

A Random and Pseudo-Gradient Approach for Analog Circuit Sizing with Non-Uniformly Discretized Parameters

Michael Pehl, Tobias Massier, Helmut Graeb, and Ulf Schlichtmann
Institute for Electronic Design Automation
Technische Universität München
Department for Electrical Engineering and Information Technology
Arcisstr. 21, 80 333 Munich, Germany
christian.michael.pehl@mytum.de

Abstract— Many methods for analog circuit sizing are available as commercial, in-house and academic tools. They are based on continuous optimization, e.g., of transistor geometries, although the subsequent layout step requires values on a pre-defined grid. In addition, sizing of transistors for bipolar and RF circuits frequently necessitates the use of multiples of predefined values for the design parameters. This paper presents a novel method for solving this type of discrete optimization problem. An iterative approach is presented, which is based on pseudo-gradients and a randomized calculation of search regions and steps. Experimental comparisons with simulated annealing and a continuous sizing approach with subsequent discretization clearly show the effectivity and efficiency of the presented method.

I. INTRODUCTION

Typical analog systems contain hundreds of analog circuit blocks with up to about 100 transistors each [14]. Due to their high complexity, designing such systems still requires a lot of hand design effort. One important design step is sizing analog circuits, such that performance specifications and the yield requirements are fulfilled. To automate this step, multi-objective optimization approaches have been developed and commercialized. The published approaches to analog sizing are based on optimization on a continuous-valued parameter domain [14][2][7][10][1].

In practice however, some design parameter domains are non-uniformly discretized. Typical examples can be given as follows:

- In analog layout, the geometric sizes of subcircuit transistors must have rational proportions, to enable a manufacturing of transistors as folded transistors or comparable structures, which allow a reduction of mismatch and noise. Accordingly the geometric sizes of the transistors must be on a pre-defined grid.
- Despite the existence of geometrically scalable models like the HICUM model [15], bipolar design is frequently done with non-scalable models. Circuit sizing then requires the selection of transistors from a library of basic transistors and the determination of the number of parallelized transistors, which can be represented by a integer multiplier.
- In deep submicron design, transistors are used, whose widths are scaled in discrete steps, e.g., FinFETs [6].

So analog circuit sizing intrinsically is a nonlinear optimization problem with non-uniformly discretized parameters. It seems that this important analog design problem has not been tackled in literature until now. So far, analog sizing is considered a continuous optimization problem. The real-valued solutions are simply rounded to the nearest available value, e.g., before entering the layout design [4]. It is obvious that this approach is arbitrary and produces non-optimal solutions. The results in Section IV.A demonstrate that the rounding can even turn into a violation of the previously satisfied specifications and constraints.

In this paper, a novel approach is presented that treats analog circuit sizing as an optimization on non-uniformly discretized parameter domains. As the problem is discrete, it could be formulated as a nonlinear integer program (NLIP) for which a large number of approaches exist, mainly from the field of combinatorial optimization [11][8] and evolutionary algorithms (e.g., [9]). For instance, in analog synthesis by use of symbolic analyses a branch-and-bound method has been used in the past [12][13]. However, if a NLIP were used we would turn physical values into uniform integer values and lose the information about the distance between two parameter values in physical units. On the contrary, the presented approach utilizes these distances in pseudo-gradients. These pseudo-gradients are used to mirror the search-direction and step length calculation of continuous gradient-based optimization on a mixed deterministic randomized search algorithm.

The paper is organized as follows. In Section II, the optimization problem is formulated. The iterative method for calculating a new parameter set is presented in Section III. In Section IV, experimental results are shown. Section V concludes the paper.

II. PROBLEM FORMULATION

A. Example

The LNA (Low Noise Amplifier) given in Fig. 1 is a typical example for illustrating analog circuit sizing. Two specifications, the forward voltage gain S_{21} and the bandwidth BW of this LNA, are given:

$$S_{21} \geq 15\text{dB} \quad \text{and} \quad BW \geq 1.5\text{GHz} .$$

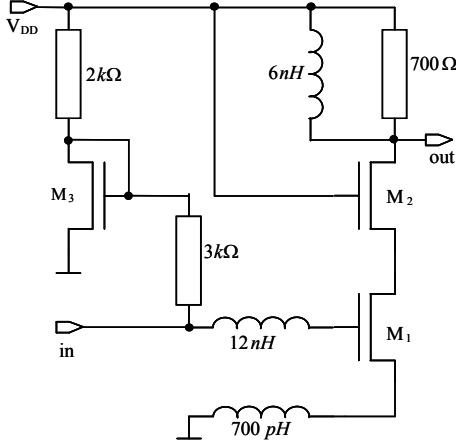


Fig. 1. LNA (Low Noise Amplifier)

Without loss of generality, we will assume in the following that circuit sizing aims at fulfilling these specifications and that a sizing process stops at once when all specifications have been met. Nevertheless, the algorithm presented in this paper can be applied to other problem formulations.

