ELEKTROTEHNIŠKI VESTNIK 82(4): 191–196, 2015
ORIGINAL SCIENTIFIC PAPER

Parametric yield optimization with mesh adaptive direct search

Árpád Bűrmen[1]

[1]*Faculty of Electrical Engineering, University of Ljubljana, Tržaška cesta 25, 1000 Ljubljana, Slovenia*

[1] *E-mail: arpad.buermen@fe.uni-lj.si*

Abstract. Random manufacturing process variations can affect the performance of an integrated circuit to the extent that a significant number of the manufactured circuits must be discarded because they fail to satisfy the specifications. To increase the yield, random variations must be taken into account in the design phase. This can be achieved by choosing appropriate values for the parameters accessible to the circuit designer. This process (circuit sizing) can be automated by means of parametric optimization. As simulators do not compute sensitivities, derivative-free optimization algorithms, like mesh adaptive direct search (MADS), are well suited for optimizing circuits. We propose an MADS-based approach for finding a circuit that satisfies the minimum yield requirement. The approach is tested on two integrated circuit-sizing problems. The results demonstrate its effectiveness and speed.

Keywords: circuit sizing, yield analysis, yield optimization, mesh adaptive direct search

Optimizacija izplena z adaptivnim mrežnim direktnim optimizacijskim postopkom

Naključne variacije postopka izdelave integriranih vezij lahko vplivajo na lastnosti vezja do te mere, da precejšnje število izdelanih vezij ne izpolnjuje specifikacij. Ta vezja morajo biti zavržena. Da bi povečali izplen, moramo te variacije upoštevati pri načrtovanju vezja. To lahko dosežemo z izbiro ustreznih vrednosti načrtovalskih parametrov med dimenzioniranjem vezja. Le tega lahko avtomatiziramo s pomočjo parameterske optimizacije. Ker simulatorji ne računajo občutljivosti, so brezgradientni postopki, kot so adaptivni mrežni optimizacijski postopki, primerni za optimizacijo vezij. V prispevku predlagamo pristop k dimenzioniranju vezij, ki izpolnjujejo zahtevo po nekem minimalnem izplenu. Predlagani pristop preizkusimo na načrtovanju dveh integriranih vezij. Rezultati kažejo, da je pristop učinkovit in hiter.

1 INTRODUCTION

Due to random variations of the integrated circuit (IC) manufacturing process not all produced circuits satisfy the design requirements over the declared range of operating conditions resulting in a yield that is lower than 100%. With shrinking transistor dimensions, the effect of these variations is increasing with every new process node. Therefore, it is necessary to account for random manufacturing process variations during the design stage [1], [2], [3].

The effect of process variations can be reduced by increasing the transistor size that contributes most to the variability of the circuit performance. On the other hand, increased transistor sizes result in a larger and more expensive circuit. The process of choosing the transistor dimensions (circuit sizing) can be automated by using optimization algorithms [4]. Unfortunately, computing the metric subject to optimization (yield) is by itself a computationally intensive task, particularly when classical methods, like Monte Carlo analysis, are employed. Furthermore, Monte Carlo analysis requires a large number of samples for reaching a reasonable level of confidence when the circuit yield is close to 100%.

Random variations of the manufacturing process can be modeled with statistical parameters. Yield estimation can be significantly accelerated if deterministic methods are used [5]. These methods involve an optimization in the space of statistical and operating parameters. The resulting worst-case point can be used for estimating the circuit yield. Finding the optimal values of the circuit design parameters (i.e. parameters that can be chosen by the circuit designer) is also an optimization problem. This task is commonly referred to as circuit sizing.

Mesh adaptive direct search (MADS) [6] is a family of optimization algorithms that do not require the derivatives of the function subject to optimization. Because most circuit simulators do not compute sensitivities (i.e. derivatives), MADS is a good candidate for estimating the yield with a deterministic approach [5] as well as

Received 30 June 2015
Accepted 20 July 2015

finding the optimal design parameter values.

