Defining Cost Functions for Robust IC Design and Optimization

Árpád Bűrmen, Janez Puhan, and Tadej Tuma
University of Ljubljana
Faculty of Electrical Engineering
Tržaška 25, SI-1000 Ljubljana, Slovenia
arpadb@fides.fe.uni-lj.si

Abstract

The ever increasing pace of analog IC design demands efficient means of automated design and optimization. Especially important is robust design. Its goal is to produce circuits whose behaviour stays within some predefined range when the manufacturing process variations and environmental effects remain bounded. Most of the design process is still handled by IC designers manually. We present a simple mathematical formulation of the robust design and optimization problem and its transformation into a constrained optimization problem by means of penalty functions. We illustrate the method on a robust differential amplifier design problem. The resulting circuits show that a computer not only can improve circuits designed by humans, but is also capable of designing a circuit with very little initial knowledge. Optimization runs resulted in circuits with similar or even better performance when compared to humanly designed circuits. The method can take advantage of parallel processing, but is still efficient enough to be run on a single computer.

1. Introduction

In order for the designers to be capable of coping with ever-shorter deadlines [6], efficient means of automated robust IC design are required. The goal of robust design is to create circuits whose characteristics remain within the called design requirements regardless of possible manufacturing process variations and changing operating conditions.

A circuit characteristic is any real quantity that can be measured on a circuit. The operating conditions (i.e. temperature, power supply voltage, etc.) are bounded and the bounds are agreed upon before the designer starts the design process. The manufacturing process variations which can occur during IC fabrication are provided by the circuit foundry. We focus on those design problems where these variations are provided in the form of corner models of circuit elements. For a CMOS process usually the following corners are provided: worst one (WO), worst zero (WZ), worst power (WP), worst speed (WS), and typical mean (TM). Corner models can also be provided for resistors, capacitors, BJTs, etc.

In the past research efficient means of automated nominal design were sought [4, 12, 10, 15]. Nominal design does not result in robust circuits. In order to obtain a robust circuit an additional step of design centering is required. Design centering techniques are either statistical [2, 9] or deterministic [1, 5, 11].

Robust design as often practised by IC designers relies on the assumption that the circuit characteristics take their extreme values at the extremes of operating conditions and process variations. A designers checks if a circuit is robust by simulating it at combinations of extreme operationg conditions and process variations. In the following sections we refer to these extreme values (models) as corner values (models). A combination of extreme values is called a corner point. We consider the nominal operating conditions and the nominal model (typical mean) as yet another corner value (model).

The above mentioned assumption is valid only if circuit characteristics are monotonic functions on the intervals enclosing operating conditions and model parameters of corner models. If this is not the case false conclusions can be made which guide the design process in the wrong direction. The probability of a false conclusion increases with the distance between neighbouring corner points. Therefore the designers sometimes check the circuit not only for the extreme values of operating conditions and process variations but also at several intermediate points. This reduces the distance between neighbouring corner points along with the probability of a false conclusion.

Another problem is the number of corner points. It increases exponentially with the number of operating conditions and process variations making an exhaustive corner search intractable. Some means of heuristic corner search must be used in order to keep the problem managable.

The remainder of this paper is divided as follows. First the robust design problem as perceived by an IC designer is mathematically formulated. The transformation of the robust design and optimization problem into a constrained optimization problem by means of penalty functions is presented. The method is illustrated by automated robust differential amplifier design. Finally the conclusions and ideas for future work are given.

2. Automated robust design

2.1. Mathematical formulation

The robust design process as perceived and practised by an IC designer is based on the notion of corner points. A corner point is a combination of corner models (that describe some manufacturing process variation) and operating conditions. Suppose that we have m different kinds of circuit elements (e.g. CMOS transistors, capacitors, resistors, ...) with a set of n_i corner models for every one of them, describing process variations that affect that particular kind of circuit element

$$\mathcal{P}_i = \{p_i^1, p_i^2, ..., p_i^{n_i}\}, i = 1, 2, ..., m. \tag{1}$$

There are $M - m$ operating conditions and for every such operating condition we have a set of n_i values that are of particular interest to the designer

$$\mathcal{P}_i = \{p_i^1, p_i^2, ..., p_i^{n_i}\}, i = m+1, m+2, ..., M. \tag{2}$$