In order to fulfill the LNA's specifications, the channel geometries of the transistors M_1, M_2, M_3 are tuned. The resulting design parameters are the numbers of parallel transistors,

$$m_1, m_2, m_3 \in \{1, 2, \dots, 100\}$$

which refers to channel widths on a uniform grid, and the non-uniformly discretized lengths,

$$l_1, l_2, l_3 \in \{120\text{nm}; 180\text{nm}; 240\text{nm}; 350\text{nm}; 560\text{nm}\}$$

which refers to different transistor models to be used for simulation.

B. Discrete parameter values

Each design parameter \tilde{d}_i of the circuit can take one of n_{d_i} discrete real values from set \tilde{D}_i :

$$\tilde{d}_i \in \tilde{D}_i = \{\tilde{d}_{i,1}, \dots, \tilde{d}_{i,n_{d_i}}\} \quad (1)$$

For instance, the length of transistor M_1 of the LNA in Section II.A can be given as:

$$\tilde{d}_1 := l_1 \in \{120\text{nm}, 180\text{nm}, 240\text{nm}, 350\text{nm}, 560\text{nm}\}$$

Normalized values for each parameter can be obtained by:

$$d_i = \frac{\tilde{d}_i - \tilde{d}_{i,\min}}{\tilde{d}_{i,\max} - \tilde{d}_{i,\min}}; \quad i = 1, \dots, n_{d_i} \quad (2)$$

$$\text{with } \tilde{d}_{i,\min} = \min \tilde{D}_i; \quad \tilde{d}_{i,\max} = \max \tilde{D}_i$$

For each parameter, a normalized discrete set D_i can be given, which can be ordered by the relation <:

$$d_i \in D_i := (D_i, <) = \{d_{i,1}; \dots; d_{i,n_{d_i}}\}; \quad \bigvee_{l=1}^{n_{d_i}-1} d_{i,l} < d_{i,l+1} \quad (3)$$

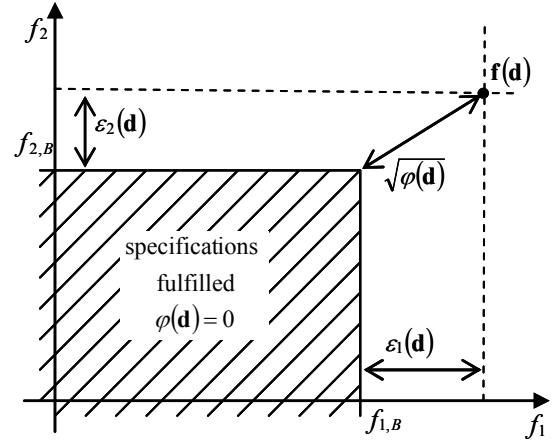


Fig. 2. Illustration of the used objective function (6), (5)

For instance, the normalized set for l_1 can be given by:

$$d_1 \in \{0.0, 0.14, 0.27, 0.52, 1.0\}$$

For an optimization problem with N_d parameters, the normalized domain can be written as:

$$D^{N_d} = \bigtimes_{l=1}^{N_d} D_l \quad (4)$$

$\mathbf{d} \in D^{N_d}$ is a point inside this basic set.

C. Discrete analog sizing problem

For each $\mathbf{d} \in D^{N_d}$, a set of n_f circuit performances $\tilde{f}_1(\mathbf{d}), \dots, \tilde{f}_{n_f}(\mathbf{d})$ can be calculated by simulation. Circuit function is defined by a set of specifications of these performances. Without loss of generality, these bounds can be given as a set of n_B normalized inequalities $f_k(\mathbf{d}) \leq f_{k,B}$. An error $\varepsilon_k(\mathbf{d})$ is defined, which describes the deviation of each performance from its specification. This error equals 0 if the specifications are fulfilled:

$$\varepsilon_k(\mathbf{d}) = \begin{cases} f_k(\mathbf{d}) - f_{k,B}; & \text{if } f_k(\mathbf{d}) > f_{k,B} \\ 0; & \text{else} \end{cases} \quad (5)$$

The sum over the squared errors is used as the scalarized objective function of circuit sizing:

$$\varphi(\mathbf{d}) = \sum_{k=1}^{n_B} \|\varepsilon_k(\mathbf{d})\|^2 \quad (6)$$

According to the formulation of the error in (5), an over-achievement of the specifications does not influence the value of the objective function $\varphi(\mathbf{d})$ and is consequently ignored during the optimization (Fig. 2).