The paper is organised as follows. Section 2 establishes the connection between the worst-case distance and the circuit yield. Section 3 introduces a methodology for automating the circuit-sizing process with the goal of achieving the target yield. Section 4 gives a brief overview of MADS used for estimating the circuit yield and sizing the circuit. The results obtained on two circuit-sizing problems are given in Section 5.

2 Worst-case performance and parametric yield

The m performances of a circuit (e.g. gain, phase margin, swing, etc.) are given by vector $\mathbf{f}$. Every performance f_i depends on three groups of parameters: operating parameters $\mathbf{x}_O$, statistical parameters $\mathbf{x}_S$, and design parameters $\mathbf{x}_D$. The operating parameters define the environment in which a circuit operates (e.g. ambient temperature, supply voltage, bias current, etc.). The statistical parameters model random variations of the manufacturing process). Finally, the design parameters are the ones that can be adjusted by a designer (e.g. transistor channel widths and lengths, resistances of resistors, etc.).

A circuit satisfies the design requirements at a particular combination of the operating, statistical, and design parameters if all performances satisfy inequalities of the form $f_i(\mathbf{x}_O, \mathbf{x}_S, \mathbf{x}_D) \geq G_i$, where G_i is the target value.

Typically, a circuit must satisfy all the design requirements over a given range of the operating parameters. This range is specified with lower ($\mathbf{x}_O^L$) and upper ($\mathbf{x}_O^H$) bounds on the operating parameters. Let $\mathbf{x}_O^{W,i}(\mathbf{x}_S, \mathbf{x}_D)$ denote the operating parameters corresponding to the worst performance at given statistical and design parameters.

$$\mathbf{x}_O^{W,i}(\mathbf{x}_S, \mathbf{x}_D) = \arg \min_{\mathbf{x}_O^L \leq \mathbf{x}_O \leq \mathbf{x}_O^H} f_i(\mathbf{x}_O, \mathbf{x}_S, \mathbf{x}_D). \quad (1)$$

The inequality applies to vectors component-wise. To simplify the notation, we define

$$f_i^W(\mathbf{x}_S, \mathbf{x}_D) = f_i\left(\mathbf{x}_O^{W,i}(\mathbf{x}_S, \mathbf{x}_D), \mathbf{x}_S, \mathbf{x}_D\right) \quad (2)$$

For given design parameter values ($\mathbf{x}_D$) point $\mathbf{x}_S$ in the space of the statistical parameters belongs to the acceptance region of f_i (denoted by $\mathbf{x}_S \in \mathcal{A}_i(\mathbf{x}_D)$) if $f_i^W(\mathbf{x}_S, \mathbf{x}_D) \geq G_i$.

The statistical parameters originate from random variations of the manufacturing process. By transforming the process parameters, one can obtain independent normally-distributed random variables with zero mean and variance one. We refer to these variables as statistical parameters. The joint probability density of the statistical parameters is

$$p(\mathbf{x}_S) = (2\pi)^{-n_S/2} \mathrm{e}^{-\|\mathbf{x}_S\|^2/2}, \quad (3)$$

where n_S denotes the number of the statistical parameters. The origin in the space of the statistical parameters ($\mathbf{x}_S = \mathbf{0}$) corresponds to the nominal process parameters.

If a particular $\mathbf{x}_S$ does not belong to the acceptance region of f_i, the manufactured circuit corresponding to $\mathbf{x}_S$ must be discarded. Consequently the parametric yield of a circuit drops below 100%. The designer tries to maximize the acceptance region (and consequently the parametric yield) by adjusting the design parameters. The parametric yield of f_i can be computed by integrating (3) over $\mathcal{A}_i(\mathbf{x}_D)$ as

$$Y_i(\mathbf{x}_D) = \int_{\mathbf{x}_S \in \mathcal{A}_i(\mathbf{x}_D)} p(\mathbf{x}_S)\, d\sigma, \quad (4)$$

where $d\sigma$ is a differential volume element in the space of the statistical parameters. This integral cannot be expressed analytically. Usually, the numerical approaches, like Monte Carlo analysis, are used for estimating (4).