$p_1^1, p_2^1, ..., p_m^1$ stand for the characteristics of the nominal IC fabrication process and $p_{m+1}^1, p_{m+2}^1, ..., p_M^1$ for the nominal operating conditions. The cross product of the M sets from eqs. (1) and (2) is the set of corner points $\mathcal{C}$ with cardinality $K = \prod_{i=1}^M n_i$. In general a subset of these points is examined during the process of robust design

$$\mathcal{C} = \mathcal{P}_1 \times \mathcal{P}_2 \times ... \times \mathcal{P}_M. \tag{3}$$

The performance of the circuit, (which is the result of some combination of process variations during its fabrication and operating conditions during its use), is described by a vector of N real values $\boldsymbol{y} = [y_1, y_2, ..., y_N] \in \mathbb{R}^N$.

We represent the circuit as a function that for any combination of n circuit parameters denoted by vector $\boldsymbol{x}$ and some combination of process variations and operating conditions denoted by $\boldsymbol{q}$ produces a vector of circuit characteristics $\boldsymbol{y}$.

$$\begin{aligned} D : (\boldsymbol{x}, \boldsymbol{q}) \mapsto \boldsymbol{y}, \quad & \boldsymbol{x} \in \mathbb{R}^n, \boldsymbol{q} \in \mathcal{C}, \boldsymbol{y} \in \mathbb{R}^N, \\ \boldsymbol{y}(\boldsymbol{x}, \boldsymbol{q}) = [y_1(\boldsymbol{x}, \boldsymbol{q}), & y_2(\boldsymbol{x}, \boldsymbol{q}), ..., y_N(\boldsymbol{x}, \boldsymbol{q})]. \end{aligned} \tag{4}$$

In the following sections we also use the following notation for eq. (4)

$$D_i : (\boldsymbol{x}, \boldsymbol{q}) \mapsto y_i, \quad \boldsymbol{x} \in \mathbb{R}^n, \boldsymbol{q} \in \mathcal{C}, y_i \in \mathbb{R}^N.$$

Two vectors express the design requirements: a vector of lower bounds $\boldsymbol{b} = [b_1.b_2, ..., b_N] \in \mathbb{R}^N$ and a vector of upper bounds $\boldsymbol{B} = [B_1.B_2, ..., B_N] \in \mathbb{R}^N$. For the sake of simplicity we allow for any lower bound to take the value $-\infty$, meaning that there is no lower bound on the respective circuit characteristic. Similarly any upper bound can take the value $+\infty$, meaning that no upper bound exists on the respective circuit characteristic. A circuit with circuit parameters $\boldsymbol{x}$ satisfies the design requirements for a particular corner point $\boldsymbol{q} \in \mathcal{C}$ if the following set of relations holds

$$b_i \leq y_i \leq B_i, i = 1, ..., N. \tag{5}$$

Let $g(x)$ denote some continuous monotonically increasing function defined for $x \geq 0$. The basic penalty function is defined as

$$f(x) = \begin{cases} 0 & x < 0 \\ g(x) - g(0) & x \geq 0 \end{cases}. \tag{6}$$

Eq. (6) is used to establish the relation between the robust design problem and the constrained optimization problem.

A circuit design is satisfactory if it satisfies the design requirements for all corner points from set $\mathcal{C}$.

2.2 The cost function

Since optimization is a process that strives to decrease the cost function value, the cost function must: 1. penalize designs that fail to satisfy some basic requirements ($r_\mathrm{C}(\boldsymbol{x})$), 2. penalize designs that are not robust ($r_\mathrm{P}(\boldsymbol{x})$), and 3. define the tradeoffs between individual circuit characteristics ($r_\mathrm{T}(\boldsymbol{x})$). The cumulative cost function is

$$\begin{gathered} r(\boldsymbol{x}) = r_\mathrm{C}(\boldsymbol{x}) + r_\mathrm{R}(\boldsymbol{x}) + r_\mathrm{C}(\boldsymbol{x}), \\ r_\mathrm{C}(\boldsymbol{x}) >> r_\mathrm{R}(\boldsymbol{x}) >> r_\mathrm{T}(\boldsymbol{x}). \end{gathered} \tag{7}$$

First of all one has to consider the case that the simulation itself fails to converge thus rendering the optimization incapable of determining the cost function value for a particular combination of circuit parameters. In some cases the simulator may succeed to simulate the circuit, but its performance is far from the desired performance (e.g. some of the transistors that are supposed to be in saturation, are not). In such cases an additional penalty term $r_\mathrm{C}(\boldsymbol{x})$ is introduced. The value of $r_\mathrm{C}(\boldsymbol{x})$ for such circuits should be significantly larger than the contribution of the penalty functions $r_\mathrm{P}(\boldsymbol{x})$. $r_\mathrm{C}(\boldsymbol{x})$ should be proportionate to the severity of the convergence problem (circuit performance problem).

