Optimization Letters (2019) 13:1557–1571
https://doi.org/10.1007/s11590-018-1364-5

ORIGINAL PAPER

Derivative free methodologies for circuit worst case analysis

Vittorio Latorre[1] · Husni Habal[2] · Helmut Graeb[3] · Stefano Lucidi[4]

Received: 9 June 2017 / Accepted: 19 November 2018 / Published online: 12 December 2018

Abstract

In this paper, a new derivative-free method for Worst Case Analysis (WCA) of circuit design is defined. A WCA of a device can be performed by solving a particular minimization problem where the objective function values are obtained by a simulation code and where some variables are subject to a spherical constraint and others to box constraints. In order to efficiently tackle such a problem, the paper defines a new DF algorithm which follows a two blocks Gauss Seidel approach, namely it alternates an approximated minimization with respect to the variables subject to the spherical constraint with an approximated minimization respect to the variables subject to the box constraints. The algorithm is described and its global convergence properties are analyzed. Furthermore it is tested in the WCA of a MOSFET operational amplifier and its computational behaviour is compared with the one of the efficient optimization tool of the WiCkeD suite for circuit analysis. The obtained results seem to indicate that the proposed algorithm is promising in terms of average efficiency, accuracy and robustness.

✉ Vittorio Latorre
v.latorre@federation.edu.au

Husni Habal
Husni.Habal@infineon.com

Helmut Graeb
graeb@tum.de

Stefano Lucidi
lucidi@dis.uniroma1.it

[1] Faculty of Science and Technology, Federation University Australia, Mt Helen, VIC 3353, Australia

[2] Infineon Technologies AG, Neubiberg, Germany

[3] Institute for Electronic Design Automation, Department of Electrical Engineering and Information Technology, Technische Universität München, München, Germany

[4] Department of Computer, Control and Management Engineering, University of Rome Sapienza, Via Ariosto 25, Rome, Italy

Keywords Derivative free optimization · Bilevel optimization · Circuit design · Yield optimization

1 Introduction

Commodity integrated circuit (IC) chips are commonly fabricated with complementary metal-oxide-semiconductor field-effect transistor (MOSFET) devices. This is due to the low power draw, small dimensions, and cheap manufacturing cost of MOSFET IC chips. Analog circuits, such as operational amplifiers, analog-digital converters, and radio frequency transducers are often implemented as components in an IC chip [18].

MOSFET devices exhibit complex and multifaceted behaviors and this implies that accurate analyses of their behaviors very often require complex numerical circuit simulators. These simulators usually need to calculate the circuit response to electrical stimulus and, hence, they can require very high computational costs.

Despite this difficulty, the numerical circuit simulators represent important tools for the designs of analog circuits (see, i.e., [3,8,9,13,14,16,24,26]) and mature commercial tools are also available [22]. Furthermore they are also extremely useful in helping designers to study the robustness of the circuits produced, namely to estimate the worst differences between "optimal theoretical" behaviours of the circuit and the "real" ones. These differences can be due to (for example):

- imperfections of sub-wavelength lithography, random dopant fluctuations, and line edge roughness during the integrated circuit fabrication procedure;
- different electrical and environmental operating conditions such as changes in supply voltage from the nominal value, variations in circuit temperature.

There is an interest in circuit designs which are as robust as possible with respect to the effects of process and environmental variations on the circuit performance metrics. However it is necessary to have techniques to accurately estimate the behavior of a circuit according to all the possible realizations of such variations.

An important technique to perform a robust circuit optimization is the so called "worst case analysis" (WCA) [15]. WCA considers a particular performance of the circuit and determines the worst value that such performance can assume within all the manufacturing and environmental fluctuations. A robust design could be the one that corresponds to the "best" worst case analysis.

More formally, the worst case analysis consists in minimizing a objective function, whose values are computed by a circuit simulation code, over a feasible set given by all the manufacturing and environmental fluctuations. Therefore no first order information of the objective function is available.

The algorithms previously proposed in literature (see [3,15]) overcome this difficulty by approximating the first order information of the objective function with finite difference techniques and then by using gradient-based optimization methods to solve the given optimization problem. This approach generally performs well, however it is well known [19] that sometimes the use of first-order approximation can be inefficient due to the presence of noise (such as the one caused by a simulator in calculating the value of a objective function).

In order to offer a possible alternative to the proposed gradient based approach, we define a new derivative-free method specifically designed for solving the optimization problem deriving from the WCA.

This method belongs the class of derivative free algorithms which try to overcome the lack of gradient information by efficiently sampling of the objective function over the feasible region (see, f. e., [1,10,17,20]). Its distinguishing feature is to perform a sampling of the objective function which exploits as much as possible the particular structure of the feasible set of the considered optimization problem. In fact the feasible set is the Cartesian product of two particular closed convex sets. The proposed algorithm takes into account this feature by following a two block Gauss Seidel approach which combines the derivative free algorithms described in [19,21].

We present the algorithm and its convergence proprieties towards a stationary point of the objective function. Then we report an analysis of its numerical performance on a MOSFET operational amplifier. This analysis is completed with the comparison of the proposed algorithm with WiCkeD [22], a suite for circuit analysis and optimization. WiCkeD implements an efficient deterministic yield WCA where the gradient is calculated throughout the use of finite differences.

The paper is organized as follows. In the next section we introduce the parameters involved in the problem and explain WCA in detail. In Sect. 3 we report the algorithm with its convergence proprieties. Finally in Sect. 4 we present the Miller operational amplifier together with numerical results.

2 Circuit worst case analysis problem

WCA considers a particular performance which can characterize the circuit behavior (for example, gain or slew-rate described in the results section). Each performance is an expression of the circuit node voltages and terminal currents, and can be calculated by numerical circuit simulations.