The worst-case point $\mathbf{x}_S^{W,i}(\mathbf{x}_D)$ is the point on the boundary of the acceptance region closest to the origin of the statistical parameters. Computing the worst-case point is an optimization problem given by

$$\mathbf{x}_S^{W,i} = \begin{cases} \arg\min_{\mathbf{x}_S \notin \mathcal{A}_i(\mathbf{x}_D)} \|\mathbf{x}_S\|^2, & \mathbf{0} \in \mathcal{A}_i(\mathbf{x}_D) \\ \arg\min_{\mathbf{x}_S \in \mathcal{A}_i(\mathbf{x}_D)} \|\mathbf{x}_S\|^2, & \text{otherwise.} \end{cases} \quad (5)$$

The distance of the worst-case point from the origin is reflected in the worst-case distance which is defined as

$$\beta_i(\mathbf{x}_D) = \begin{cases} \|\mathbf{x}_S^{W,i}(\mathbf{x}_D)\|, & \mathbf{0} \in \mathcal{A}_i(\mathbf{x}_D) \\ -\|\mathbf{x}_S^{W,i}(\mathbf{x}_D)\|, & \text{otherwise.} \end{cases} \quad (6)$$

Figure 1 illustrates the worst-case point and the worst-case distance when $\mathbf{0} \in \mathcal{A}_i(\mathbf{x}_D)$. It is possible to compute a good analytical approximation of (4) by linearizing the circuit performance in the neighborhood of the worst-case point $\mathbf{x}_S^{W,i}(\mathbf{x}_D)$. Integration of (3) over the acceptance region obtained with a linearized circuit performance (shaded in light grey) results in the approximate yield

$$\tilde{Y}_i(\mathbf{x}_D) = \frac{1}{2}\left(1 + \operatorname{erf}\left(\beta_i(\mathbf{x}_D)/\sqrt{2}\right)\right) \approx Y_i(\mathbf{x}_D). \quad (7)$$

The error of this approximation is equal to the integral of (3) over the region shaded in dark grey in Figure 1. In most cases, this error is small [5]. Therefore, it is reasonable to expect that maximizing the worst-case distance maximizes the parametric yield.

3 Finding a circuit with a given minimum yield

Suppose one wants to find the design parameters for which

$$Y_i(\mathbf{x}_D) \geq Y_0, \quad i = 1, 2, ..., m, \quad (8)$$

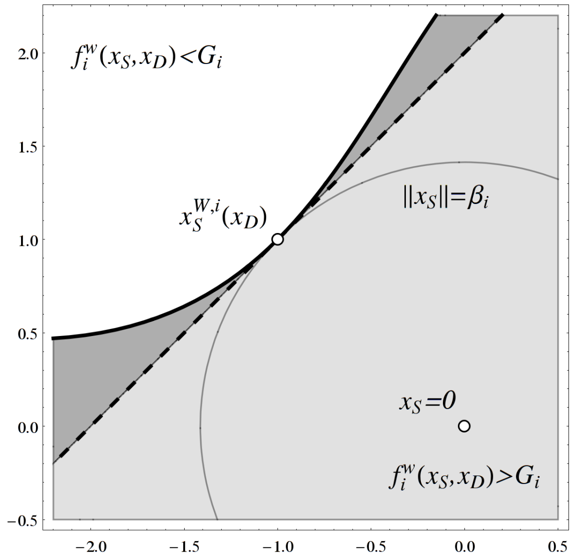

Figure 1. Worst case point $\mathbf{x}_S^{W,i}$ in the space of the statistical parameters for $n_S = 2$ at the design parameters given by $\mathbf{x}_D$. The boundary of the acceptance region (shaded) is depicted by a thick line. Linearization of the boundary at the worst-case point (dashed line) makes it possible to analytically compute the approximate yield. The error corresponds to the integral of (3) over the region shaded in dark grey.

where Y_0 is the target yield. By replacing Y_i with $\tilde{Y}_i$, we can approximate (8) with

$$\beta_i(\mathbf{x}_D) \geq \beta_0, \quad i = 1, 2, ..., m, \tag{9}$$

where β_0 is the worst case-distance that corresponds to approximate yield Y_0. It can be obtained by solving

$$Y_0 = \frac{1}{2}\left(1 + \operatorname{erf}\left(\beta_0/\sqrt{2}\right)\right). \tag{10}$$