$F(\boldsymbol{y})$ assigns a penalty to the circuit if any of its characteristics lies outside its bounds defined by the design requirements

$$F(\boldsymbol{y}) = \sum_{i=1}^{N} \{f[(y_i - B_i)/A_i] + f[(b_i - y_i)/A_i]\}. \tag{8}$$

A_i is the coefficient that sets the steepness of the penalty function for i-th design requirement. The smaller A_i is, the bigger the penalty. $F(\boldsymbol{y}) = 0$ means that all circuit characteristics satisfy the design requirements expressed by $\boldsymbol{b}$ and $\boldsymbol{B}$.

Let $\mathcal{C}_\mathrm{S} = \{\boldsymbol{s}_i : i = 1, 2, ..., K_\mathrm{H}\} \subseteq \mathcal{C}$ denote the set of examined corners. Then by means of eq. (8) we can construct $r_\mathrm{P}(\boldsymbol{x})$

$$r_\mathrm{P}(\boldsymbol{x}) = \sum_{i=1}^{K_\mathrm{H}} F\left[D(\boldsymbol{x}, \boldsymbol{q}_i)\right]. \qquad (9)$$

$K_\mathrm{H} < K$ indicates that some means of heuristic corner search is used. We propose the following scheme. First the individual influences of corner values and corner models are examined. Based on the results corners are predicted where the circuit characteristics are expected to take their minimal (maximal) values. In the first part of the search the following $\sum_{i=1}^{M}(n_i - 1) + 1$ corners are examined

$$\begin{array}{lcll} \boldsymbol{q}_\mathrm{nom} & = & \boldsymbol{s}_1^1 = \boldsymbol{s}_2^1 = ... = \boldsymbol{s}_M^1 & = (p_1^1, p_2^1, ..., p_M^1); \\ \boldsymbol{s}_1^i & = & (p_1^i, p_2^1, ..., p_M^1), & i = 2, 3, ..., n_1; \\ \boldsymbol{s}_2^i & = & (p_1^1, p_2^i, ..., p_M^1), & i = 2, 3, ..., n_2; \\ ... & & & \\ \boldsymbol{s}_M^i & = & (p_1^1, p_2^1, ..., p_M^i), & i = 2, 3, ..., n_M. \end{array} \qquad (10)$$

Based on the results, further $2N$ corners are generated (two for every circuit characteristic; one where the lowest value and one where the highest value is expected to take place) and examined

$$\begin{array}{lcll} \boldsymbol{q}_\mathrm{L}^i & = & (p_1^{l_i^1}, p_2^{l_i^2}, ..., p_M^{l_i^M}), & i = 1, 2, ..., N; \\ \boldsymbol{q}_\mathrm{H}^i & = & (p_1^{h_i^1}, p_2^{h_i^2}, ..., p_M^{h_i^M}), & i = 1, 2, ..., N. \end{array} \qquad (11)$$

Indices l_i^j and h_i^j for $i = 1, 2, ..., N$ and $j = 1, 2, ..., M$ are defined as follows

$$\begin{array}{lcl} l_i^j & = & \arg \min_{k=1,2,...,n_j} y_i(\boldsymbol{x}, \boldsymbol{s}_j^k), \\ h_i^j & = & \arg \max_{k=1,2,...,n_j} y_i(\boldsymbol{x}, \boldsymbol{s}_j^k). \end{array}$$

$K_\mathrm{H} = \sum_{i=1}^{M}(n_i - 1) + 1 + 2N$ corners are examined by utilizing the heuristic search described in eqs. (10) and (11). The price to pay is the risk of obtaining a narrower range for the circuit characteristic y_i in case the function $D_i(\boldsymbol{x}, \boldsymbol{q})$ is not monotonic on the intervals enclosing operating conditions and model parameters of corner models.