For every possible choice of the apparatus of the circuit, every performance depends on two classes of parameters. The first class consists of the process parameters, which specify the apparatus as well, but their values can have fluctuations due to the manufacturing process. The second class consists in the operational parameters which represent the environmental conditions in which the circuit works and, hence, their values surely have fluctuations.

In correspondence of a particular design, the aim of the worst case analysis is to predict the maximum deterioration of the performance due to the variations of the process and operational parameters. Formally, the worst case analysis tries to find the worst value of a objective function with the following structure:

$$f(x, y),$$

where $x \in \mathbb{R}^n$ is the vector of process parameters, $y \in \mathbb{R}^m$ is the vector of operational parameters and $f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ represents the value of the chosen performance for a given choice of (x, y), We recall that the values of f are usually given by a complex simulation process (see, for example, [7]).

As regards x, it is a real valued vector which represents the random variations of process parameters between different manufactured circuits. Therefore x can be represented by a vector of random variables associated with an arbitrary probability density function, that without loss of generality, can be transformed into a normal standard distribution of random variables [11]. Therefore we can write

$$x \in \mathbb{R}^n, \quad x \sim pdf_N\,(\mathbf{0}, \mathbf{I})\,. \tag{1}$$

Consequently, the mean of x is the zero vector, $\mathbf{0}$, and the covariance matrix is the identity matrix, $\mathbf{I}$, of dimension $n \times n$.

As regards y it is a real valued vector which represents the random variations of operational parameters, namely represents the different environmental conditions in which the circuit can work after production. Therefore y can be represented by a vector of random variables uniformly distributed in a box constrained region, namely:

$$\left\{y \in \mathbb{R}^m : l \leq x \leq u, l \in \mathbb{R}^m, u \in \mathbb{R}^m\right\}.$$

However, WCA can be performed by avoiding the difficulties associated with the handling of random variables. Assuming, without loss of generality, that the designer is interested in a lower bound on the values of the given performance, WCA can be obtained by solving the following optimization problem:

$$\begin{array}{ll} \min\limits_{x,y} & f(x,y) \\ & \|x\|^2 \leq r^2\,\cdot \\ & l \leq y \leq u \end{array} \tag{2}$$

The first constraint relates to the need to find the lowest value of the performance with respect all the process fluctuations that are under a certain threshold r. This threshold is chosen so to guarantee that there is a (high) probability that the realizations of the random variable x fall into such region. Since the process parameters x are distributed according to a normal standard distribution, there is a one to one relation between the values of the threshold r and the probability that the realizations of the random variable x fall into the spherical constraint. For example $r = 1$ gives a region $\{x \in R^n, \|x\|^2 \leq r^2\}$ which corresponds to a probability of the 84% of the total possible realizations of the circuit, while $r = 3$ it corresponds to the 99.9% of the total realizations. Therefore, if $r = 1$, this constraint implies that the minimization of the performance is performed with respect of 84% of the possible variations of process parameters, namely with respect of 84% of circuit realizations [15].

The second constraint takes into account that the variations of the operational parameters are uniformly distributed in a defined box. In fact this constraint ensures that the minimization of the performance is performed with respect all the possible values of the random variable y.

3 Optimization procedure

As described in the preceding sections, the worst case analysis requires the solution of a minimization problem with the following structure:

$$\begin{aligned} \min_{x,y} \quad & f(x,y) \\ & x \in S \\ & y \in X \end{aligned} \tag{3}$$

Where $x \in \mathbb{R}^n$, $y \in \mathbb{R}^m$, $f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ is a C^1 function and

$$S := \left\{ x \in \mathbb{R}^n, \|x\|^2 \leq r^2 \right\}, \quad Y := \left\{ y \in \mathbb{R}^m, l \leq y \leq u \right\}, \tag{4}$$

with $l \in \mathbb{R}^m$, $u \in \mathbb{R}^m$ and r is a positive scalar.

The feasible set of Problem (3) can be rewritten as: $\mathcal{F} := S \times Y$.

In connection with the preceding problem it is possible to introduce the following definition of a stationary point.

Definition 1 A feasible point $(\bar{x}, \bar{y}) \in \mathcal{F}$ is a stationary point of Problem (3) if

$$\nabla_x f(\bar{x}, \bar{y})^T d_x + \nabla_y f(\bar{x}, \bar{y})^T d_y \geq 0 \quad \forall d_x \in T(\bar{x}), \quad \forall d_y \in D(\bar{y}).$$

where for all $x, y \in \mathcal{F}$:

$$T(x) := \begin{cases} d \in \mathbb{R}^n & \text{if} \quad \|x\|^2 < r^2 \\ d \in \mathbb{R}^n : x^T d \leq 0 & \text{if} \quad \|x\|^2 = r^2 \end{cases}$$

$$D(y) := \left\{ d \in \mathbb{R}^m : d_i \geq 0 \text{ if } y_i = l_i, d_i \leq 0 \text{ if } y_i = u_i \right\}.$$

Then it is possible to recall a well known optimality condition (see [5] or Chapter 2 of [6] for the proof).

Theorem 1 *If a feasible point $(x^*, y^*) \in \mathcal{F}$ is a local minimum point for (3) then it is also a stationary point.*

As described in the previous sections, one distinguishing feature of the class of WCA optimization problems (3) is the fact that the values of the objective function are obtained by a simulation procedure. This precludes the possibility to compute the analytical expression of the derivatives. Furthermore, the simulators sometimes introduce noise in the calculations of the objective function values, such that the first order information on the function cannot be efficiently approximated.

These difficulties indicate that a suitable approach to tackle this class of minimization problems can be the use derivative-free methodologies, namely optimization algorithms that exploit only the evaluations of the objective function (see for example [1,4,10,12,17,25]).

For many complex circuits, the simulation of performances values can be expensive in terms of CPU time. Therefore, for these real-world problems, the algorithm should make significant progress toward the solution in a few evaluations of the objective function. In order to meet this need, the class of linesearch-based derivative-free algorithms has been considered. The common idea of this class of algorithms is to use a derivative-free line search technique along some suitable sets of directions. The sets of directions must convey sufficient information on the local behavior of the objective function. The algorithm chooses a direction and checks if there is a sufficient decrease of the objective function along such direction. In this case the algorithm performs an extrapolation along the considered direction in order to exploit as much as possible the decrease of the objective function values. This extrapolation strategy usually strengthens the ability to produce points where the function has the significant decreases especially in the early iterations. Namely, when the points considered by the algorithm are far from being local minima.

Another important distinguishing features of the problem (3) is the structure of its feasible set which is the Cartesian product of two particular closed convex sets: the operating variables y are constrained by simple bounds constraints, while a spherical constraint is imposed on the process variables x. The algorithm proposed in this paper tries to exploit as much as possible this structure by adopting a two block Gauss Seidel approach. It alternates a approximated minimization with respect to the process variables x and an approximated minimization with respect to the operating variables y. In particular, the minimization with respect to x is performed by following the approach proposed in [19] where a derivative-free algorithm for the bounds constrained problem is described. Instead, the minimization with respect to y is performed by using the derivative-free model described [21] for minimization problems subject to constraints belonging to a particular class which includes, as a special case, a spherical constraint.

The following Algorithm describes the main steps of the proposed strategy.

Algorithm DFA- Derivative-Free Algorithm for Problem (3)

```
Given $(x_0, y_0) \in \mathcal{F}$, $\tilde{\alpha}_0 \in \mathbb{R}^n_+$ and $\bar{\alpha}_0 \in \mathbb{R}^m_+$.
Set $k = 0$
While $(x_k, y_k)$ is not a stationary point of Problem (3) do
    Compute $\tilde{x}_{k+1}$ and $\tilde{\alpha}_{k+1}$ by Min$_x(x_k, y_k, \tilde{\alpha}_k)$
    Compute $\tilde{y}_{k+1}$ and $\bar{\alpha}_{k+1}$ by Min$_y(\tilde{x}_{k+1}, y_k, \bar{\alpha}_k)$
    Find $(x_{k+1}, y_{k+1})$ such that $f(x_{k+1}, y_{k+1}) \leq f(\bar{x}_{k+1}, \bar{y}_{k+1})$ and $(x_{k+1}, y_{k+1}) \in \mathcal{F}$,
    Set $k = k + 1$
End while
```

Roughly speaking, given an initial point $(x_0, y_0) \in \mathcal{F}$ the main idea behind the algorithm is that, for every iteration $k = 0, 1, \ldots$:

- an approximated minimizer $\bar{x}_{k+1}$ of $f(\cdot, y_k)$ is computed by Procedure Min$_x$;
- an approximated minimizer $\bar{y}_{k+1}$ of $f(\bar{x}_{k+1}, \cdot)$ is computed by Procedure Min$_y$;
- a point (x_{k+1}, y_{k+1}) such that: $f(x_{k+1}, y_{k+1}) \leq f(\bar{x}_{k+1}, \bar{y}_{k+1})$ can be selected, this guarantees that every approximation technique (able to improve the point $(\bar{x}_{k+1}, \bar{y}_{k+1})$) can be included in the framework without precluding the global convergence properties of the algorithm (if no additional procedure is used then $(x_{k+1}, y_{k+1}) = (\bar{x}_{k+1}, \bar{y}_{k+1})$).

Both the Procedures Min_x and Min_y perform their approximate minimization by sampling the value of the objective function along two suitable sets of search directions which try to exploit the particular structure of the sets D and Y.

In particular, at every iteration k, the Procedure Min_x takes into account the spherical constraint on x by setting as search directions d_x^i, $i = 1, \dots, n$:

- the coordinate directions if the point x_k is sufficiently interior to the feasible set;
- a set of n directions composed by an orthogonal direction to the sphere in the considered point and $n-1$ directions that are an orthonormal base of the hyperplane tangent to the sphere in that point if the point x_k is close to the boundary of the feasible set (see [21] for details).

Every direction d_x^i, $i = 1, \dots, n$ is analyzed to check if an improvement of the objective function along such direction can be obtained. If the current point z_k^i is in the interior (lines 9 and 10) no change is applied on the direction. If such point is close to the boundary, direction d_x^i is projected on the cone of first order feasible variations in z_k^i with respect to the spherical constrain:

$$\hat{d}^i = P_{\left\{d:(z_k^i)^T d \le 0\right\}}\left[d_x^i\right] = \left(I - \frac{z_k^i {z_k^i}^T}{\|z_k^i\|^2}\right) d^i.$$

Thanks to the spherical nature of the constraint the projection of the direction can be calculated in closed form as reported at line 12. Then the algorithm checks if a "sufficiently" improvement of the objective function can be found along the current direction in the initial stepsize $\tilde{\alpha}_k^i$. If the algorithm fails to improve the objective function along the direction, it tries the opposite direction. If no improvement is found along the opposite direction as well, the algorithm decreases the next initial stepsize and tries a new direction. In case when the initial stepsize produces a "sufficiently" improvement of the objective function along the considered direction or along its opposite direction, the algorithm finds an efficient stepsize α_k^i by using a derivative-free projected linesearch technique (Procedure LS_x) and produces the new point z_k^{i+1}. Finally it updates the next initial stepsize.