For (9) to hold, the m worst-case points must satisfy $\|\mathbf{x}_S\| \geq \beta_0$. Consequently,

$$f_i^W(\mathbf{x}_S, \mathbf{x}_D) \geq G_i \quad \text{for} \quad \|\mathbf{x}_S\| \leq \beta_0 \quad \text{and} \quad i = 1, 2, ..., m. \tag{11}$$

Requirement (11) can be reformulated as

$$\min_{\substack{\|\mathbf{x}_S\| \leq \beta_0 \\ \mathbf{x}_O^L \leq \mathbf{x}_O \leq \mathbf{x}_O^H}} f_i(\mathbf{x}_O, \mathbf{x}_S, \mathbf{x}_D) \geq G_i, \quad i = 1, 2, ..., m. \tag{12}$$

A point in the space of the operating and statistical parameters represented by tuple $(\mathbf{x}_O, \mathbf{x}_S)$ where $f_i(\mathbf{x}_O, \mathbf{x}_S, \mathbf{x}_D)$ attains its minimal value is also referred to as a corner point. To simplify the notation, we use $f_i(c, \mathbf{x}_D)$ instead of $f_i(\mathbf{x}_O, \mathbf{x}_S, \mathbf{x}_D)$ where c is a corner point $(\mathbf{x}_O, \mathbf{x}_S)$.

Corners $c = (\mathbf{x}_O, \mathbf{x}_S)$ and $c' = (\mathbf{x}'_O, \mathbf{x}'_S)$ are considered to be similar ($c \approx c'$) if the following requirements are satisfied:

$$|\|\mathbf{x}_S\| - \|\mathbf{x}'_S\|| \leq 0.01, \tag{13}$$

$$\angle(\mathbf{x}_S, \mathbf{x}'_S) \leq 10^\circ, \tag{14}$$

$$\frac{|\mathbf{x}_{O,i} - \mathbf{x}'_{O,i}|}{\mathbf{x}_{O,i}^H - \mathbf{x}_{O,i}^L} \leq 0.01. \tag{15}$$

Algorithm 1: Design for yield.

$\mathcal{C}_i^0 \leftarrow \emptyset$ for $i = 1, ..., m$;
$\mathbf{x}_D^0$ is the initial point;
$k \leftarrow 1$;
while *True* **do**
/* Update the sets of corner points */
newcorner $\leftarrow$ False;
for $i \leftarrow 1$ **to** m **do**
$c \leftarrow \arg\min_{\substack{\|\mathbf{x}_S\| \leq \beta_0 \\ \mathbf{x}_O^L \leq \mathbf{x}_O \leq \mathbf{x}_O^H}} f_i\left(\mathbf{x}_O, \mathbf{x}_S, \mathbf{x}_D^{k-1}\right)$;
if $f_i\left(c, \mathbf{x}_D^{k-1}\right) < G_i$ *or* $\mathcal{C}_i^{k-1} = \emptyset$ **then**
if $\exists c' \in \mathcal{C}_i^{k-1} : c' \approx c$ **then**
$\mathcal{C}_i^k \leftarrow \left(\mathcal{C}_i^{k-1} \setminus \{c'\}\right) \cup \{c\}$;
else
$\mathcal{C}_i^k \leftarrow \mathcal{C}_i^{k-1} \cup \{c\}$;
newcorner $\leftarrow$ True;
end
end
end
if *newcorner is False* **then**
Exit with success, return $\mathbf{x}_D^{k-1}$;
end
/* Circuit sizing problem */
Starting from $\mathbf{x}_D^{k-1}$ find $\mathbf{x}_D^k$ such that
$f_i\left(c, \mathbf{x}_D^k\right) \geq G_i \ \forall c \in \mathcal{C}_i^k$ and $i = 1, ..., m$;
/* Circuit sizing problem ends here */
if *no such* $\mathbf{x}_D^k$ *found* **then**
Exit with failure;
end
$k \leftarrow k + 1$;
end