The penalty function $r_\mathrm{P}(\boldsymbol{x})$ enforces the constraints on circuit performance. Usually one also wants the circuit characteristics to be as good as possible. The 'optimal' circuit's performance is subject to performance constraints and tradeoffs between individual performance measures (eg gain, bandwidth, ...). The description of tradeoffs shouldn't affect the enforcement of constraints. In other words tradeoffs become possible only after all constraints are satisfied.

Tradeoffs are specified by $T = (T_1, T_2, ..., T_N) \in \mathbb{R}^N$. An individual circuit characteristic contributes to the tradeoff part of the cost function only if the respective performance constraint is satisfied. Tradeoffs are applied only to the nominal circuit performance (nominal operating conditions and typical mean IC fabrication process).

$$\begin{array}{rcl} r_\mathrm{T}(\boldsymbol{x}) & = & C \sum_{i=1}^{N} f\left\{\left[B_i - D_i(\boldsymbol{x}, \boldsymbol{q}_\mathrm{nom})\right]/T_i\right\} + \\ & & C \sum_{i=1}^{N} f\left\{\left[D_i(\boldsymbol{x}, \boldsymbol{q}_\mathrm{nom}) - b_i\right]/T_i\right\}. \end{array} \qquad (12)$$

Smaller values of T_i cause the optimizer to try harder to optimize the respective circuit characteristic at the expense of the remaining circuit characteristics. In case any of the coefficients $T_i = \infty$, the respective characteristic does not participate in the tradeoff optimization process. C makes the contribution of the tradeoffs to the cumulative cost function significantly smaller than the contribution of the penalty function $r_\mathrm{P}(\boldsymbol{x})$. One can view $r_\mathrm{T}(\boldsymbol{x})$ as a tradeoff plane bounded by the steep walls of performance constraints defined by $r_\mathrm{P}(\boldsymbol{x})$. The individual tradeoff coefficients T_i represent the angles between the tradeoff plane and the coordinate axes of the n-dimensional search space.

In case the $r_\mathrm{T}(\boldsymbol{x})$ is omitted from eq. (7), the optimization algorithm will search for a circuit that satisfies the performance constraints. As soon as some circuit with cost function value 0 is found, the optimization can be stopped.

In case the complete expression in eq. (7) is used, a search for a circuit satisfying all design requirements is conducted upon which tradeoffs among individual performance measures are applied and the circuit is further optimized in order to improve its performance at nominal operating conditions. In this case some other stopping condition must be used (i.e. optimization is stopped as soon as simplex size, population diameter, steplength, etc. become small enough). Generally such optimization takes longer to complete.

For the process of optimization any box-constrained optimization method can be used since we only need to constrain circuit parameters such as transistor widths and lengths to intervals of possible values. The implicit constraints arising from the design requirements are handled by the penalty functions and are an integral part of the cost function.

3 Results

To illustrate the method, robust design has been applied to the circuit structure in Fig. 1 [8]. The circuit is an amplifier with differential input, differential output and common

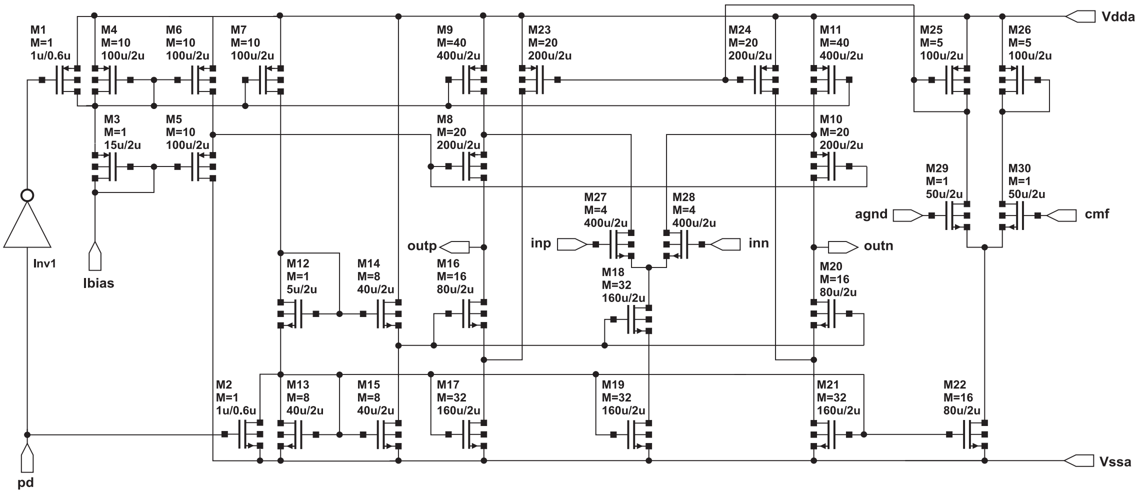

Figure 1. The differential amplifier circuit taken from a real world application. W/L and M values were designed by an experienced IC designer.