Using the definitions above, the task of sizing an analog circuit with non-uniformly discretized parameters can be formulated as:

$$\min_{\mathbf{d} \in D^{N_d}} \varphi(\mathbf{d}) \text{ s.t. } \mathbf{c}(\mathbf{d}) \geq \mathbf{0} \quad (7)$$

Where $\mathbf{c}(\mathbf{d})$ is a set of constraints which ensure the avoidance of pathological circuits [3].

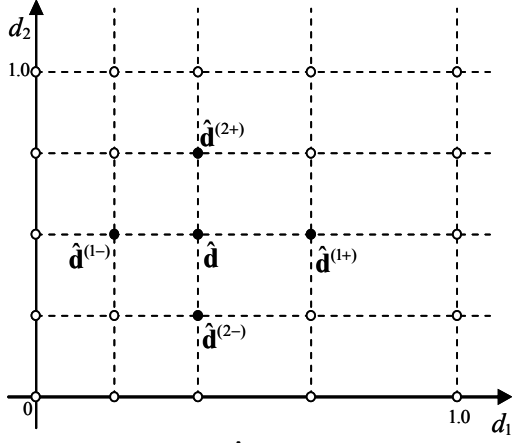


Fig. 3. Current parameter point $\hat{\mathbf{d}}$ and its next neighbors $\hat{\mathbf{d}}^{(1+)}$, $\hat{\mathbf{d}}^{(1-)}$, $\hat{\mathbf{d}}^{(2+)}$ and $\hat{\mathbf{d}}^{(2-)}$

III. DISCRETE SIZING ALGORITHM

A. Discrete pseudo-gradients

In the presented method, gradients shall be considered. However, an actual gradient cannot be calculated if parameter values are discrete. Therefore, a new method for calculating discrete pseudo-gradient is introduced, which is based on the concept of *next neighbors*. The term next neighbor is used for a neighboring parameter point which differs from the current point $\hat{\mathbf{d}}$ in exactly one component (Fig. 3).

To describe these points, the vector \mathbf{e}_i is defined:

$$\mathbf{e}_i := [e_1, \dots, e_l, \dots, e_{N_d}]^T; \quad e_l = \begin{cases} 1; & \text{if } l = i \\ 0; & \text{if } l \neq i \end{cases} \quad (8)$$

Furthermore, d_{i,a_i} with index a_i denotes the current value of the i th component of $\hat{\mathbf{d}}$:

$$\hat{d}_i = d_{i,a_i} \in \{d_{i,1}, \dots, d_{i,a_i}, \dots, d_{i,n_{d_i}}\} = D_i \quad (9)$$

Hence, the next neighbor $\hat{\mathbf{d}}^{(i+)}$ of $\hat{\mathbf{d}}$ in positive direction of component d_i is defined as:

$$\hat{\mathbf{d}}^{(i+)} := \begin{cases} \hat{\mathbf{d}} + \mathbf{e}_i \cdot (d_{i,a_i+1} - d_{i,a_i}); & \text{if } a_i \in \{1, \dots, |D_i| - 1\} \\ \hat{\mathbf{d}} & \text{if } a_i = |D_i| \end{cases} \quad (10)$$

Analogously, the next neighbor $\hat{\mathbf{d}}^{(i-)}$ of $\hat{\mathbf{d}}$ in negative direction of component d_i is defined as:

$$\hat{\mathbf{d}}^{(i-)} := \begin{cases} \hat{\mathbf{d}} - \mathbf{e}_i \cdot (d_{i,a_i} - d_{i,a_i-1}); & \text{if } a_i \in \{2, \dots, |D_i|\} \\ \hat{\mathbf{d}} & \text{if } a_i = 1 \end{cases} \quad (11)$$

In case that d_{i,a_i} is the largest or smallest value in the domain, respectively, the next neighbor of $\hat{\mathbf{d}}$ concerning d_i is the current point itself. This definition avoids a case differentiation in the calculation of a pseudo-gradient at the bounds of the domain D_i . Based on (10), (11), a definition of a discrete pseudo-gradient is introduced which corresponds to the central form of the finite-difference

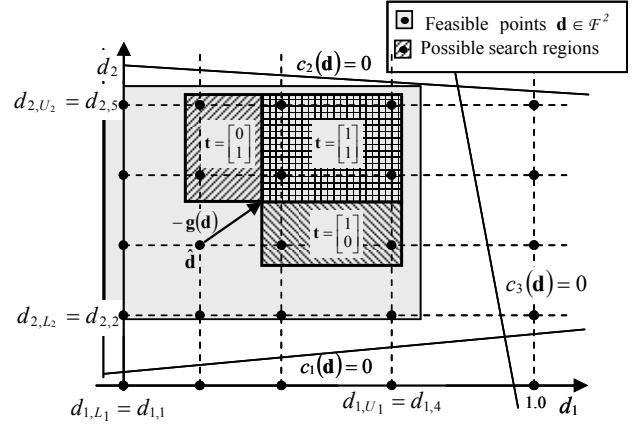


Fig. 4. Illustration of feasible region (Section III.B) and search region (Section III.C)

approximation:

$$\mathbf{g} = [g_1, \dots, g_i, \dots, g_{N_d}]^T$$

$$g_i := \frac{\varphi(\hat{\mathbf{d}}^{(i+)}) - \varphi(\hat{\mathbf{d}}^{(i-)})}{\mathbf{e}_i^T \cdot (\hat{\mathbf{d}}^{(i+)} - \hat{\mathbf{d}}^{(i-)})}; \quad i = 1, \dots, N_d \quad (12)$$

This approach provides a good approximation of the gradient. It can be seen that the definition of the next neighbors (10) and (11) leads to the forward or backward difference quotient at the bounds of the domain D_i .

B. Discrete feasible region

In this section, the feasible region of discrete parameter values defined by the constraints in (7) will be approximated. The bounds of this simply connected region are approximated by linearization of the constraint functions using their pseudo-gradients. The pseudo-gradients are obtained by substituting $\varphi(\mathbf{d})$ in (12) with the considered constraint function $c_m(\mathbf{d})$. Using the resulting pseudo-gradient \mathbf{g}_{c_m} , each constraint is linearized:

$$c_m(\mathbf{d}) \approx c_m(\hat{\mathbf{d}}) + \mathbf{g}_{c_m}^T \cdot (\mathbf{d} - \hat{\mathbf{d}}) \quad (13)$$

For calculating the borders of the feasible region, the linear model of each constraint (13) is set to 0. An estimation of the value \bar{d}_{i,c_m} for which $c_m(\hat{\mathbf{d}} + \mathbf{e}_i \cdot \bar{d}_{i,c_m}) = 0$ holds, can be formulated by:

$$\bar{d}_{i,c_m} = -\frac{c_m(\hat{\mathbf{d}})}{\mathbf{e}_i^T \cdot \mathbf{g}_{c_m}(\hat{\mathbf{d}})} \quad (14)$$

For each parameter d_i , a distance $\bar{d}_{i,U}$ from the current parameter point to the supremum and a distance $\bar{d}_{i,L}$ to the infimum of the feasible region can be selected from all values \bar{d}_{i,c_m} . Considering n_c constraints, the selection criterion for these distances can be formulated:

$$\bar{d}_{i,U} = \min_{m=1, \dots, n_c} \bar{d}_{i,c_m} \text{ s.t. } \bar{d}_{i,c_m} \geq 0$$

$$\bar{d}_{i,L} = \max_{m=1, \dots, n_c} \bar{d}_{i,c_m} \text{ s.t. } \bar{d}_{i,c_m} \leq 0 \quad (15)$$

The resulting bounds are used to define the set of feasible parameter points \mathcal{F}^{N_d} forming the discrete feasible region (Fig. 4).

$$\mathcal{F}_i = \left\{ d_{i,a_i} + \bar{d} \mid \bar{d}_{i,L} \leq \bar{d} \leq \bar{d}_{i,U} \right\} \cap D_i = \{d_{i,L_i}, \dots, d_{i,U_i}\} \quad (16)$$

$$\mathcal{F}^{N_d} = \bigtimes_{i=1}^{N_d} \mathcal{F}_i$$

The indices L_i and U_i correspond to the indices of the corresponding values of d_i in the domain D_i . Note that $\mathcal{F}_i = (\mathcal{F}_i, <)$ is an ordered set. For instance, if the domain is given as $D_1 = \{0.0, 0.14, 0.27, 0.52, 1.0\}$ and the feasible set is calculated as $\mathcal{F}_1 = \{0.27, 0.52, 1.0\}$, the values of L_1 and U_1 are 3 and 5 respectively.

C. Discrete search region

In the previous sections, discrete analogies to gradients and constraint regions were developed. In the following, a new iterative method for discrete optimization of analog circuits is presented. It consists of two iteratively repeated steps, search region calculation and calculation of a new point. According to the concept of search direction in continuous-value domains, the discrete search region is defined as a sub-region of the feasible region according to the direction of the steepest pseudo-descent $-\mathbf{g}$ (Fig. 4). However, not every direction $-g_i \neq 0$ is chosen in every step. In fact, every direction $-g_i$ is taken with a certain probability, which corresponds to the proportion of the component g_i to the length of the gradient. If a direction is chosen, the corresponding component of a vector of tendency \mathbf{t} – which is initialized by 0 – is set to $-\text{sgn}(g_i(\hat{\mathbf{d}}))$. The probability for choosing a direction for the search region is given by:

$$P(t_i) = P(t_i = -\text{sgn}(g_i(\hat{\mathbf{d}}))) = \frac{|g_i(\hat{\mathbf{d}})|}{\|\mathbf{g}(\hat{\mathbf{d}})\|} \quad (17)$$

The vector of tendency \mathbf{t} is created via $P(t_i)$ and N_d uniformly distributed random numbers $z_i \sim U(0,1)$:

$$\mathbf{t} = [t_1, \dots, t_i, \dots, t_{N_d}]^T$$

$$t_i = \begin{cases} -\text{sgn}(g_i(\hat{\mathbf{d}})); & \text{if } z_i \leq P(t_i = -\text{sgn}(g_i(\hat{\mathbf{d}}))) \\ 0; & \text{else} \end{cases} \quad (18)$$

Of course, the search region spanned by \mathbf{t} is bounded by the discrete feasibility region (Section III.B). The new point \mathbf{d}_{new} is created by scaling the components of \mathbf{t} with positive values. The domain of feasible scaling factors – called step sizes – for component t_i is:

$$S_i = \left\{ s_i = |d_i - d_{i,a_i}| \mid d_i \in \mathcal{F}_i \wedge (t_i \cdot d_i > t_i \cdot d_{i,a_i}) \right\} \\ = \{s_{i,1}, \dots, s_{i,n_s}\} \quad (19)$$

Note that $s_{i,1}$ is the smallest and s_{i,n_s} the largest possible step size, respectively, in the ordered set $S_i := (S_i, <)$.

It should be pointed out that $\mathbf{t} \neq \mathbf{0}$ must be ensured. The described method for choosing \mathbf{t} is repeated until this condition is fulfilled. This can be a disadvantage, if the

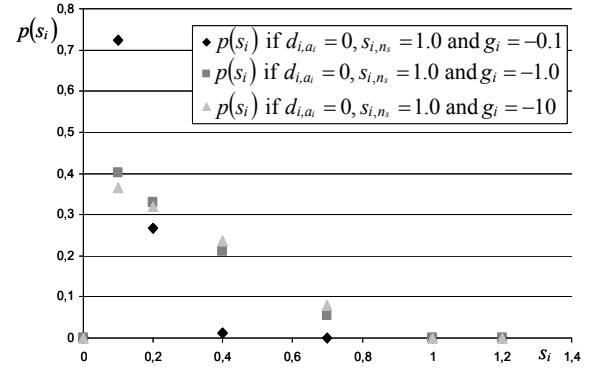


Fig. 5. Probability density function $p(s_i)$ for different values of g_i and $d_i \in \{0.0, 0.14, 0.27, 0.52, 1.0\}$

number of parameters is high and the absolute values of the components $|g_i|$ are simultaneously nearly equal. Then, the probability for choosing any component is very small. However, a vector $\mathbf{t} \neq \mathbf{0}$ has always been found after a few tries in the example results in Section IV.

D. Calculating a new point

The new point \mathbf{d}_{new} is created by scaling the components of the tendency vector \mathbf{t} (18) with positive values. The step size s_i is taken from the domain S_i (19) by a random based approach, which considers the pseudo-sensitivity $|g_i|$ for each component t_i . We propose the following probability density function for choosing a step size s_i :

$$p(s_{i,l}) = \begin{cases} \alpha_i + \beta_i \cdot \exp(-\gamma_i s_{i,l}), & \text{if } 0 < s_{i,l} \leq s_{i,n_s} \\ 0 & ; \text{else} \end{cases} \quad (20)$$

The probability should be large for components t_i that correspond to a large pseudo-sensitivity $|g_i|$ (Fig. 5). This is achieved by defining γ_i as:

$$\gamma_i = \frac{1}{|g_i|} \quad (21)$$

For calculating the values α_i and β_i in (20), two requirements are used:

1. The probability density for all step sizes s_i greater than the maximum step size s_{i,n_s} has to be zero, i.e., $p(s_i) = 0$. At the border of the domain, s_{i,n_s} , $p(s_{i,n_s}) \neq 0$ must hold. It is assumed that there is a larger step $s_{i,n_s} + \eta_i > 0$ which fulfills:

$$p(s_{i,n_s} + \eta_i) = 0 \text{ and } \neg \exists_{\eta_i > 0} p(s_{i,n_s} + \eta) = 0 \quad (22)$$

To avoid a case differentiation at the bound of the domain D_i , the value of η_i is approximated by the mean distance of two points in direction of component t_i :

$$\eta_i = \frac{s_{i,n_s}}{|S_i|} \quad (23)$$

2. The sum over all probabilities $p(s_i)$ with $0 < s_i \leq s_{i,n_s}$ has to be 1.

Initialise starting point $\hat{\mathbf{d}}$
While specifications are not fulfilled
Calculate pseudo-gradients \mathbf{g} and \mathbf{g}_{c_m} , Eq. (12)
Calculate feasibility region \mathcal{F}^{N_d} , Eq. (16)
Choose searching area, Eq. (18)
Calculate step size s_i , Eq. (26)
Calculate new point \mathbf{d}_{new} , Eq. (27)
Set $\hat{\mathbf{d}} = \mathbf{d}_{new}$

Fig. 6. Simplified algorithm for discrete optimization

With (20) and requirement 1, α_i can be defined as:

$$\alpha_i = -\beta_i \cdot \exp(-\gamma_i \cdot (s_{i,n_s} + \eta_i)) \quad (24)$$

Requirement 2 and (24) lead to:

$$\beta_i = \left(\left(\sum_{l=1}^{|S_i|} \exp(-\gamma_i \cdot |s_{i,l}|) \right) - |S_i| \cdot \exp(-\gamma_i \cdot |s_{i,n_s} + \eta_i|) \right)^{-1} \quad (25)$$

The step size s_i for component direction t_i is defined by a uniformly distributed random number $Z \sim U(0,1)$:

$$s_i = \min_{s_{i,L} \in S_i} s_{i,L} \text{ s.t. } \sum_{l=1}^L p(s_{i,l}) \geq Z \quad (26)$$

Note that the different sensitivities for different directions t_i are considered by using exactly one random number Z per iteration..

Now the new parameter point \mathbf{d}_{new} can be calculated:

$$\mathbf{d}_{new} = \hat{\mathbf{d}} + \sum_{i=1}^{N_d} t_i \cdot s_i \cdot \mathbf{e}_i \quad (27)$$

The overall procedure for discrete analog sizing with the described method is sketched in Fig. 6.

Note that the constraints may be violated at the new point. Hence, an adaptive constraint check must be done after calculating a new point. If a constraint has been violated, in the direction $\mathbf{d}_{new} - \hat{\mathbf{d}}$ the point closest to \mathbf{d}_{new} that satisfies the constraints is chosen. Furthermore, to avoid convergence at local optimal points, which do not fulfill the specifications, and which lie at the bounds of the discrete feasible region, an adaptive methodology must be applied.

IV. EXAMPLES

A. Sizing of an LNA

The LNA from Section II.A is optimized with the presented discrete sizing algorithm. The non-uniformly discretized lengths l_1, l_2 and l_3 – which correspond to different simulation models – and the uniformly discretized multipliers m_1, m_2 and m_3 of the transistors formed 6 discretized optimization parameters. The discretized parameter domains from Section II.A yield a discrete optimization problem with cardinality $1.25 \cdot 10^8$. The initial performance values, $S_{21} = 9.11\text{dB}$ and $BW = 1.47\text{GHz}$,

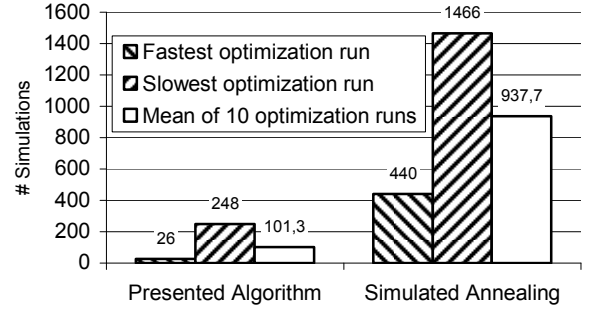


Fig. 7. Comparison of the presented algorithm and SA. The graph shows the numbers of simulations until the specifications were met.

violate the specifications in Section II.A.

A Simulated-Annealing Algorithm (SA) [5] with a cooling rate of $r = 0.005$ and a start temperature of $T_0 = 1.0$ was applied to the LNA for comparison purposes. In the iterations of the SA, every discrete parameter value could have been tuned to the next higher or lower value.

The number of iterations which were needed to find a point that fulfills the specifications is given in Fig. 7.

It can be seen that the presented algorithm was faster than SA. Note that both random based algorithms exhibit a fluctuation in the simulation effort. The presented algorithm had a speed-up of 17 with respect to SA concerning the fastest optimization run, of 10 concerning the mean of all optimization runs, and of 6 concerning the slowest optimization run. Even the slowest run of the new algorithm was 2 times faster than the fastest SA run. Note that other temperatures and cooling rates have led to even larger speed-ups for the new algorithm.

B. Sizing of a BiCMOS OTA

A BiCMOS OTA is given in Fig. 8. The discrete design parameter values for the considered problem are given in Table 1, the specifications are shown in Table 2. The parameters from Table 1 form a discrete optimization problem with cardinality $2.80 \cdot 10^{30}$.

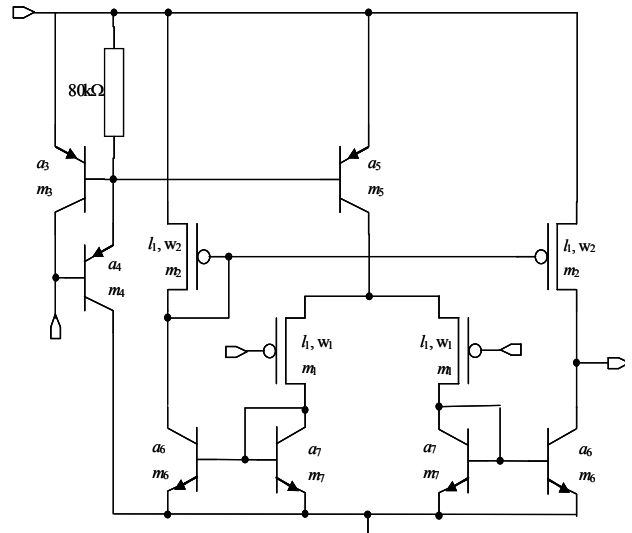


Fig. 8. BiCMOS OTA (operational transconductance amplifier)

Table 1. Parameters and basic sets for the optimization

Parameter	Basic set
w_1, w_2, w_3	$w_i \in \{3\mu m, 4\mu m, \dots, 250\mu m\}$
l_1	$l_i \in \{460nm, 480nm, 10\mu m\}$
a_3, a_4, a_5, a_6, a_7	$a_i \in \{0.5, 0.75, \dots, 50\}$
$m_1, m_2, m_3, m_4, m_5, m_6, m_7$	$m_i \in \{1, 2, \dots, 20\}$

Table 2. Specifications for the optimization

Performance	Specification	Starting Value
<i>DCGain</i> [dB]	>40	17.4
<i>CMRR</i> [dB]	>80	81.0
<i>PSRR</i> [dB]	>80	54.4
f_T [MHz]	>45	7.8
<i>PHM</i> [°]	>60	79.3
<i>SR+</i> [V/μs]	>6.0	1.54

Table 3. Comparison of the presented algorithm (NEW), with a state-of-the-art continuous sizing algorithm (CONT) with subsequent rounding to the nearest point (RNP) or to the nearest point in direction of steepest improvement (RSI)

Method	Simulations	Solution
CONT + RNP	1701	Failed: 1 spec violated
CONT + RSI	1716	Failed: 1 spec and 3 constraints violated
NEW	1048	o.k.: specs fulfilled
CONT + NEW	2018	o.k.: specs fulfilled

In this example, the presented algorithm (denoted with NEW in Table 3) was compared with a state-of-the-art continuous optimization algorithm (denoted with CONT) and subsequent rounding to the next discrete point. The rounding was done either by taking the nearest discrete point (RNP) or by taking the nearest point in direction of steepest improvement (RSI). Table 3 summarizes the results.

We can see that CONT with subsequent rounding failed because of a resulting violation in specifications and/or constraints. This makes clear that a continuous optimization with subsequent rounding can produce false results. On the other hand, the new method succeeds to provide a discretized solution that fulfils the specifications.

We can also see from Table 3 that the new algorithm as a stand-alone tool was twice as fast as in connection with a preceding continuous optimization run.

V. CONCLUSION

An iterative random and pseudo-sensitivity method for analog circuit sizing with non-uniformly discretized parameters has been presented. In each iteration step, a discrete search region is computed by randomly selecting

components of a pseudo-gradient. The probability of selecting a component is proportional to its pseudo-sensitivity. In this way, a sub-region of the feasible region is spanned. For each selected parameter direction, an individual step length is randomly selected. The probability density of the step length decreases exponentially, such that larger step lengths are more probable for parameter directions with larger pseudo-sensitivity.

Even if further experiments and comparisons with state of the arte mixed integer nonlinear programming approaches are preferable, presented results show a mean speed-up of 10 compared with a simulated annealing approach and illustrate the failure of the continuous sizing method with subsequent rounding to solve this generalized sizing problem. Due to the good results of the presented method in exclusive discrete cases, the method should be extended for mixed continuous discrete problems next.

REFERENCES

- [1] S. Balkir, G. Dündar, G. Alpaydin, "Evolution Based Synthesis of Analog Integrated Circuits", *NASA/DoD Conference on Evolvable Hardware 2004*, June 2004
- [2] G. G. E. Gielen, "Design tool solutions for mixed-signal/RF circuit design in CMOS nanometer technologies", *ASP-DAC '07*, Jan. 2007
- [3] H. Graeb, S. Zizala, J. Eckmueller, K. Antreich, "The Sizing Rules Method for Analog Integrated Circuit Design", *IEEE/ACM ICCAD 2001*, Nov. 2001
- [4] E. Hjalmarson, R. Hägglund, L. Wanhammar, "An Optimization-Based Approach for Analog Circuit Design", *Proc. European Conference on Circuit Theory and Design*, Krakow, Poland, Sep. 2005
- [5] S. Kirkpatrick, C. D. Gelatt, M. P. Vecchi, "Optimization by Simulated Annealing", *Science*, vol. 220, No. 4598, May 1983
- [6] G. Knoblinger, F. Kuttner, A. Marshall, C. Russ, P. Haibach, P. Patruno, T. Schulz, W. Xiong, M. Gostkowski, K. Schrufer, C. R. Cleavelin, "Design and Evaluation of Basic Analog Circuits in an Emerging MuGFET Technology", *IEEE International SOI Conference 2005*, Oct. 2005
- [7] M. J. Krasnicki, R. Phelps, J. R. Hellums, M. McClung, R. R. A. Rutenbar, L. R. Carley, "ASF: A Practical Simulation-Based Methodology for the Synthesis of Custom Analog Circuits", *IEEE/ACM ICCAD 2001*, Nov. 2001
- [8] D. Li, X. Sun, "Nonlinear Integer Programming", Springer, 2006
- [9] N. Li, F. Ye, "Optimal Design of Discrete Structure with Directed Mutation Genetic Algorithms", *WCICA 2006*, Juni 2006
- [10] F. Medeiro, F. V. Fernández, R. Domínguez-Castro, A. Rodríguez-Vázquez, "A Statistical Optimization-Based Approach for Automated Sizing of Analog Cells", *IEEE/ACM ICCAD*, Nov. 1994
- [11] G. L. Nemhauser, L. A. Wolsey, "Integer and Combinatorial Optimization", John Wiley & Sons, Inc., 1988
- [12] P. C. Maulik, L. R. Carley, R. A. Rutenbar, "A Mixed-Integer Nonlinear Programming Approach to Analog Circuit Synthesis", *29th ACM/IEEE DAC*, 1992
- [13] P. C. Maulik, L. R. Carley, R. A. Rutenbar, "Integer Programming Based Topology Selection of Cell-Level Analog Circuits" *IEEE TCAD*, vol. 14, No. 4, April 1995
- [14] R. A. Rutenbar, G. G. E. Gielen, J. Roychowdhury, "Hierarchical Modeling, Optimization, and Synthesis for System-Level Analog and RF Designs", *Proceedings of the IEEE*, vol. 95, No. 3, pp. 640-668, Mar. 2007
- [15] M. Schroter, A. Chakravorty, "HICUM – A Geometry Scalable Physics-Based Compact Bipolar Transistor model, Documentation of model level 2 version 2.2", ECE Dept - Univ. of California San Diego; Chair for Electron Devices and Integrated Circuits – Univ. of Technology Dresden, Nov. 2005