A circuit designer searches for $\mathbf{x}_D$ such that (12) is satisfied (i.e. all the worst-case performances satisfy the corresponding design requirements). This is not a trivial task because the worst-case performance depends on $\mathbf{x}_D$. Such a design problem can be solved by iteratively introducing the corner points and solving a sequence of circuit-sizing problems. A set of the corner points ($\mathcal{C}_i$) is associated with every circuit performance. Algorithm 1 searches for the design parameters for which (8) (i.e. (12)) is satisfied. The initial value of the design parameters is denoted by $\mathbf{x}_D^0$. The circuit-sizing problem is solved by formulating a weighted penalty function [4] for the m design requirements over the corresponding corners in sets $\mathcal{C}_i$. The penalty function is then minimized to obtain $\mathbf{x}_D^k$ which is used as the initial point for the circuit-sizing problem in the next iteration of Algorithm 1.

Algorithm 2: Size a circuit over corners

Let $\mathcal{C}_i$ denote the set of corners for design requirement $f_i \geq G_i$;
Set $\mathcal{B}_i^0 = \emptyset$ for $i = 1, ..., m$;
$\mathbf{x}_D^0$ is the initial point;
$k \leftarrow 1$;
while *True* **do**
  added ← False;
  for $i \leftarrow 1$ **to** m **do**
    $c \leftarrow \arg\min_{c \in \mathcal{C}_i} f_i\left(c, \mathbf{x}_D^{k-1}\right)$;
    if $c \in \mathcal{B}_i^{k-1}$ **then**
      Exit with failure;
    else
      $\mathcal{B}_i^k \leftarrow \mathcal{B}_i^{k-1} \cup \{c\}$;
      added ← True;
    end
  end
  if *added is False* **then**
    Exit with success, return $\mathbf{x}_D^{k-1}$;
  end
  Starting from $\mathbf{x}_D^{k-1}$ find $\mathbf{x}_D^k$ such that
   $f_i\left(c, \mathbf{x}_D^k\right) \geq G_i \ \ \forall c \in \mathcal{B}_i$ and $i = 1, ..., m$;
  if *no such* $\mathbf{x}_D^k$ *found* **then**
    Exit with failure;
  end
  $k \leftarrow k + 1$;
end

A circuit performance in one corner is evaluated from the results of one or more simulations. Solving the circuit sizing problem requires many evaluations of all circuit performances (f_i) over the corresponding corners in $\mathcal{C}_i$. As the number of the corners grows with the number of the outer loop iterations of Algorithm 1, it makes sense to identify the corner $c_i^W \in \mathcal{C}_i$ with the lowest (worst) value of f_i at $\mathbf{x}_D^k$. It is sufficient to size the circuit with the corner sets reduced to just this one element (i.e. $\mathcal{C}_i = \left\{c_i^W\right\}$). Algorithm 2 identifies this corner iteratively. A corner set used in the k-th iteration of Algorithm 2 (denoted by $\mathcal{B}_i^k$) is a superset of the set used in iteration $k-1$. This strategy makes the outer loop finite and generally reduces the number of the required simulations. In most cases, two iterations are sufficient for solving the circuit-sizing problem corresponding to one iteration of the outer loop of Algorithm 1.

4 MESH ADAPTIVE DIRECT SEARCH

MADS [6] is a family of derivative-free optimization algorithms [7] that rely on a dense set of normalized poll directions for guaranteeing convergence properties on nonsmooth and constrained optimization problems of the form

$$\min_{\mathbf{x} \in \Omega \subseteq \mathbb{R}^n} f(\mathbf{x}) \tag{16}$$

The points examined in the k-th iteration lie on mesh $\mathcal{M}_k$ which has a finite countable intersection with any bounded subset of the search space. The mesh density is controlled by the mesh size parameter (Δ_k^m) which approaches zero in the limit (i.e. the mesh becomes infinitely dense). Note that the superscript m is a standard notation used in papers on MADS and has nothing to do with the number of the circuit performances. The length of the steps taken by MADS is controlled by the step size parameter Δ_k^p. The step size parameter approaches zero at a slower rate compared to the mesh size parameter ($\Delta_k^m / \Delta_k^p \to 0$).

MADS can handle constraints with the extreme barrier approach where f is replaced with f_Ω which is equal to f if $\mathbf{x} \in \Omega$ and $+\infty$ otherwise. The incumbent solution and the corresponding value of f_Ω in iteration k are denoted by $\mathbf{x}_k$ and f_k.

Algorithm 3: Iteration k of a MADS algorithm using the extreme barrier approach.

/* Search step */
Evaluate f_Ω on a finite subset $\mathcal{S}_k \subset \Omega$;
/* Poll step */
Evaluate $f_\Omega\left(\mathbf{x}_k + \mathbf{d}\right)$ for all $\mathbf{d} \in \mathcal{D}$;
/* Update incumbent solution, Δ_k^m, and Δ_k^p */
Let $\mathbf{x}'$ be the point with the lowest value of f_Ω evaluated in this iteration;
if $f_\Omega\left(\mathbf{x}'\right) < f_k$ **then**
  $\mathbf{x}_{k+1} \leftarrow \mathbf{x}'$;
  Choose $\Delta_{k+1}^p \geq \Delta_k^p$ and $\Delta_{k+1}^m \geq \Delta_k^m$;
else
  $\mathbf{x}_{k+1} \leftarrow \mathbf{x}_k$;
  Choose $\Delta_{k+1}^p < \Delta_k^p$ and $\Delta_{k+1}^m < \Delta_k^m$;
end

The algorithm outline is given by Algorithm 3. Set $\mathcal{D}$ is a set of the scaled poll steps in iteration k. Note that its members are chosen in such manner that $\mathbf{x}_k + \mathbf{d} \in \mathcal{M}_k$. The convergence properties of MADS are guaranteed by the poll step if certain requirements are satisfied [6]. The search step does not affect the convergence properties, but it can significantly speed up the algorithm. In our implementation, the members of $\mathcal{S}_k$ are chosen using a quadratic model of the objective and constraint functions. An overview of the convergence properties and implementation details can be found in [8]. MADS is used for solving the optimization problem in the inner loop of Algorithm 1 [9] and for solving the circuit-sizing problem. It can also be used for computing the worst-case distance (by solving problem (5) and inserting the obtained result into (6)).

5 EXAMPLES AND RESULTS

The proposed algorithm is implemented in Python as part of the PyOPUS library [10]. SPICE OPUS [11] is used as the circuit simulator. The algorithm is parallelized where possible (i.e. evaluation of the m worst-case points in Algorithm 1 and evaluations of a circuit over multiple corners in Algorithm 2). All experiments are performed on a 3.2GHz Intel XEON processor with four cores and eight threads. One thread is reserved for the manager task and the remaining seven threads serve as workers.

The proposed approach is tested by sizing two operational transconductance amplifiers (OTA) [12]. Both circuits are sized for a $0.18\mu\text{m}$ manufacturing process. The Pelgrom model of the device mismatch [1] is used. Global variations are not taken into account.

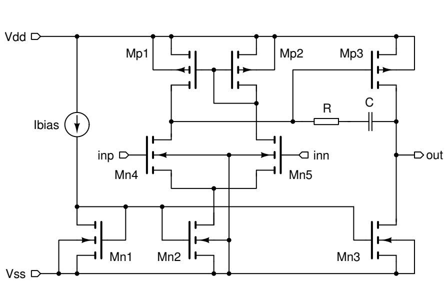

Figure 2. Miller OTA.

The first circuit (Figure 2) is a simple Miller OTA. The circuit has 13 design parameters (11 transistor channel widths and lengths, one resistance, and one capacitance), 16 statistical parameters (two parameters for every transistor), and two operating parameters (temperature and supply voltage). The bias current (Ibias) is set to $100\mu\text{A}$.

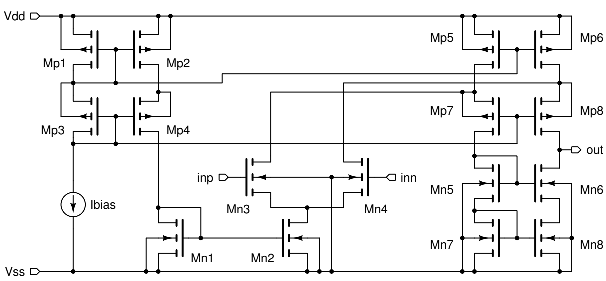

Figure 3. Folded cascode OTA.

The second circuit (Figure 3) is a folded cascode OTA (FCOTA) with 11 design parameters, 32 statistical parameters, and two operating parameters. The bias current is set to $5\mu\text{A}$.

Table 1. Operating parameters.

	Low	High	Nominal
Tempetarure [°C]	0	100	25
Supply voltage [V]	1.7	2.0	1.8

The ranges and the nominal values of the operating parameters are given in Table 1. All transistors are required to operate in the staturation region at the operating point. This results in two additional requirements for every transistor (i.e. $V_{GS} \geq V_T$ and $V_{DS} \geq V_{GS} - V_T$). These requirements are enforced in all corners, but they are not subject to the worst-case analysis or yield optimization.

Every circuit is first sized at the nominal operating and nominal statistical parameter values (i.e. $\mathbf{x}_S = \mathbf{0}$). The resulting design parameters are used as a starting point for Algorithm 1. The target yield is set to 99.87% ($\beta_0 = 3$).

Table 2. Results for the Miller OTA optimization.

	Goal	Final	WCD
Area [μm^2]	≤ 9000	2117	-
Supply current [μA]	≤ 1000	175	> 8
Swing [V]	≥ 1.0	1.21	> 8
Gain [dB]	≥ 60	77.8	> 8
UGBW [MHz]	≥ 10	11.0	6.1
Phase margin [°]	≥ 50	56.6	> 8
CMRR [dB]	≥ 90	90.1	3.3
PSRR V_{DD} [dB]	≥ 60	84.9	> 8
PSRR V_{SS} [dB]	≥ 60	84.5	> 8
Overshoot $\downarrow$ [%]	≤ 10	8.4	> 8
Overshoot $\uparrow$ [%]	≤ 10	9.5	> 8
$T_{set} \downarrow$ [μs]	≤ 1	0.784	> 8
$T_{set} \uparrow$ [μs]	≤ 1	0.831	> 8
Slew rate $\downarrow$ [V/μs]	≥ 2	2.02	4.4
Slew rate $\uparrow$ [V/μs]	≥ 2	2.01	3.7

The results of the Miller OTA optimization are listed in Table 2. The final optimization result is verified by computing the worst-case distances of the circuit performances. All the worst-case distances are greater than β_0 implying that the target yield is exceeded. The process of circuit sizing takes 57 minutes.

Table 3. Results for the FCOTA optimization.

	Goal	Final	WCD
Area [μm^2]	≤ 2000	1637	-
Supply current [μA]	≤ 400	63.8	> 8
Offset (low) [mV]	≥ -20	-8.39	7.1
Offset (high) [mV]	≤ 20	8.56	7.1
Swing [V]	≥ 0.3	4.09	> 8
Gain [dB]	≥ 70	70.2	4.2
UGBW [MHz]	≥ 4	4.04	3.9
Phase margin [°]	≥ 60	61.4	6.2
CMRR [dB]	≥ 100	126	> 8
PSRR V_{DD} [dB]	≥ 70	140	> 8
PSRR V_{SS} [dB]	≥ 70	71.0	> 8
Overshoot $\downarrow$ [%]	≤ 10	4.8	> 8
Overshoot $\uparrow$ [%]	≤ 10	0.34	> 8
$T_{set} \downarrow$ [μs]	≤ 2.5	0.189	> 8
$T_{set} \uparrow$ [μs]	≤ 2.5	0.319	> 8
Slew rate $\downarrow$ [V/μs]	≥ 2	6.07	> 8
Slew rate $\uparrow$ [V/μs]	≥ 2	4.08	> 8

The results of the FCOTA optimization are listed in Table 3. The process of circuit sizing takes 9.5 minutes.

The final result is verified with the worst-case distance computation which confirms that the final yield is greater than the target yield.

6 Conclusion

Designing circuits that exhibit a high parametric yield is an important task in the process of the analog integrated circuit design. Computing the yield as well as finding the design parameters that satisfy the minimum yield requirement is an optimization problem. As circuit simulators do not compute sensitivities, the derivatives of the function subject to optimization are not available. Therefore, it makes sense to use derivative-free methods for solving such optimization problems.

An automated design methodology is proposed based on a set of corners that represent the operating conditions and manufacturing process variations where the circuit exhibits the worst performance. The corners are computed by solving an optimization problem. The optimal design parameter values are found by solving an optimization problem obtained from the circuit-sizing problem by using a penalty function-based approach. The set of corners is built iteratively by repeatedly solving the two optimization problems. The process stops when all the design requirements are satisfied and there are no new corners found.

For each optimization a mesh adaptive direct search optimization algorithm is used. The proposed approach is tested on two analog circuit-sizing problems. The results show that the approach is effective and capable of finding a circuit satisfying the minimum yield requirement.

7 Acknowledgement

The research was co-funded by the Ministry of Education, Science, and Sport (Ministrstvo za Šolstvo, Znanost in Šport) of the Republic of Slovenia through the programme P2-0246 Algorithms and optimization methods in telecommunications.

References

[1] M.J.M. Pelgrom, A.C.J. Duinmaijer, A.P.G. Welbers, Matching properties of MOS transistors, *IEEE J. Solid-State Circuits* Vol. 24, pp. 1433–1440, 1989.

[2] C. Michael, M. Ismail, Statistical modeling of device mismatch for analog MOS integrated circuits, *IEEE Journal of Solid-State Circuits*, Vol. 12, pp. 154–166, 1992.

[3] J. Bastos, M. Steyaert, A. Pergoot, W. Sansen, Mismatch characterization of submicron MOS transistors, *Analog Integrated Circuits and Signal Processing*, Vol. 12, pp. 95–106, 1997.

[4] Á. Bűrmen, D. Strle, F. Bratkovič, J. Puhan, I. Fajfar, T. Tuma, Automated robust design and optimization of integrated circuits by means of penalty functions, *AEU-International Journal of Electronics and Communications*, Vol. 57, pp. 47–56, 2003.

[5] H. Graeb, Analog Design Centering and S izing, Springer, 2007.

[6] C. Audet, J.E. Dennis Jr., Mesh adaptive direct search algorithms for constrained optimization, *SIAM Journal on Optimization*, Vol. 17, pp. 188–217, 2006.

[7] A.R. Conn, K. Scheinberg, L.N. Vincente, *Introduction to Derivative-Free Optimization*, SIAM, Philadelphia (PA), 2009.

[8] Á. Bűrmen, J. Olenšek, T. Tuma, Mesh adaptive direct search with second directional derivative-based Hessian update, *Computational Optimization and Applications*, DOI 10.1007/s10589-015-9753-5, April 2015.

[9] Á. Bűrmen, H. Habal, Computing Worst-case performance and yield of analog integrated circuits by means of mesh adaptive direct search, Informacije MIDEM - Journal of Microelectronics, Electronic Components and Materials, Vol. 45, scheduled for publication in issue 2, 2015.

[10] Á. Bűrmen Optimizacija vezij s knjižnico PyOPUS, *Elektrotehniški vestnik*, Vol. 79, pp. 149–152, 2012.

[11] Á. Bűrmen, T. Tuma, Circuit Simulation with SPICE OPUS, Theory and Practice, Birkhäuser, 2009.

[12] R.J. Baker, CMOS Circuit Design, Layout, and Simulation John Wiley & Sons, 2008.

Árpád Bűrmen received his Ph.D. degree in electrical engineering from the Faculty of Electrical Engineering, University of Ljubljana, Slovenia, in 2003. Currently, he is an associate professor at the same faculty. His research interests include continuous and event-driven simulation of circuits and systems, optimization methods and their convergence theory and applications, and algorithms for parallel and distributed computation. He is one of the principal developers of the SPICE OPUS circuit simulator and the PyOPUS design automation library.