mode feedback. The M and W/L values of transistors in Fig. 1 (reference circut) were designed by an IC designer.

Let V_{ds} and V_{dsat} denote the drain-source voltage and the drain-source saturation voltage. For p-MOS they represent the absolute values of respective quantities. $\mathcal{M}_{\mathrm{rel}}$ denotes the set of all MOS transistors in the circuit except M_1, M_2, and transistors in Inv_1. The saturation measure is defined as $P_{\mathrm{sat}} = \sum_{M \in \mathcal{M}_{\mathrm{rel}}} \mathrm{ramp}(V_{\mathrm{dsat}} + 5\mathrm{mV} - V_{\mathrm{ds}})$. We also use the following notation: $\mathrm{V(x)}$ denotes the potential at node x, $\mathrm{V(x,y) = V(x) - V(y)}$, $\mathrm{V_{dif}(x,y) = (V(x) - V(y))/2}$ and $\mathrm{V_{com}(x,y) = (V(x) + V(y))/2}$.

In normal operation the pd signal is kept low. An external current source pulls $I_{\mathrm{bias}} = 16\mu\mathrm{A}$ from the $bias$ input. V_{dda} is set to 5V and V_{ssa} to 0V. The agnd input voltage must be in the middle between V_{dda} and V_{ssa} (analog reference level) and the cmf input should be kept at $(v(\mathrm{outp}) + v(\mathrm{outn}))/2$. Refer to Fig. 2 for the test circuit.

Dimensions of mutually dependent transistors are altered in parallel (like in e.g. [7]). These groups are: M_3...M_{11}, M_{12}...M_{22}, M_{23}...M_{24}, M_{25}... M_{26}, M_{27}...M_{28}, and M_{29}...M_{30}. The width of M_{24} is twice the width of M_{25}. Widths of M_3 and M_{12} are adjusted with regard to the width of M_4 and M_{13}. For the optimization the same values for M were used as in Fig. 1.

The common mode offset voltage is defined as $\mathrm{V_{com}(outp, outn)}$ at $\mathrm{V_{dif}(inp, inn)} = 0\mathrm{V}$ and $\mathrm{V_{com}(inp, inn)} = 0\mathrm{V}$. The linear range is defined as the percentage of the maximal output voltage range $[V_{\mathrm{ssa}} - V_{\mathrm{dda}}, V_{\mathrm{dda}} - V_{\mathrm{ssa}}]$ where the differential amplification is above $1/2$ of its maximum value. The common mode range (CMR) is the span of $\mathrm{V_{com}(inp, inn)}$ (measured at $\mathrm{V_{dif}(inp, inn)} = 0$) where $V_{\mathrm{ds}} - V_{\mathrm{dsat}} > 0$ for all transistors in $\mathcal{M}_{\mathrm{rel}}$. In the AC analysis (transfer function from $\mathrm{V(inp, inn)}$ to $\mathrm{V(outp, outn)}$) the gain at 0Hz, phase margin (difference to 180° at 0dB gain) and the frequency where the gain falls to 0dB are measured. NOISE analysis is performed with output at $\mathrm{V(outp, outn)}$ and input at voltage source $\mathrm{V_{dif}}$. Input noise spectrum density is measured at two frequencies: 10Hz (n_1) and 1kHz (n_2). The measure of the amplifier area is defined as the sum of WL products for all transistors in $\mathcal{M}_{\mathrm{rel}}$.

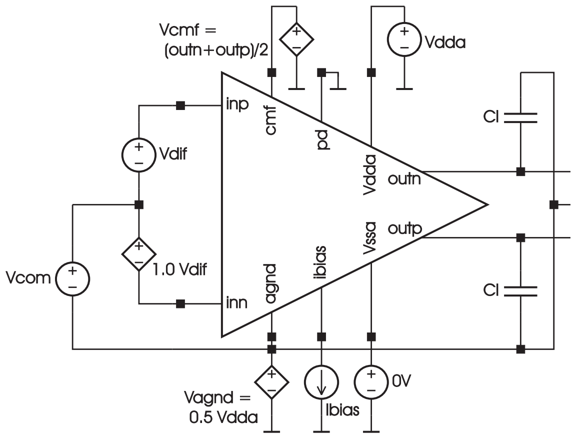

Figure 2. Test setup for the circuit in Fig. 1.

The following corner values/models are examined: TM, WO, WZ, Wp, and WS for the CMOS process, 25 °C, −40 °C, and 125 °C for the temperature, 5V, 4.5V, and 5.5V for V_{dda}, $16\mu\mathrm{A}$, $13.6\mu\mathrm{A}$, and $18.4\mu\mathrm{A}$ for I_{bias}, and 6pF, 4.2pF, and 7.8pF for the load capacitance. The design requirements and tradeoffs are listed in Table 1.

Characteristic	req.	A	T
Sat. measure	≤ 0V	1μV	∞
CM offs. v.	≤ 50mV	1mV	2mV
Linear range	$\geq 73\%$	0.1%	1%
CMR (low)	≤ -1.2V	1mV	100mV
CMR (high)	≥ 1.2V	1mV	100mV
0Hz gain	≥ 60dB	1dB	0.5dB
Phase margin	$\geq 50°$	1°	1°
0dB frequency	≥ 7.0MHz	0.1MHz	0.2MHz
n_1 (*)	≤ 620	100	50
n_2 (*)	≤ 62	10	5
Area ($\mu\mathrm{m}^2$)	≤ 8300	100	200

Table 1. Design requirements and tradeoffs. (*) Noise spectrum density unit is $\mathrm{nV}/\mathrm{Hz}^{1/2}$.

For all optimization runs $C = 10^{-6}$ was used. Additional penalty terms were introduced in case of a failure: in the OP analysis a penalty of 10^6 was added and the remaining analyses were skipped, in the differential mode DC sweep the linear range was set to 0%, in the common mode DC sweep analysis the lower (upper) bound of the common mode range was set to $+5$V (-5V), in the AC analysis 0Hz gain, phase margin and 0dB frequency were set to 0, and in the NOISE analysis n_1 (n_2) were set to $10^{-4}\mathrm{V}/\mathrm{Hz}^{1/2}$ ($10^{-5}\mathrm{V}/\mathrm{Hz}^{1/2}$). In case any of these failures occured in the first part of the heuristic search, the second part was skippen with additional penalty of 10^9. In case of a failure in the OP analysis when the remaining analyses were skipped for a particular corner, the skipped analyses were treated as failed.

SPICE was used as the circuit simulator [14]. The optimization method [13] was a modified constrained simplex method based on [3]. Three optimization runs were executed. In every run 12 parameters were optimized (5 widths, 5 lengths and 2 width ratios). In the first run the bounds on optimized parameters spanned a range 5 times above an below the values from the reference circuit. In case any lower bound was below 0.6μm it was truncated to that value. W_3/W_4 and W_{12}/W_{13} ratios were constrained to $[0.03, 0.75]$ and $[0.025, 0.625]$ respectively. In the second and third run lengths were constrained to $[0.6\mu\mathrm{m}, 1000\mu\mathrm{m}]$ and widths to $[0.6\mu\mathrm{m}, 3\mu\mathrm{m}]$. W_3/W_4 and W_{12}/W_{13} ratios were constrained to $[0.01, 1.00]$. In the first run the computer tried to improve the reference design, which was used as the initial point for the optimization. In the second and third run the computer started with a design that didn't work (all widths were 20μm, lengths 2μm, and W_3/W_4 and W_{12}/W_{13} ratios were 0.2). In the first and third run the optimization was stopped as soon as the relative simplex size became smaller than 0.001. In the second run the optimization was stopped as soon as some circuit with cost function value 0 was found.

The results are listed in Table 2. In the nominal corner the common mode offset voltage obtained in the first and second run was worse than in the reference circuit. The linear range from the first run, 0Hz gain from the second and third run, and the upper bound of the common mode range and noise from all three runs were slightly worse than in the reference circuit.

In the respective worst corners the common mode offset voltage from the second run, linear range from the first run, upper bound of the common mode range from the second and third run, and the noise from the first run were slightly worse than in the reference circuit.

All other performance measures were better in the computer designed circuits than in the reference circuit, furthermore all performance mesures for the obtained circuits were within the intervals prescribed in the design requirements. This implies that the obtained design is robust.

A significant decrease of the circuit area was observed. Even in the second run, where tradeoffs were not applied, a 5% decrease compared to the reference circuit's area was observed. When optimizing the reference design (first run), a 30% decrease was obtained. **When the optimizer had no initial design (third run), the obtained circuit's area was more than 50% lower than the area of the reference circuit.**

4 Conclusions

The IC design methodology applied by IC designers in their everyday work has been mathematically formulated. The resulting automated design method utilizes a cost function which is optimized by a direct search optimization method, avoiding the calculation of sensitivities. The cost function formulation requires from the designer to specify the design requirements in the form of upper/lower bounds on performance mesures, the design requirement violation penalty rates, and the performance tradeoff rates. To decrease the number of optimized parameters, groups of transistors with mutually dependent dimensions and M parameters must also be specified. Finally the optimizer requires a set of independent circuit parameters (dimensions) and a set of corresponding intervals of their possible values.

The method was demonstrated on robust differential amplifier design. All three runs resulted in an overall better circuit when compared to the reference circuit's performance. All runs were conducted on an 450MHz Intel Pentium III computer with 128MB of RAM. The longest run took 57 hours. By using a cluster of 4-5 state-of-the-art computers (approx. 5 times faster) this figure could be cut down to 4-6 hours per run.

There remain several possible applications of the method

Characteristic	Nominal corner				Respective worst corner			
	Ref. circuit	1^{st} run	2^{nd} run	3^{rd} run	Ref. circuit	1^{st} run	2^{nd} run	3^{rd} run
CM offset	5.5mV	16.1mV	34.4mV	0.736μV	32.7mV	19.0mV	38.9mV	12.8μV
Linear range	79.4%	78.0%	83.1%	79.8%	74.0%	73.0%	78.3%	74.8%
CMR (low)	-1.40V	-1.45V	-1.45V	-1.40V	-1.15V	-1.20V	-1.20V	-1.20V
CMR (high)	3.95V	3.80V	3.85V	3.80V	3.45V	3.50V	3.35V	3.25V
0Hz gain	74.0dB	79.3dB	73.2dB	72.2dB	61.6dB	78.4dB	72.0dB	70.7dB
Phase margin	62.8°	68.3°	68.5°	71.5°	56.2°	61.9°	62.0°	65.0°
0dB freq.	13.1MHz	16.5MHz	17.0MHz	17.6MHz	8.23MHz	10.1MHz	10.2MHz	10.6MHz
n_1 (*)	386	478	443	444	599	614	571	570
n_2 (*)	39.3	48.3	44.8	44.8	60.8	62.0	57.8	57.5
Area (μm^2)	8240	5853	7810	3825	-	-	-	-

Table 2. Nominal and worst case performance and area. (*) Noise spectrum density unit is $\mathrm{nV}/\mathrm{Hz}^{1/2}$.

to be examined in the course of future research: automated low power design, technology migration [16], circuit synthesys, etc. Especially technology migration is an area, where a lot of designer's time can be saved by utilizing an automated approach. The presented method can easily be applied to such problems.

References

[1] H. Abdel-Malek and A. Hassan. The ellipsoidal technique for design centering and region approximation. *IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems*, 10(8):1006–1014, August 1991.

[2] S. A. Aftab and M. A. Styblinski. IC variability minimization using a new Cp and Cpk based variability/performance measure. In *Proceedings of IEEE International Symposium on Circuits and Systems - ISCAS '94*, pages 149–152, London, UK, 1994.

[3] M. J. Box. A new method of constrained optimization and a comparison with other methods. *Computer Journal*, 8(1):42–52, April 1965.

[4] M. del Mar Hershenson, S. P. Boyd, and T. H. Lee. Optimal design of a CMOS op-amp via geometric programming. *IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems*, 20(1):1–21, January 2001.

[5] A. Dharchoudhury and S. M. Kang. Worst-case analysis and optimization of VLSI circuit performances. *IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems*, 14(4):481–492, April 1995.

[6] G. G. E. Gielen and R. A. Rutenbar. