuit: run #1 – optimization w.r.t. all 16 parameters, run #2 – w.r.t. 12 parameters, run #3 – w.r.t. 7 parameters.  $\Delta^{(k)} \equiv \bar{\psi}(\mathbf{x}^{(k)}) + 1$ .

Setting up the optimization problems raises an interesting question - which elements of the  $\mathbf{x}$  vector should be optimized. For the example op-amp the minimum value of  $\bar{\psi}(\mathbf{x})$  is equal -1, so at the optimum all 8 bounds (created by 4 two-sided specs) are active. Since we assumed 16 parameters to be optimized - there seem to be some degrees of freedom left, making the solution not unique. One might suspect that spare variables can cause bad numerical conditioning of the optimization process, and increase its computational cost. To get a feeling of the problem two other optimizations were run. In run #2 three pairs of tracking parameters were identified, reducing the number of optimized variables to 12. In run #3 (with 7

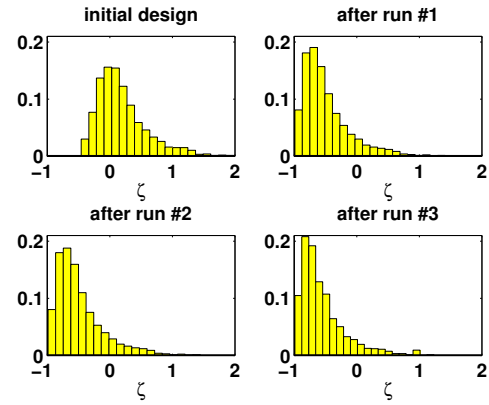


Fig. 4. Histograms of  $\zeta \equiv \psi(\mathbf{x}, \theta)$  samples, for initial and three optimum design points  $\mathbf{x}$  of the example CMOS op-amp.

parameters) all transistor lengths were additionally fixed (see Table 1 for details of parameter selection). As can be seen in Fig. 3 all runs found close neighborhood of the absolute minimum of  $\bar{\psi}(\mathbf{x})$  equal to -1. It can be seen, that indeed optimization w.r.t. all designable parameters was the slowest – mostly since for each design point candidate the optimizer had to make more circuit simulations to estimate objective function gradient ( $\dim(\mathbf{x}) + 1$  by single-side perturbations). For large accuracy ( $10^{-8}$ ) run #3 seems to be the best (due to numerical conditioning), but for medium accuracy ( $10^{-4}$ ) run #2 was marginally fastest.  $\square$

Prediction of computation complexity (as measured with the number of circuit simulations  $N_{sim}$ ) of nominal design centering is not trivial – as it depends on many factors: the number of design parameters  $n$ , number of independent design parameters  $n_x$ , number of active constraints (related both to the number  $m_y$  of partial quality indices determining value of the objective function, and active constraints determining the  $\mathbb{X}$  domain), non-linearity of  $\mathbf{x} \rightarrow \mathbf{y}$  relationship, selection of the initial design point and requested inaccuracy. In a single-CPU (serial processing) computing environment the centering run-time is determined by the total number of circuit simulations  $N_{sim}$ , necessary for finding optimum and making the optimizer stop. If  $\mathbf{x} \rightarrow \mathbf{y}$  dependence is linear and  $m_y \geq n$  the minimum  $N_{sim} = 2n + 2$ . If  $\mathbf{x} \rightarrow \mathbf{y}$  dependence can be approximated with a quadratic function – at least two approximations are needed to find and confirm optimum, so  $N_{sim} \geq (n + 1)(n + 2)$ . If the mini-max algorithm builds the quadratic approximation incrementally, then  $N_{sim}$  can be dependent on  $n_x < n$ . So, roughly  $N_{sim}$  is of the order of  $Kn_x^2$ , with  $n_x$ , where  $K$  is a small integer, increasing with non-linearity of response functions and requested accuracy.

When it is possible to perform gradient estimation, using  $N_{par} \leq n_x + 1$  parallel circuit simulation units (processors, cores) – ideally a speed-up of  $N_{par}$  can be achieved, so (at best) the nominal centering run-time scales linearly with  $n_x$ .

#### IV. VARIABILITY AWARE DESIGN CENTERING

Comparison of any two nominal designs  $\mathbf{x}^{(1)}$  and  $\mathbf{x}^{(2)}$  can be performed by just comparing respective values of  $\bar{\psi}(\mathbf{x}^{(1)})$

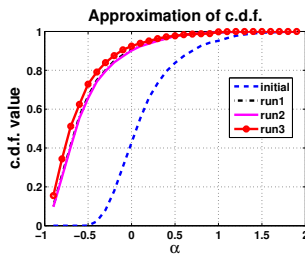


Fig. 5. Approximation of c.d.f.  $F_{\zeta}^{\mathbf{x}}(\alpha)$  of  $\zeta \equiv \psi(\mathbf{x}, \theta)$  for four design points ( $\mathbf{x}$ ): initial and 3 optimized (see Example 1).

and  $\bar{\psi}(\mathbf{x}^{(2)})$  (with  $\bar{\psi}(\mathbf{x})$  defined with (6)). The situation changes qualitatively, if we consider variability sources  $\theta$ . First, let us notice that if  $\psi(\mathbf{x}^{(1)}, \theta^{(j)}) < \psi(\mathbf{x}^{(2)}, \theta^{(j)})$  for some realization  $\theta^{(j)}$  of the random  $\theta$  vector – it might not be so for other realization of that random vector. So, for the two design points  $\mathbf{x}^{(1)}$  and  $\mathbf{x}^{(2)}$  whole distributions of two functions ( $\zeta^{(1)} \equiv \psi(\mathbf{x}^{(1)}, \theta)$  and  $\zeta^{(2)} \equiv \psi(\mathbf{x}^{(2)}, \theta)$ ) of random vector  $\theta$  have to be compared.

**Example 3.** For the initial design point  $\mathbf{x}^{(0)}$ , and for the three optimum points  $\hat{\mathbf{x}}^{(k)}$ ,  $k = 1, 2, 3$  of the example circuit optimized in Example 2, Monte Carlo simulation was performed with  $N_{MC} = 2000$  samples, and estimates of the cumulative density function (c.d.f.)  $F_{\zeta}^{\mathbf{x}}(\alpha)$  of random  $\zeta \equiv \psi(\mathbf{x}, \theta)$  were determined.  $F_{\zeta}^{\mathbf{x}}(\alpha)$  denotes probability, that values of  $\zeta \equiv \psi(\mathbf{x}, \theta)$  will not be larger than  $\alpha$ . The value  $F_{\zeta}^{\mathbf{x}}(0)$ , called yield, expresses mean ratio of the number of circuits satisfying specifications ( $y_i \in [S_i^L, S_i^U]$ ,  $i = 1, \dots, m$ , i.e.  $\zeta \leq 0$ ) to the whole populations of circuits (having the same design parameter values  $\mathbf{x}$ ).

We can see in Fig. 5, that all 3 runs of nominal design centering improved not only nominal performance – but also statistical distribution of circuit performance. For all 3 optimized designs the nominal performance index is practically the same:  $\bar{\psi}(\hat{\mathbf{x}}^{(i)}) \approx -1$ , yet we can see from Fig. 5, that the third optimized design is statistically the best – as the c.d.f. of  $\zeta$  is the largest – for each “satisfaction level”  $\alpha$ .  $\square$

Note, that the intuitive ranking of statistical quality of design points in the above example was made via comparison of c.d.f. According to this approach a design 1 ( $\mathbf{x} = \mathbf{a}$ ) is superior to a design 2 ( $\mathbf{x} = \mathbf{b}$ ) if

$$F_{\zeta}^{\mathbf{a}}(\alpha) \geq F_{\zeta}^{\mathbf{b}}(\alpha) \quad \text{for all } \alpha \quad (8)$$

This is so called “usual stochastic order”. Such an ordering is partial, as comparison of some random variables is not possible. E.g. for normal distribution comparable random variables have to have the same variance [10]. So for statistical design centering it is necessary to use less restrictive ways of imposing a partial order. In what follows two families of quality measures will be overviewed, together with selected optimization based techniques to solve resulting statistical design centering problems. For sake space the emphasis will be placed on the most efficient/promising approaches known at the moment.

### A. Optimization of Yield and Its Extensions

1) *Design yield and its Monte Carlo estimation:* One of the most important families of quality measures stems from the concept of yield, introduced already in examples 2 and 3. To simplify notation, let us assume, that a circuit characterized with a vector of designable parameters  $\mathbf{x}$  and a particular realization of the disturbances  $\theta$  is *acceptable* – if its responses  $\mathbf{y} = \varphi(\mathbf{x}, \theta)$  satisfy all lower-upper bounds ( $y_i \in [S_i^L, S_i^U]$ ,  $i = 1, \dots, m$ ), and *unacceptable* – otherwise. The set of all vectors  $\theta$  that result in acceptable circuit is called acceptability region (in the space of disturbances) and will be denoted  $\mathcal{A}_{\theta}(\mathbf{x})$ . Statistical percentage of acceptable designs in the whole population of circuits, having the same designable vector  $\mathbf{x}$  is called (design) yield. The yield can be mathematically expressed in many equivalent forms, e.g.:

$$Y(\mathbf{x}) = \int_{\mathcal{A}_{\theta}} f_{\theta}(\theta) d\theta = E_{\theta} \{ \chi(-\psi(\varphi(\mathbf{x}, \theta))) \} \quad (9)$$

where  $\chi(z) = 1$ , if  $z \geq 0$ , and 0 – otherwise;  $f_{\theta}(\theta)$  denotes p.d.f. of the statistical variability sources  $\theta$ , and  $E_{\theta} \{ \cdot \}$  – expected value w.r.t. random variable  $\theta$ . The measure is very popular, because of its clear meaning, and importance for economics of manufacturing process – even if its evaluation is computationally intensive and inaccurate. The most universal method of yield evaluation is Monte Carlo based. If  $N_P$  of  $N_{MC}$  independent samples of the disturbances  $\theta^{(m)}$ ,  $m = 1, \dots, N_{MC}$  make the circuit responses acceptable – the Monte Carlo yield estimate  $\hat{Y}$  and an estimate of its standard deviation  $\hat{\sigma}_{\hat{Y}}$  are, respectively:

$$\hat{Y} \equiv \frac{N_P}{N_{MC}}, \quad \hat{\sigma}_{\hat{Y}} \approx \sqrt{\frac{\hat{Y}(1 - \hat{Y})}{N_{MC} - 1}} \quad (10)$$

Let us note here, that such a yield estimation can be easily made parallel, providing speed-up of up-to  $N_{par} \leq N_{MC}$  with  $N_{par}$  independent simulation units (each one evaluating circuit response for a different disturbance instance  $\theta^{(m)}$ ).

**Example 4.** For the example CMOS op-amp Monte Carlo simulation was performed with  $N_{MC} = 2000$  samples for the initial design vector  $\mathbf{x}^{(0)}$ , and for the 3 vectors obtained by nominal design centering in Example 2 ( $\hat{\mathbf{x}}^{(1)}$ ,  $\hat{\mathbf{x}}^{(2)}$ ,  $\hat{\mathbf{x}}^{(3)}$ ). Yield estimates with  $3\sigma$  confidence interval were found as shown in Table III.

It is seen, that differences between yield estimates for optimized solutions are within  $3\sigma$  confidence intervals of yield estimates  $\hat{Y}$ . So, despite large number of circuit simulation used one cannot be certain if the run #3 gave the best solution. Much more samples (or more subtle statistical analysis) are

TABLE III  
ESTIMATES OF YIELD AND  $3\sigma$  CONFIDENCE INTERVAL FOR INITIAL DESIGN AND 3 OPTIMIZED DESIGNS OF CMOS OP-AMP

	$\hat{Y}[\%]$	$3\sigma_{\hat{Y}}[\%]$
initial design: $\mathbf{x}^{(0)}$	36.25	3.2
run #1 design: $\hat{\mathbf{x}}^{(1)}$	89.4	2.1
run #2 design: $\hat{\mathbf{x}}^{(2)}$	89.0	2.1
run #3 design: $\hat{\mathbf{x}}^{(3)}$	90.95	1.9

required to provide such a (statistical) certainty. Note, that Monte Carlo improves accuracy as  $1/\sqrt{N_{MC}}$ , so 10-fold decrease of  $\sigma_{\hat{Y}}$  requires 100 increase of samples  $N_{MC}$ .  $\square$

Since accuracy of a single yield estimation depends on the number of samples used – it is not obvious how to allocate a typically fixed number of available circuit simulations (for a design problem at hand) to a number of steps, which a yield optimization solver is to take. There are two opposite approaches. The first, called indirect (or large sample) optimization, uses a combination of a medium or large size sample Monte Carlo estimator of yield (and perhaps its gradient) and an optimizer (usually deterministic), which is tolerant of non-smoothness of objective function. The second, called direct (or small sample) optimization, uses a very small-size sample Monte Carlo estimator of yield gradient and a stochastic approximation type optimizer. The first approach uses pretty accurate estimates and deterministic search, while the other – inaccurate estimates and a stochastic optimization algorithm, which is filtering out the noise from estimates while searching for solution.

2) *Large random sample based yield optimization:* Fundamental properties of the large-sample (LS) approach to yield optimization are given in the next example.

**Example 5.** For the example CMOS op-amp Nelder-Mead non-gradient optimizer of MATLAB was used, to maximize Monte Carlo yield estimates (10). Initially three optimization runs were performed using  $N_{MC} = 50, 100$  and  $200$  samples per single objective function estimation. Progress of optimization is shown in Fig. 6 (curves marked LS).

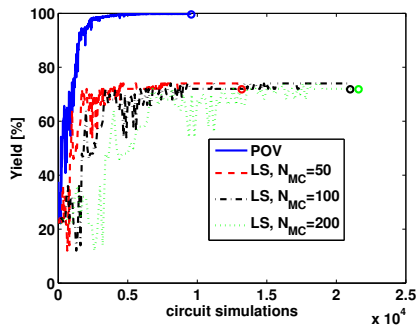


Fig. 6. Progress of yield optimization with large sample M. Carlo (LS) and propagation of variance (POV, see Example 7); actual 2000-sample estimates at optimum are marked with circles.

It might be surprising/disappointing, that the three “large sample” (LS) optimization runs haven’t found design points as good as that found with the nominal design centering in Example 2 (89.4% yield found in ca. 70 circuit simulations only).

Let us note, that despite use of a deterministic optimizer results of (LS) optimization are random – as they depend on the actual sequence of random numbers  $\theta^{(m)}$  used for yield estimation (10)<sup>1</sup>. To see if better solutions might be found for

<sup>1</sup>Because deterministic optimizer was used – the same random sequence was used for each Monte Carlo yield estimation of a particular optimization run.

different random sequences of the disturbance parameters  $\theta$ , optimization was repeated 100 times with different random sequences, Fig. 7 contains overlaid plots of the objective function value w.r.t. number of circuit simulation used, while Fig. 8 presents histograms and mean value/standard deviation of yield estimates evaluated (separately) at the design points found by the optimizer, after using  $N = 2000, 4000, 8000, 16000$  circuit simulations.

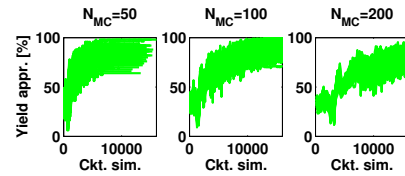


Fig. 7. Overlaid progress of 100 yield optimization runs; objective function used a)  $N_{MC} = 50$ , b) 100, c) 200 circuit simulations per single call.

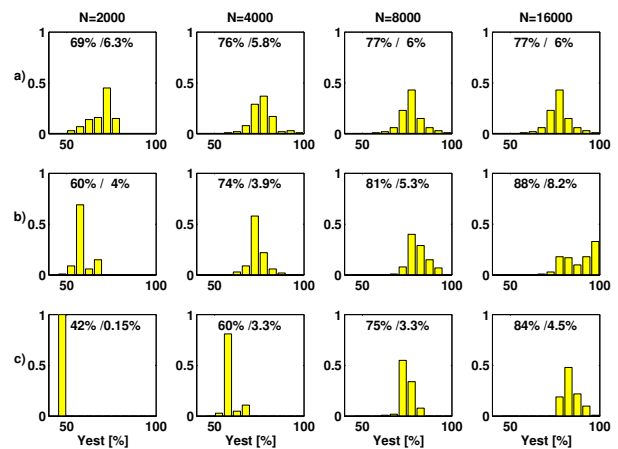


Fig. 8. Statistics of final yield of 100 runs of large sample M. Carlo optimizations with a) 50, b) 100, c) 200 samples per objective function call and different limits of available circuit simulations  $N$ . Numerical values of mean/standard deviation for each yield histogram are displayed as well.

It is seen, that dispersion of final yield is significant, so there could be found both better as well as much worse results, than these for nominal design. Furthermore,  $N_{MC} = 100$  seems to be a reasonable compromise between speed of convergence (the smaller the better) and accuracy (for  $N_{MC} \nearrow \infty$  Monte Carlo based results become accurate).  $\square$

In the last Example it was shown, that the number of the op-amp circuit simulation needed by LS yield optimization is 2 orders of magnitude greater than for the nominal design, and so can be prohibitive. To decrease computational cost some researchers advocate partial replacement of expensive circuit simulations with evaluation of some lower accuracy/local models (see e.g. [11], [12], [4], [13] for overviews) or application of specialized yield estimators [14] or deterministic quadratures [15] (see Section IV-A4 for some encouragement). Unfortunately, all these improvements introduce dependence of cost on the number of variability sources  $\theta$  - dependence, which is non-existent in the simple Monte Carlo experiment. Probably the only way to keep this independence is to parallelize yield estimation – as LS optimizer makes decision

after getting Monte Carlo yield estimate at a tentative design point. As was discussed in Section IV-A1 with  $N_{par} \leq N_{MC}$  circuit simulation units – the time for a single evaluation of the objective function is reduced roughly  $N_{par}$  times, and can be about the time of a single circuit simulation. Even with the parallelization one shouldn't expect, that run time of yield optimization will be closed to that of nominal design. One should be aware, that efficiency of a deterministic optimizer used in LS approach has to be lower than best optimizers capable of nominal design – since LS solver has to cope with non-smoothness of the objective function<sup>2</sup>

3) *Small random sample based yield optimization*: A small sample stochastic approximation based yield optimization, introduced in [16], does not use (relatively) accurate estimates of yield, but inaccurate estimates of yield gradient, averaging uncertain information concurrently with moving design point towards its optimum.

Perhaps the most favorable situation for gradient estimation occurs, when circuit parameters, that exhibits random variability, can be modeled as follows:

$$\mathbf{e} = \mathbf{x} + \theta \quad (11)$$

where the random vector  $\theta$  is Gaussian with mean  $\bar{\theta}$  and covariance matrix  $\mathbf{C}_\theta$ . Taking into account (1) the yield function can be shown [16], [17], to take the form:

$$\nabla Y(\mathbf{x}) = E_\eta \{ \chi(-\psi(\varphi(\mathbf{x}, \theta(\eta)))) \mathbf{L}_\theta^{-1} \eta \} \quad (12)$$

Note, that yield estimator can be derived not from information on probability of success in satisfying specs, but from (complementary) probability of failures. It was shown in [18], [19], that a combined yield estimator can be built:

$$\hat{Y}(\mathbf{x}) = w \hat{Y}(\mathbf{x}) + (1 - w)(1 - \hat{Y}_F(\mathbf{x})) \quad (13)$$

with  $w \in [0, 1]$  selected such, as to reduce estimator variance. In effect the following combined yield gradient estimator can be formulated:

$$\begin{aligned} \hat{\nabla} Y(\mathbf{x}) &= \text{diag} \left( \frac{1}{N_G \sigma_i^2} \right) \sum_{j=1}^{N_G} \varpi^{(j)} \mathbf{L}_\theta^{-1} \eta^{(j)} \quad (14) \\ \varpi^{(j)} &= w - 1 + \chi(-\psi(\varphi(\mathbf{x}, \theta(\eta^{(j)})))) \end{aligned}$$

Unfortunately, the formulae presented above cannot be directly used for centering of the example circuit, because random variability sources do not add to designable parameters<sup>3</sup>. For such design situations yield gradient can be estimated by a perturbation method [20], [6]. The method assumes application of extra random perturbations  $\mathbf{u}$  to designable parameters eg. Gaussian with 0 mean and diagonal covariance matrix  $\text{diag}(\sigma_i^2)$ . Formally, addition of the perturbations modifies the yield function, which becomes:

$$\hat{Y}(\mathbf{x}) = E_{\mathbf{u}} \{ Y(\mathbf{x} + \mathbf{u}) \} \quad (15)$$

<sup>2</sup>The Monte Carlo yield estimate (10) is a piece-wise constant function of design parameter vector  $\mathbf{x}$ .

<sup>3</sup>They could be used though, if transistor parameter dimensions (W,L) were assumed to be manufactured inaccurately, with Gaussian distributed inaccuracies of each parameter.

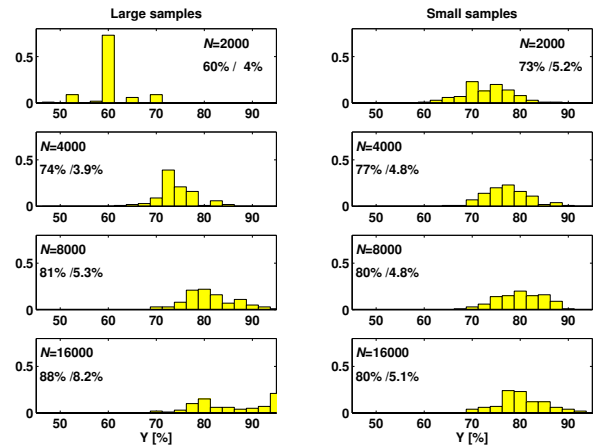


Fig. 9. Statistics of yield for 100 runs of large sample and small sample Monte Carlo optimizations with different limits of available circuit simulations  $N$ . Values of mean/standard deviation for each yield histogram are displayed for each case.

Now it is possible, to use formula (14) to estimate gradient of the modified yield function (15) in the following format:

$$\begin{aligned} \hat{\nabla} \tilde{Y}(\mathbf{x}) &= \text{diag} \left( \frac{1}{N_G \sigma_i^2} \right) \sum_{j=1}^{N_G} \varpi^{(j)} \mathbf{u}^{(j)} \quad (16) \\ \varpi^{(j)} &= w - 1 + \chi(-\psi(\varphi(\mathbf{x} + \mathbf{u}^{(j)}, \theta^{(j)}))) \end{aligned}$$

where  $\mathbf{u}^{(j)}$  are  $N_G$  independent random samples of Gaussian  $\mathbf{u}$ , and  $\theta^{(j)}$  – samples of original disturbances  $\theta$ .

**Example 6.** *Capability of the perturbation estimator (16) with the stochastic approximation algorithm based on [21] was tested against the optimization problem of the previous example.  $N_G = 2$ ,  $w = 0.95$ ,  $\sigma_i = 0.02$  were assumed.*

*Results, shown in Fig. 9, demonstrate clearly faster initial convergence of the small samples algorithm. However, if large number of samples is available – the large samples approach provides (on average) better solutions, with smaller spread and better distribution of the final yield. □*

*Let us note, that the small sample algorithm cannot take much advantage of parallel simulation capability, if available – because of small number of Monte Carlo used for a single yield gradient estimation. So, at presented version, it can be seen as inferior (to the LS algorithm) in parallel computation framework.*

4) *Optimization of deterministic yield approximation*: For designs with high yield (and so small variability of circuit responses) it is hard to get improvement with Monte Carlo based methods, because direction of improvement is found from a very small percentage of samples, which failed specs. For so called  $6\sigma$  design quality only two failures per 1 billion samples are expected, and so using Monte Carlo for yield estimation becomes non-practical.

For such high-yield cases deterministic yield formulae are of use. To see it let us assume, that distribution of each circuit response  $y_i$  is Gaussian, with mean value  $\bar{y}_i(\mathbf{x})$  and standard deviation  $\sigma_i(\mathbf{x})$ . Than probability of satisfying 2-side specs  $y_i^L \leq y_i \leq y_i^U$  (partial yield) can be calculated

by a deterministic formula:

$$Y_i^{LU}(\mathbf{x}) \equiv \Pr \{y_i^L \leq y_i(\mathbf{x}, \theta) \leq y_i^U\} = \frac{1}{2} \left( \operatorname{erf} \left( \frac{y_i^U - \bar{y}_i}{\sqrt{2}\sigma_i} \right) + \operatorname{erf} \left( \frac{\bar{y}_i - y_i^L}{\sqrt{2}\sigma_i} \right) \right) \quad (17)$$

where  $\operatorname{erf}(z) = 2/\sqrt{\pi} \int_0^z \exp(-u^2) du$  is the standard error function. Since inequality

$$\underbrace{Y(\mathbf{x})}_{1 - \sum_i (1 - Y_i^{LU}(\mathbf{x}))} \leq Y(\mathbf{x}) \leq \underbrace{\bar{Y}(\mathbf{x})}_{\min_i Y_i^{LU}(\mathbf{x})} \quad (18)$$

holds – thus, for designs with high yield and Gaussian distributed response values, the lower bound function  $\underline{Y}(\mathbf{x})$  can be a good candidate for maximization instead of exact yield value  $y(\mathbf{x})$ , which belongs to the interval  $[\underline{Y}(\mathbf{x}), \bar{Y}(\mathbf{x})]$ .

The same type of reasoning can be performed for other marginal p.d.f. of response functions:  $\mathbf{y} = \varphi(\mathbf{x}, \theta)$  - if only such a p.d.f. can be found for given non-linear circuit response function  $\varphi$ . To make the method implementable in practice some assumptions are made. First, the variability sources  $\theta$  are assumed to be Gaussian random variables with covariance matrix  $\mathbf{C}_\theta$  and  $\mathbf{0}$  means. Besides, dependence of each response value  $y_i$  on  $\theta$  has to be linear, i.e.  $\mathbf{y}(\mathbf{x}, \theta) = \bar{\mathbf{y}}(\mathbf{x}) + \mathbf{S}_\theta^y \theta$ , for some sensitivity matrix  $\mathbf{S}_\theta^y$ . Under these conditions the well known Propagation Of Variance (POV) can be used to calculate mean and covariance matrix of the response vector:

$$\bar{\mathbf{y}} = \mathbf{y}(\mathbf{x}, \mathbf{0}), \quad \mathbf{C}_y = \mathbf{S}_\theta^y \mathbf{C}_\theta (\mathbf{S}_\theta^y)^T \quad (19)$$

**Example 7.** The above presented combination of deterministic yield estimation (17-18) and POV approximation (19) of  $\bar{\mathbf{y}}$  and  $\mathbf{C}_y$ , was used for design centering of the example CMOS op-amp. The objective function was not really the lower bound  $\underline{Y}(\mathbf{x})$ , but a heuristic function:

$$F(\mathbf{x}) = (1 - c) \cdot \bar{Y}(\mathbf{x}) + c \cdot \max(\underline{Y}(\mathbf{x}), 0), \quad c = \bar{Y}(\mathbf{x}) \quad (20)$$

which for low yield is close to the upper bound  $\bar{Y}$ , and for high yield – close to the lower bound  $\underline{Y}$ . Results of optimization (with the same Nelder-Mead non-gradient optimizer of MATLAB which was used for LS optimization in Example 5) are shown in Fig. 6 as POV curve. Rapid progress and high final yield are easily seen. Even though LS optimization can reach similar progress (see Fig. 7) for some sequences of random numbers, but the POV based optimization does not exhibit such performance variation, making the approach highly competitive.

It is important to stress limitations of this approach, though. One-sided difference approximation of the sensitivity matrix was used, so a single objective function evaluation involved  $n_\theta + 1$  circuit simulations (10 for the example case). For more realistic statistical modeling of the op-amp (including all local variability sources) the computational cost, which increases linearly with the number of variability sources  $n_\theta$ , might be much higher. Fortunately, calculation of sensitivity matrix for POV can be parallelized - making run-time almost independent on  $n_\theta$ , given sufficiently large number of concurrent circuit simulation units.

More important limitation is, that linearity assumption might not be valid, and even POV including quadratic terms might

not be sufficient to prevent large response distribution inaccuracy, that could mislead the optimization process. It is known [22], that for circuits with strong frequency dependent feedback loop (e.g. filters) non-linearity of circuit response functions can be so strong, that acceptability region is not only non-convex, but might contain holes and disjoint sub-areas. Such a behavior invalidates accuracy of both nominal design centering, as well as POV based yield optimization. Thus, further research is needed, to formulate cost-effective validity test for POV – to make its use safe<sup>4</sup>. To see, that the problem is non-trivial – Fig. 10 presents joint and marginal distribution of two responses of the example CMOS op-amp. Marginal distributions are not Gaussian, besides there exists a non-linear correlation among the response values. Yet, the optimization was successful.  $\square$

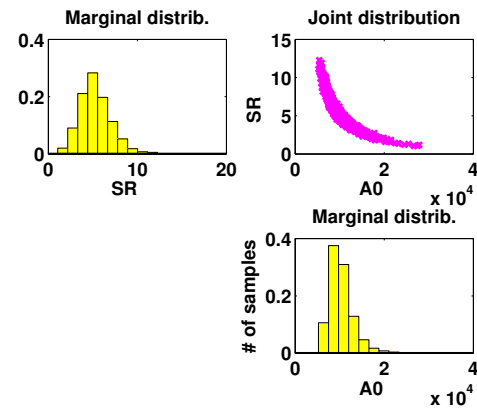


Fig. 10. Joint and marginal distributions of two responses of the circuit optimized in Example 1. 2000 samples were used in the Monte Carlo simulation (at the design point from run1).

Another approach to design centering at high yields was proposed by the author in [19]. Contrary to standard yield – the objective function, called the income index, is an expected value of a smooth function of the scaled worst response  $\psi$ , and not a step function  $\chi(\psi(\cdot))$  which is used in the yield formulation (9). In consequence for Gaussian variability sources we have a smooth Monte Carlo estimate of the objective function, and not piecewise-constant – as is the case for the yield (10). Interesting implementations of the income index might be found, e.g. in [24], [9], [25].

### B. Worst-Case Optimization and Its Extensions

Originally the worst-case (WC) optimization aimed at finding the designable parameters vector  $\mathbf{x}$  such, that all specifications were satisfied, i.e.  $\psi(\mathbf{x}, \theta) \leq 0$  for all allowable disturbances, i.e.  $\forall \theta \in \Theta$  for some compact domain of disturbances  $\Theta \subset \mathbf{R}^{n_\theta}$ . Typically such a formulation might not have a solution or have infinite many. To resolve the problem the domain of disturbances was parametrized (scaled) with a set of parameters, called tolerances ( $\varepsilon_i$ ). To avoid trivial

<sup>4</sup>In [23] a local non-linear approximation of  $\theta \rightarrow \mathbf{y}$  mapping is dynamically constructed, followed by an inexpensive Monte Carlo estimation of response moments. This addresses the problem of non-Gaussian distribution of responses partially, since estimation of yield is stochastic and not deterministic.

solution with  $\varepsilon_i = 0$  – a cost function  $F(\mathbf{x}, \varepsilon)$  was used, which was decreasing with increase of tolerances, but which could also depend on vector  $\mathbf{x}$ . Altogether a difficult optimization problem with infinite number of constraints resulted:

$$\min_{\mathbf{x} \in \mathbb{X}} F(\mathbf{x}, \varepsilon), \quad \text{s.t.} : \psi(\varphi(\mathbf{x}, \theta)) \leq 0 \quad \forall \theta \in \Theta(\varepsilon) \quad (21)$$

Generally, checking of the constraints for a single value of  $\mathbf{x}$  vector is equivalent to solution of an auxiliary global optimization problem:

$$\bar{\psi}(\mathbf{x}) \equiv \max_{\theta \in \Theta(\varepsilon)} \psi(\varphi(\mathbf{x}, \theta)) \quad (22)$$

and might be very difficult for non-convex acceptability regions. In [26] the worst-case problem was re-thought, giving rise to so called Realistic Worst-Case (RWC) formulation [27], [28], [29], [30]. The RWC formulation is in fact a WC formulation, with several assumptions added, so that its solution becomes computationally attractive and statistically meaningful.

The traditional WC was oriented towards circuits built with pre-tested discrete components having individually controlled tolerances, and so it does not fit integrated circuit design. RWC approach acknowledges, that dispersions of disturbances are not controllable – typically assuming Gaussian distribution with fixed mean and covariance. For such unbounded domains  $\Theta(\alpha)$  was re-defined, to denote a set of all points, for which p.d.f. of disturbance distribution is not smaller than some probability value  $p_{min}(\alpha)$  dependent on  $\alpha$ .

Then it was possible to follow the main mission of WC formulation – to inscribe the largest  $\Theta(\alpha)$  into the acceptability region  $\mathcal{A}_\theta$ . Formally:

$$\max_{\alpha, \mathbf{x} \in \mathbb{X}} \alpha, \quad \text{s.t.} : \psi(\varphi(\mathbf{x}, \theta)) \leq 0 \quad \forall \theta \in \Theta(\alpha) \quad (23)$$

Let us note, that in the WC formulation (21) there is inherent a guarantee, that for each realization of variability sources  $\theta$  – resulting circuit responses are *acceptable*. 100% certainty is vital for safety critical applications, but not for mass products. In this respect RWC (23) is indeed more realistic, as it wants to assure, that for at least  $p_{min}(\alpha) \cdot 100\%$  realizations of variability sources  $\theta$  – resulting circuit responses are acceptable. In a sense RWC increases some bound on design yield, admitting that it is lower than 100%. Thus RWC provides a link between yield optimization and traditional worst-case design.

For Gaussian disturbances, defined as in (1), the parametrized  $\Theta(\alpha)$  set can be defined as:

$$\Theta(\alpha) = \{\theta \in R^{n_\theta} \mid \theta - \bar{\theta} = L_\theta \eta, \|\eta\| \leq \alpha\} \quad (24)$$

while the related minimum probability can be calculated from:

$$p_{min}(\alpha) = \frac{1}{\sqrt{(2\pi)^{n_\theta}}} \exp\left(-\frac{\alpha^2}{2}\right) \quad (25)$$

For other distributions parametrization can be done similarly (see e.g. [27], [28] for examples).

Solution methods of the RWC problem take into account each of design specification separately, finding worst-case conditions (called limit parameters in [26]) as follows:

$$\theta^{WCz,i}(\mathbf{x}, \alpha) = \arg \max_{\theta \in \Theta(\alpha)} \psi_i^z(\varphi_i(\mathbf{x}, \theta)) \quad (26)$$

where  $z \equiv L$  for lower specification ( $y_i \geq y_i^L$ ), and  $z \equiv U$  for upper specification ( $y_i \leq y_i^U$ ). To reduce computational

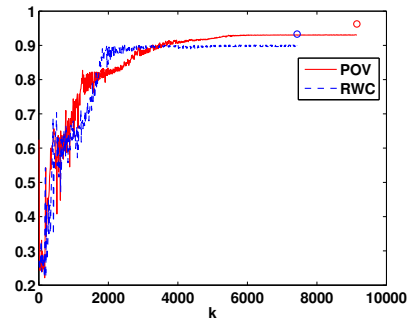


Fig. 11. Comparison of two RWC type centering: using partial yields (POV) and using design margins (RWC).

complexity RWC proponents assume linearity of responses  $y$  w.r.t variability sources:

$$\mathbf{y} = \bar{\mathbf{y}} + \mathbf{S}_\theta^y(\theta - \bar{\theta}) \quad \text{for all } \theta \in \Theta(\alpha) \quad (27)$$

Substituting (27) and (3) in (26) the worst-case conditions can be expressed as:

$$\theta^{WCz,i}(\mathbf{x}, \alpha) - \bar{\theta} = \iota_z \frac{\mathbf{a}}{\|\mathbf{a}\|} \alpha, \quad \mathbf{a} = (\mathbf{S}_\theta^{y_i} \mathbf{L}_\theta)^T \quad (28)$$

where  $\iota_L = -1$  (lower specification) and  $\iota_U = 1$  (upper specification). Thus it is possible, to find values of  $\alpha$  scaling factor, which make responses meet respective bounds[27].  $|\alpha| > 0$  can be seen a measure of a distance of worst-case response from respective specification bound, for  $\alpha > 0$  the worst-case response value is acceptable.

For lower bounds the distance  $\alpha_{WCL,i}$  can be found from the condition:  $y_i^L = y_i(\mathbf{x}, \bar{\theta}) - \alpha \mathbf{S}_\theta^{y_i} \mathbf{L}_\theta / \|\mathbf{S}_\theta^{y_i} \mathbf{L}_\theta\|$ , while for upper constraints:  $y_i^U = y_i(\mathbf{x}, \bar{\theta}) + \alpha \mathbf{S}_\theta^{y_i} \mathbf{L}_\theta / \|\mathbf{S}_\theta^{y_i} \mathbf{L}_\theta\|$ . Thus the worst case distances are:

$$\alpha_{WCL,i} = \frac{y_i(\mathbf{x}, \bar{\theta}) - y_i^L}{\|\mathbf{S}_\theta^{y_i} \mathbf{L}_\theta\|}, \quad \alpha_{WCU,i} = \frac{y_i^U - y_i(\mathbf{x}, \bar{\theta})}{\|\mathbf{S}_\theta^{y_i} \mathbf{L}_\theta\|} \quad (29)$$

The above procedure produces a set of  $2m$  “distances to non-acceptability”, which should be maximized in the course of RWC design centering. In [27] the objective function for design centering is formed directly from the worst case distances:

$$F = \sum_{i \in \{T\}_L} [\exp(-\alpha_{WCL,i})]^2 + \sum_{i \in \{T\}_U} [\exp(-\alpha_{WCU,i})]^2 \quad (30)$$

Other ways of scalarization can be used as well, e.g. worst (smallest) value of the  $\alpha$  distance can be maximized.

**Example 8.** For the example CMOS op-amp realistic-worst case optimization was performed. In Fig. 11 optimization of RWC distances (29) is compared to optimization of deterministic estimates of partial yields using POV (20). It is seen, that the techniques have quite similar performance.  $\square$

Let us note, that the values  $\alpha_{WCL,i}$ ,  $\alpha_{WCU,i}$  are directly related to partial yields (17), which can be found for each specification separately from p.d.f. of  $\eta$ :

$$Y_i^z = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\alpha_{WCz,i}} \exp\left(-\frac{\xi^2}{2}\right) d\xi, \quad \text{where } z \equiv L \text{ or } U \quad (31)$$



In effect we end up with a vector of quality indices  $Y_i^L, i \in \{\mathcal{T}\}_L, Y_i^U, i \in \{\mathcal{T}\}_U$  to be optimized. Let us notice, that the RWC formulation coincides with the previously reviewed POV based yield optimization (Example 7) – exactly, when using  $\underline{Y}(\mathbf{x})$  (defined in (19) as the objective function. No wonder, that computational complexity is similar to that of POV based yield optimization. The main computational cost is generated by the estimation of sensitivity matrix  $\mathbf{S}_\theta^y$ , and so it grows linearly with the number of disturbances  $n_\theta$ , but the calculation can be easily made parallel.

## V. SUMMARY

The paper overviewed three most important groups of design centering formulations: nominal design, yield optimization, RWC design. For each group most prominent solution techniques were presented, and characterized comparatively – with the aid of the same standard CMOS op-amp design example and a simplified computational complexity analysis (to enable generalization). The author hasn't intended to suggest reader the best formulation of design centering or the best solution method. Rather, trade-offs were revealed, and exemplified – so presented comments might be of help, when a reader is to approach centering of a specific circuit.

For statistical design the most important problems are related to efficient characterization of variability of circuit responses  $\mathbf{y}$ . A simplistic view contrasts generic, but costly Monte Carlo based calculations with approximate but inexpensive POV calculations. It was demonstrated in this paper, that such a view is justified for design with small number of variability sources. If the statistical model of example circuit included not only global variability sources (correlations of parameters of a CMOS transistor model, separately for N and P types) but all local differences between these models, and variability of transistor W and L – the cost of POV could become comparable. For larger circuits the computational cost of single POV based objective function evaluation might easily exceed that of corresponding calculation with the Monte Carlo method.

Another popular view is, that statistical design is so much more time consuming, than nominal design, that it should be tried only after all else failed. The paper demonstrated, that essentially all presented solution techniques have large (although different) potential for easy parallelization. Thus large computation time doesn't have to translate into large turn-around time of design centering jobs anymore. Yield or RWC optimization with sufficiently rich parallel computing environment can take comparable amount of time to that of nominal design centering executed on a single CPU.

## REFERENCES

- [1] J. Chen and M. Styblinski, "A systematic approach to statistical modeling and its application to CMOS circuits," in *Proc. ISCAS*, 1993, pp. 1805–1808.
- [2] C. Michael and M. Ismail, *Statistical modeling for computer-aided design of MOS VLSI circuits*. Kluwer Acad. Publ., 1993.
- [3] M. Conti, P. Crippa, S. Orcioni, and C. Turchetti, "Parametric yield formulation of MOS IC's affected by mismatch effect," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 18, no. 5, pp. 582–596, May 1999.
- [4] K. Low and S. Director, "An efficient methodology for building macro-models of IC fabrication processes," *IEEE Trans. Comput.-Aided Design*, vol. 8, no. 12, pp. 1299–1313, December 1989.
- [5] S. Nassif, A. Strojwas, and S. Director, "FABRICS II: a statistically based IC fabrication process simulator," *IEEE Trans. Comput.-Aided Design*, vol. 3, no. 1, pp. 20–46, January 1984.
- [6] M. Stybliński and L. Opalski, "Algorithms and software tools for IC yield optimization based on fundamental fabrication parameters," *IEEE Trans. Comput.-Aided Design*, vol. 5, no. 1, pp. 79–89, January 1986.
- [7] M. Pehl and H. Graeb, "RaGAzi: A random and gradient-based approach to analog sizing for mixed discrete and continuous parameters," in *Proc. ISIC*, 2009, pp. 113–116.
- [8] W. Nye, D. Riley, and A. Sangiovanni-Vincentelli, "DELIGHT.SPICE: an optimization-based system for the design of integrated circuits," *IEEE Trans. Comput.-Aided Design*, vol. 7, no. 4, pp. 501–519, April 1988.
- [9] M. Stybliński, "Design for circuit quality: yield maximization, minimax and Taguchi approach," in *Proc. ICCAD*, 1990, pp. 112–115.
- [10] A. Müller, "Stochastic ordering of multivariate normal distributions," *Ann. Inst. Statist. Math.*, vol. 53, no. 3, pp. 567–575, 2001.
- [11] G. Gielen and R. Rutenbar, "Computer-aided design of analog and mixed-signal integrated circuits," *Proc. IEEE*, vol. 88, no. 12, pp. 1825–1854, December 2000.
- [12] R. Rutebnar, G. Gielen, and J. Roychowdhury, "Hierarchical modeling, optimization and synthesis for system-level analog and RF designs," *Proc. of the IEEE*, vol. 95, no. 3, pp. 640–669, March 2007.
- [13] J. Zhang and M. Styblinski, *Yield and variability optimization of integrated circuits*. Kluwer Acad. Publ., 1995.
- [14] R. Spence and R. Soin, *Tolerance design of electronic circuits*. Addison-Wesley, 1988.
- [15] L. Opalski, "Deterministic optimization of stochastic circuit quality measures," in *Proc. Nat'l Conf. on Ckt. Th. and El. Ckts. (in Polish)*, Polanica Zdrój., 1994, pp. 579–584.
- [16] M. Stybliński and A. Ruszczyński, "Stochastic approximation approach to production yield optimization," *Electronic Letters*, vol. 19, no. 8, pp. 300–301, 1986.
- [17] B. Batalov, Y. Belyakov, and F. Kurmaev, "Some methods for statistical optimization of integrated microcircuits," *Soviet Microelectronics (USA)*, vol. 7, no. 4, pp. 228–238, July-August 1978.
- [18] J. Lonc and L. Opalski, "Yield optimization by stochastic programming with adaptive yield gradient estimator," in *Proc. Nat'l Conf. on Ckt. Th. and El. Ckts. (in Polish)*, Sulejów, Poland, 1982, pp. 141–156.
- [19] L. Opalski and M. Styblinski, "Generalization of yield optimization problem: maximum income approach," *IEEE Trans. Comput.-Aided Design*, vol. 5, no. 2, pp. 346–360, April 1986.
- [20] R. Rubinstein, *Simulation and the Monte Carlo Method*. N. York: J. Wiley & Sons, 1981.
- [21] A. Ruszczyński and W. Syski, "A method of aggregate stochastic sub-gradients with on-line stepsize rules for convex stochastic programming problems," *Math. Progr. Study*, vol. 28, pp. 113–131, 1986.
- [22] J. Ogrodzki, "One-dimensional orthogonal search – a method for a segment approximation to acceptability region," *Circ. Th. and Applications*, vol. 14, pp. 181–194, 1986.
- [23] A. Dharchoudhury and S. Kang, "Worst-case analysis and optimization of VLSI circuit performances," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 14, no. 4, pp. 481–492, April 1995.
- [24] L. Opalski, "Yet another approach to statistical circuit design – stochastic minimax," in *Proc. ISCAS*, N. Orleans, LA, USA, 1990, pp. 2264–2267.
- [25] L. Opalski and M. Styblinski, "GOSSIP – a generic system for statistical circuit design," in *Proc. Euro-DAC*, Hamburg, Germany, 1992, pp. 572–577.
- [26] G. Müller-L., "Limit parameters: the general solution of the worst-case problem for the linearized case," in *Proc. ISCAS*, N. Orleans, LA, USA, 1990, pp. 2256–2269.
- [27] K. Antreich, H. Graeb, and C. Wieser, "Circuit analysis and optimization driven by worst-case distances," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 13, no. 1, pp. 57–71, January 1994.
- [28] A. Seifi, K. Ponnambalan, and J. Vlach, "A unified approach to statistical design centering of integrated circuits with correlated parameters," *IEEE Trans. Circuits Syst. I*, vol. 46, no. 1, pp. 190–196, January 1999.
- [29] H. Graeb, *Analog Design Centering and Sizing*. Springer, 2007.
- [30] R. Schwencker, F. Schenkel, M. Pronath, and H. Graeb, "Analog circuit sizing using adaptive worst-case parameter sets," in *Proc. DATE Conf. and Exhib.*, 2002, pp. 581 – 585.