The approaches of Procedure Min_y and Procedure LS_y are quite similar to the ones of Procedure Min_x and Procedure LS_x but they exploit the simple structure of the box constraints which the new point y_{k+1} must satisfy. In particular, at every iteration, Procedure Min_y can use, as search directions, the coordinate axes and does not need any projection. Similarly Procedure LS_y can determine an efficient stepsize along a direction avoiding any projection by computing first the maximum possible stepsize such that the new point is feasible.

The following theorem reports the convergence property of Algorithm .

Theorem 2 *Let* $\{(x_k, y_k)\}$ *be a sequence generated by Algorithm then every accumulation point of* $\{(x_k, y_k)\}$ *is a stationary point of Problem* (3).

Proof First we have to prove that the constraint $\|x\|^2 - r^2 \le 0$ satisfies the Conditions B and C in [21]. It is easy to notice that it surely exists a $d \in \mathbb{R}^n$ satisfying:

$$2\tilde{x}^T d < 0 \quad \text{with} \quad \|\tilde{x}\|^2 = r^2.$$

Procedure $\mathbf{Min}_x(x_k, y_k, \tilde{\alpha}_k)$

```
Data $\gamma \in (0,1)$, $\theta \in (0,1)$ and $\epsilon > 0$.
If $\|x_k\|^2 - r^2 \le -\epsilon$ then
    Set $d_x^i = e_i,\ i = 1, \dots, n$
Else
    Compute the directions $d_x^1, \dots, d_x^n$ such that:
        $$d_x^1 = -\frac{x_k}{\|x_k\|}, \qquad d_x^2, \dots, d_x^n \text{ are an orthonormal base of } \{x \in \mathbb{R}^n : x_k^T x = 0\}$$
End if
Set $i = 1$, $z_k^i = x_k$,
For i=1,..., n
    If $\|z_k^i\|^2 - r^2 \le -\epsilon$ then
        $\hat{d}^i = d_x^i$
    Else
        $\hat{d}^i = \left(I - \frac{z_k^i {z_k^i}^T}{\|z_k^i\|^2}\right) d^i$
    End if
    If $f(P_{\mathcal{S}}[z_k^i + \tilde{\alpha}_k^i \hat{d}_x^i], y_k) \le f(z_k^i, y_k) - \gamma(\tilde{\alpha}_k^i)^2$ then
        Compute $\alpha_k^i$ by $\mathbf{LS}_x(\tilde{\alpha}_k^i, z_k^i, y_k, \hat{d}_x^i, \alpha_k^i)$, set $\tilde{\alpha}_{k+1}^i = \alpha_k^i$
    Else
        If $f(P_{\mathcal{S}}[z_k^i - \tilde{\alpha}_k^i \hat{d}_x^i], y_k) \le f(z_k^i, y_k) - \gamma(\tilde{\alpha}_k^i)^2$ then
            Compute $\alpha_k^i$ by $\mathbf{LS}_x(\tilde{\alpha}_k^i, z_k^i, y_k, -\hat{d}_x^i, \alpha_k^i)$, set $\tilde{\alpha}_{k+1}^i = \alpha_k^i$, $\hat{d}_x^i = -\hat{d}_x^i$
        Else
            Set $\alpha_k^i = 0$ and $\tilde{\alpha}_{k+1}^i = \theta \tilde{\alpha}_k^i$
        End if
    End if
    Set $z_k^{i+1} = P_{\mathcal{S}}[z_k^i + \alpha_k^i d_x^i]$
End for
Set $\tilde{x}_{k+1} = z_k^{n+1}$
Return $\tilde{x}_{k+1}$ and $\tilde{\alpha}_{k+1}$
```

Procedure $\mathbf{LS}_x(\tilde{\alpha}_k^i, z_k^i, y_k, d_x^i, \alpha_k^i)$

```
Data $\gamma \in (0,1)$ and $\delta \in (0,1)$.
Set $\beta = \tilde{\alpha}_k^i / \delta$ and $\alpha_k^i = \tilde{\alpha}_k^i$
While $f(P_{\mathcal{S}}[z_k^i + \beta d_x^i], y_k) \le f(z_k^i, y_k) - \gamma(\beta)^2$
    $\alpha_i^k = \beta$ and $\beta = \frac{\beta}{\delta}$
End while
Return $\alpha_k^i$
```

This implies that the constraint satisfies the MFCQ at every point where the spherical constraint is active. Recalling the results reported on page 43 of [21], Assumption B is satisfied.

Procedure Min$_y(x_{k+1}, y_k, \bar{\alpha}_k)$

```
Data $\gamma \in (0, 1)$ and $\theta \in (0, 1)$
Set set $w_k^1 = y_k$ and $d_y^i = \tilde{e}_i$, for $i = 1, \ldots, m$
For i=1,…,m
  Compute $\alpha_{max}$ such as $(w_k^i + \alpha_{max} d_y^i) \in \partial Y$ and set $\alpha = \min\{\bar{\alpha}_k^i, \alpha_{max}\}$.
  If $\alpha > 0$ and $f(\bar{x}_{k+1}, w_k^i + \alpha d_y^i) \leq f(\bar{x}_{k+1}, w_k^i) - \gamma\alpha^2$ then
    Compute $\alpha_k^i$ by LS$_y(\alpha, \alpha_{max}, \bar{x}_{k+1}, w_k^i, d_y^i, \alpha_k^i)$, set $\bar{\alpha}_{k+1}^i = \alpha_k^i$
  Else
    Compute $\alpha_{max}$ such as $(w_k^i - \alpha_{max} d_y^i) \in \partial Y$ and set $\alpha = \min\{\bar{\alpha}_k^i, \alpha_{max}\}$.
    If $\alpha > 0$ and $f(\bar{x}_{k+1}, w_k^i - \alpha d_y^i) \leq f(\bar{x}_{k+1}, w_k^i) - \gamma\alpha^2$ then
      Compute $\alpha_k^i$ by LS$_y(\alpha, \alpha_{max}, \bar{x}_{k+1}, w_k^i, -d_y^i, \alpha_k^i)$, set $\bar{\alpha}_{k+1}^i = \alpha_k^i$, $d_i = -d_i$
    Else
      Set $\alpha_k = 0$, $\bar{\alpha}_{k+1}^i = \theta\bar{\alpha}_k^i$
    End if
  End if
  Set $w_k^{i+1} = w_k^i + \alpha_k^i d_y^i$.
End for
Set $\tilde{y}_{k+1} = w_k^{m+1}$
Return $y_{k+1}$ and $\bar{\alpha}_{k+1}$
```

Procedure LS$_y(\alpha, \alpha_{max}, \bar{x}_{k+1}, w_k^i, d_y^i, \alpha_k^i)$

```
Data $\gamma > 0$, $\delta \in (0, 1)$
Set $\beta = \min\left\{\alpha_{max}, \frac{\alpha}{\delta}\right\}$ and $\alpha_k^i = \alpha$
While $f(x_k, w_k^i + \beta d_y^i) \leq f(x_k, w_k^i) - \gamma\beta^2$
  If $\beta = \alpha_{max}$ then
    Set $\alpha_i^k = \beta$ and Return $\alpha_k^i$
  End if
  Set $\alpha_i^k = \beta$ and $\beta = \min\left\{\alpha_{max}, \frac{\beta}{\delta}\right\}$
End while
Return $\alpha_k^i$
```

For Assumption C we have to prove that for every $\bar{x} \in \mathcal{F}$ there exist scalars $\xi > 0$ and $\eta > 0$ such that

$$\min_{z \in S} \|z - x\| \leq \eta[(\|x\|^2 - r^2)]_+ \quad \forall x \in \mathcal{B}_\xi(\bar{x}).$$

Such condition is trivially satisfied if $x \in S$. Otherwise we can easily prove that the solution of $\min_{z \in S} \|z - x\|$ is obtained when z is the projection of x on S, namely $z = r\frac{x}{\|x\|}$. Therefore we have:

$$\min_{z \in S} \|z - x\| = \|r\frac{x}{\|x\|} - x\| = \frac{1}{\|x\|}\|(r - \|x\|)x\|.$$

From the Cauchy–Swartz inequality:

$$\frac{1}{\|x\|}\|(r-\|x\|)x\| \leq \frac{1}{\|x\|}|(r-\|x\|)|\|x\| = |(r-\|x\|)|\frac{|r+\|x\||}{|r+\|x\||}$$
$$= \frac{\|x\|^2-r^2}{|r+\|x\||} \leq \frac{1}{r}\left(\|x\|^2-r^2\right),$$

if we choose $\eta < \frac{1}{r}$, then Assumption C holds.

From the instructions of Algorithm we have for all k:

$$f(\bar{x}_{k+1}, y_k) \leq f(x_k, y_k) - \gamma \sum_{i=1}^{n}\left(\alpha_k^i\right)^2,$$
$$f(\bar{x}_{k+1}, \bar{y}_{k+1}) \leq f(\bar{x}_{k+1}, y_k) - \gamma \sum_{i=1}^{m}\left(\alpha_k^i\right)^2,$$
$$f(x_{k+1}, y_{k+1}) \leq f(\bar{x}_{k+1}, \bar{y}_{k+1}).$$

Considering the compactness of the feasible set, the sequences $\{f(x_k, y_k)\}$, $\{f(\bar{x}_k, y_k)\}$, $\{f(\bar{x}_k, \bar{y}_k)\}$ all converge to the same limit. By using this propriety, all the theoretical proofs reported in [19,21] can be repeated with small modifications. Therefore if $(\hat{x}, \hat{y})$ is an accumulation point of the sequence $\{(x_k, y_k)\}$ the results reported in [21] imply that:

$$\nabla_x f(\hat{x}, \hat{y})^T d_x \geq 0 \quad \forall d_x \in T(\hat{x}), \tag{5}$$

while the results reported in [19] ensure that:

$$\nabla_y f(\hat{x}, \hat{y})^T d_y \geq 0 \quad \forall d_y \in D(\hat{y}). \tag{6}$$

Then the (5) and the (6) guarantee that $(\hat{x}, \hat{y})$ is a stationary point of $f(\cdot)$. □

4 Numerical results

The derivative-free algorithm described in this paper is applied to a practical problem from the domain of analog circuit design. A typical problem is the analysis of a MOSFET operational amplifier. The reader is referred to reference [2] for a thorough introduction to MOSFET circuit design—including operational amplifiers. A MOSFET operational amplifier is typically operated in a negative feedback configuration. It is used to define a closed-loop transfer function with high precision.

In this paper, the Miller Operational Amplifier (MOA) topology is used as shown in Fig. 1a. The MOA has differential (plus and minus) input terminals and a single-ended output terminal. It is composed of six p-type MOSFET devices, $P1$–$P5$, and three n-type devices, $N1$–$N3$. A coupling capacitor, C_c, creates an internal feedback path between circuit stages. The DC current source I_{bias} can be tuned to change the quiescent

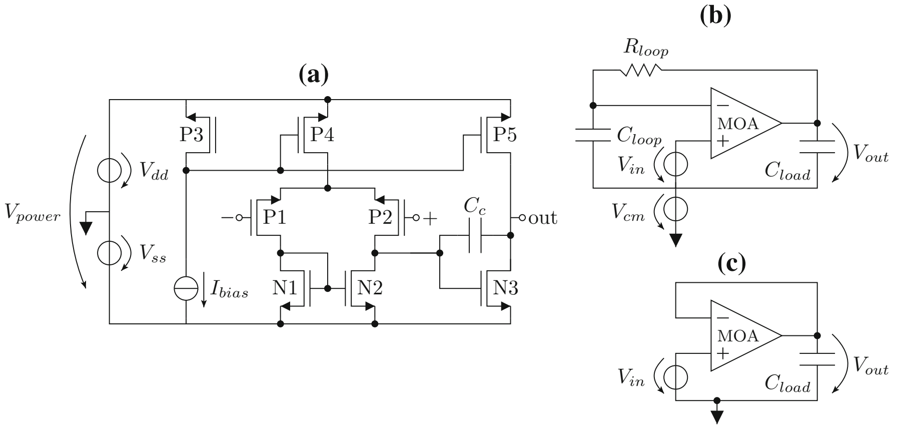

Fig. 1 **a** The Miller operational amplifier topology (MOA). **b** Testbench used in transfer function (XF) analysis. **c** Testbench used in transient and DC-sweep analysis

operating point of the circuit node voltages and branch currents. The DC voltage of sources V_{dd} and V_{ss} are the positive and negative voltage supplies respectively; their sum defines the voltage drop, V_{power}, between the positive and negative supply nodes: $V_{power} = V_{dd} + V_{ss}$.

Accurate numerical circuit simulation requires highly detailed mathematical MOSFET models. In particular, the results reported in this paper are based on the the industry-standard BSIM4 models [23] and on the commercial tool Spectre for numerical circuit simulation [7].

For the MOA, the operating parameters are:

- I_{bias} the value of the DC bias current shown in Fig. 1a. The nominal value is 4.0 μA. The range is $3.0 \leq I_{bias} \leq 5.0$ μA.
- V_{power} the value of the voltage drop between the positive and negative supply nodes. The nominal value is 2.2 V. The range is $2.0 \leq V_{power} \leq 2.4$ V.
- T circuit temperature during operation. The nominal value is 27.0 °C. The range is $0\,°\mathrm{C} \leq T \leq 100\,°\mathrm{C}$.

As regards the process parameters, a screening phase has pointed that only eight process parameters significantly affect the MOA performance features and that only these parameters need to be considered:

- $Vth_{N1}, Vth_{N2}, Vth_{P3}, Vth_{P4}$ the threshold voltages at zero substrate bias of devices $N1$, $N2$, $P3$, and $P4$, respectively.
- X_L the offset in channel length due to over or under etching, line edge roughness, and lithography imperfections.
- T_{ox} the gate oxide thickness.
- U_0 the low-field mobility.
- N_{dep} the substrate doping concentration.

The threshold voltages are local parameters particular to an individual device. Parameters X_L, T_{ox}, U_0 N_{dep} are global parameters; their values are shared amongst all devices. The process parameters have been transformed to have standard normal distributions with zero correlation in the device models.

Eleven basic performance features of an operational amplifier are listed below. They will be the objectives of worst-case analysis. The first six performances depend on the circuit frequency response. They can be calculated using the testbench shown in Fig. 1b and a transfer function (XF) analysis in Spectre. Components R_{loop} and C_{loop} are set to large values, so that R_{loop} acts as a high impedance path, and C_{loop} acts as a low impedance path for frequencies $f \geq 100$ Hz.

1. Low-frequency voltage gain (A_{lf}): defined as the magnitude of the frequency domain transfer function from the input $V_{in}(f)$ to the output $V_{out}(f)$ for $f = 100$ Hz;
2. Unity gain bandwidth (f_{UGBW}): the frequency such that $A(f_{UGBW}) = 1$.
3. Phase margin (PM): it is the difference between phase at $f = f_{UGBW}$ and $-180°$;
4. Rejection of low-frequency signals injected at the positive supply node (PSRR-V_{dd}): it is defined as the ratio of two transfer functions. The numerator is A_{lf}. The denominator is the magnitude of the frequency domain transfer function from the positive voltage supply, $V_{dd}(f)$, to the output, $V_{out}(f)$;

$$H_{dd}(f) = \left| \frac{V_{out}(f)}{V_{dd}(f)} \right|, \quad f = 100 \text{ Hz} \Longrightarrow \text{PSRR-}V_{dd} = \frac{A_{lf}}{H_{dd}(100)}.$$

5. Rejection of low-frequency signals injected at the negative supply node (PSRR-V_{ss}): it is defined similarly to PSRR-V_{dd}, but starting with the negative supply, $V_{ss}(f)$.
6. Rejection of low-frequency signals common to both the plus and minus input terminals (CMRR): it is the ratio of two transfer functions. The numerator is A_{lf}. The denominator is the magnitude of the frequency domain transfer function from the common-mode supply, $V_{cm}(f)$, to the output, $V_{out}(f)$;

$$H_{cm}(f) = \left| \frac{V_{out}(f)}{V_{cm}(f)} \right|, \quad f = 100\,\text{Hz} \Longrightarrow \text{CMRR} = \frac{A_{lf}}{H_{cm}(100)}.$$

Performances 7–10 are features of the circuit transient step response. They are calculated using the testbench shown in Fig. 1c and a transient analysis in Spectre. Performance 11 depends on the static response of the MOA (Fig. 2).

7. Rising slew rate (SR-rising): for a rising step function, SR-rising is the average rate of change in the output voltage, V_{out}, per unit of time; SR-rising is expressed in V/μs.
8. Falling slew rate (SR-falling): Similar to SR-rising, but for a falling step function.
9. Settling time when rising (TS-rising): for a rising step function, TS-rising is the time in μs required for V_{out} to settle within a 2% band about the final value.
10. Settling time when falling (TS-falling): Similar to TS-rising for a falling step function.
11. Output voltage swing (V_{swing}): the linear range of V_{out} without clipping.

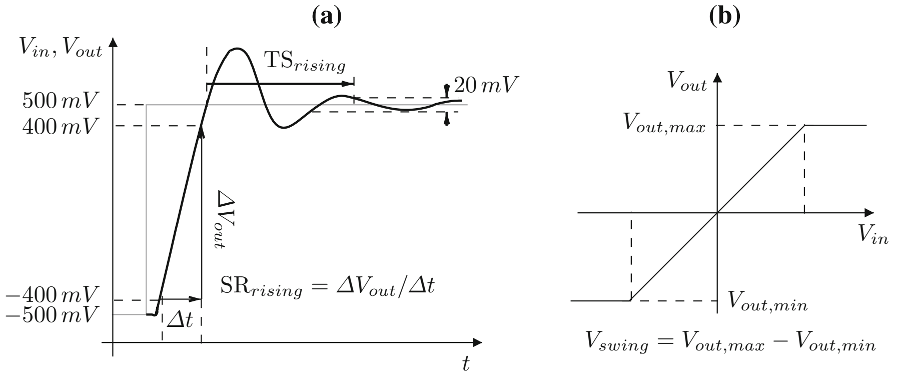

Fig. 2 **a** The transient step response for the testbench in Fig. 1c. The calculation of SR-rising and TS-rising is illustrated. **b** DC sweep analysis for the testbench in Fig. 1c. The DC value of V_{in} is swept, V_{out} is plotted, and V_{swing} is calculated as illustrated

The MOA DC transfer characteristic is nonlinear; V_{out} is clipped if V_{in} is too large or too small. The output voltage swing limits the magnitude of the input signals that can be applied, so a large V_{swing} is desired.

Some performances are only influenced by a subset of the process parameters. We use the sensitivity analysis to identify such parameters and perform the optimization only with the relevant ones for every performance.

The experiments are performed on the computational grid available at the Technical University of Munich. The tests are performed remotely on a grid of processors, and in order to measure the speed of the two methods we report the number of simulations needed to reach a solution. Such number, in the case of DFA, also corresponds to the number of iterations. The simulation is the heaviest computational burden of the algorithms, and the other operations are negligible.

In Table 1, the results of worst-case analysis using the new DFA algorithm are reported. Both the process and the operational parameters are considered by the algorithm. In order to have a feel about the possible practical interest of the proposed derivative-free algorithm, we also report the results for WiCkeD 6.7. WiCkeD uses a deterministic local search algorithm, where the derivatives of each performance feature with respect to the input parameters are approximated with sampling techniques at each iteration step of the search process. For both the algorithms the starting point for the operational variables are their nominal values, while for the process variables the initial point is the origin in $\mathbb{R}^n$. The parameter r is set to $r = 6$ in order to assure a so called six-sigma design.

In the table we report: the performance name; the number of process and operating parameters $(n + m)$; whether the worst case is a performance minimum or maximum (max/min); the nominal performance value (f_0); the number of evaluations and the performance at worst-case for the DFA algorithm; the number of evaluations and the performance at worst-case for the WiCkeD algorithm;

The results reported in Table 1 seem to indicate that DFA is capable of substantially improve the objective function with relatively few simulations. In particular the method

Table 1 Worst-case analysis results for the DFA and WiCkeD

Performance	n + m	max/min	f_0	DFA (new)		WiCkeD	
				EVALS	$f_{l/u}$	EVALS	$f_{l/u}$
A_{lf}	11	min	79.683	49	75.232	53	75.271
PM	11	min	82.439	73	80.9180	66	80.920
f_{UGBW}	11	min	990.8	73	584.2	54	586.2
CMRR	7	min	87.887	28	78.7481	303	78.196
PSRR-V_{dd}	7	min	122.096	29	87.321	319	83.756
PSRR-V_{ss}	7	min	83.536	29	76.134	52	76.094
SR-rising	8	min	0.5661	33	0.347	70	0.340
SR-falling	6	max	– 0.622	24	– 0.388	42	– 0.386
TS-rising	8	max	1.760	32	2.899	72	2.901
TS-falling	6	max	1.583	23	2.529	67	2.531
V_{swing}	5	min	1.675	14	1.387	38	1.390

is able to quickly decrease the value of the objective function in the first iterations, while the majority of iterations are necessary to assure the convergence to a stationary point. Such behavior shows a capability to easily recognize the process parameters that have the highest influence on the performance. Such propriety of the algorithm can assure the possibility to reach a good point even with a looser stopping criterium.

For what regards the comparison with WiCkeD, it is possible to notice that for the performances V_{swing}, TS-falling, TS-rising, SR-falling, PSRR-V_{ss} and CMRR the two algorithms reach similar values of the objective function, but DFA saves a significant number of simulations.

For the PSRR-V_{dd}, wicked reaches a better value of the objective function, but DFA takes one tenth of the simulations to reach the reported solutions.

For the performance A_{lf} the two algorithms perform in a similar way with DFA being a little better for the value of the objective function and the number of simulations, while for the PM the DFA takes little more simulations than WiCkeD to reach the solution.

Finally for the f_{UGBW} DFA reaches a better value of the objective function with few more simulations.

Therefore the results show that DFA is able to find solutions that consistently improve the value of the objective function in the initial point. Furthermore, the comparison to the results obtained by using WiCkeD (that is the current software used in industrial applications for circuit optimization) seems to show that the proposed method is efficient in term of average number of simulations needed to reach a solution.

In conclusions this preliminary numerical experimentation suggest that the definition of specialized derivative-free algorithms could be an interesting approach for getting good values of the objective function by requiring a reasonable and efficient computational burden.

References

1. Abramson, M.A., Audet, C., Dennis Jr., J.E., Digabel, S.: Orthomads: a deterministic mads instance with orthogonal directions. SIAM J. Optim. **20**(2), 948–966 (2009)
2. Allen, P.E., Holberg, D.R.: CMOS Analog Circuit Design. Oxford University Press, Oxford (2010)
3. Antreich, K., Eckmueller, J., Graeb, H., Pronath, M., Schenkel, F., Schwencker, R., Zizala, S.: WiCkeD: analog circuit synthesis incorporating mismatch. In: Proceedings of the IEEE Custom Integrated Circuits Conference, pp. 511–514 (May 2000)
4. Audet, C., Dennis Jr., J.E.: A progressive barrier for derivative-free nonlinear programming. SIAM J. Optim. **20**(1), 445–472 (2009)
5. Bazaraa, M.S., Sherali, H.D., Shetty, C.M.: Nonlinear Programming: Theory and Algorithms. Wiley, Hoboken (2013)
6. Bertsekas, B.P.: Nonlinear Programming. Athena Scientific, Belmont (1999)
7. Cadence Design Systems GmbH: Virtuoso spectre circuit simulator and accelerated parallel simulator user guide. In: Product Version 13.1.1, (April 2014). https://www.cadence.com
8. Ciccazzo, A., Di Pillo, G., Latorre, V., et al.: A SVM Surrogate Model Based Method for Yield Optimization in Electronic Circuit Design. Technical Report, Department of Computer, Control and Management Engineering, Universita'degli Studi di Roma" La Sapienza" (2015)
9. Ciccazzo, A., Latorre, V., Liuzzi, G., Lucidi, S., Rinaldi, F.: Derivative-free robust optimization for circuit design. J. Optim. Theory Appl. **164**(3), 842–861 (2013)
10. Conn, A.R., Scheinberg, K., Vicente, L.N.: Introduction to Derivative-Free Optimization, vol. 8. SIAM, Philadelphia (2009)
11. Eshbaugh, K.S.: Generation of correlated parameters for statistical circuit simulation. IEEE Trans. Comput. Aided Des. Integr. Circuits Syst. **11**(10), 1198–1206 (1992)
12. Fasano, G., Liuzzi, G., Lucidi, S., Rinaldi, F.: A linesearch-based derivative-free approach for nonsmooth constrained optimization. SIAM J. Optim. **24**(3), 959–992 (2014)
13. Gao, W., Hornsey, R.: A power optimization method for CMOS Op-Amps using sub-space based geometric programming. In: Proceedings of the IEEE Conference on Design, Automation and Test in Europe, pp. 508–513 (March 2010)
14. Gielen, G.G.E., Walscharts, H.C.C., Sansen, W.M.C.: Analog circuit design optimization based on symbolic simulation and simulated annealing. IEEE J. Solid State Circuits **25**(3), 707–713 (1990)
15. Graeb, H.: Analog Design Centering and Sizing, Chapter 5, pp. 90–99. Springer, Berlin (2007)
16. Kesidis, G.: Analog optimization with Wong's stochastic neural network. IEEE Trans. Neural Netw. **6**(1), 258–260 (1995)
17. Kolda, T.G., Lewis, R.M., Torczon, V.: Optimization by direct search: new perspectives on some classical and modern methods. SIAM Rev. **45**(3), 385–482 (2003)
18. Kundert, K., Chang, H., Jefferies, D., Lamant, G., Malavasi, E., Sendig, F.: Design of mixed-signal systems-on-a-chip. IEEE Trans. Comput. Aided Des. Integr. Circuits **19**(12), 1561–1571 (2000)
19. Lucidi, S., Sciandrone, M.: A derivative-free algorithm for bound constrained optimization. Comput. Optim. Appl. **21**(2), 119–142 (2002)
20. Lucidi, S., Sciandrone, M.: On the global convergence of derivative-free methods for unconstrained optimization. SIAM J. Optim. **13**(1), 97–116 (2002)
21. Lucidi, Stefano, Sciandrone, Marco, Tseng, Paul: Objective-derivative-free methods for constrained optimization. Math. Program. **92**(1), 37–59 (2002)
22. MunEDA GmbH: Manual for WiCkeD 6.7 (2014). https://www.muneda.com
23. Morshed, T.H., et al.: BSIM4v4. 7 MOSFET model. Department of Electrical Engineering and Computer Science, University of California, Berkeley, CA, USA, Technical Report (2011)
24. Nye, W., Riley, D., Sangiovanni-Vincentelli, A., Tits, A.: DELIGHT.SPICE: an optimization-based system for the design of integrated circuits. IEEE Trans. Comput. Aided Des. Integr. Circuits **7**, 501–519 (1988)
25. Powell, M.J.D.: A direct search optimization method that models the objective and constraint functions by linear interpolation. In: Advances in Optimization and Numerical Analysis, pp. 51–67. Springer (1994)
26. Somani, A., Chakrabarti, P.P., Patra, A.: An evolutionary algorithm-based approach to automated design of analog and RF circuits using adaptive normalized cost functions. IEEE Trans. Evol. Comput. **11**(3), 336–353 (2007)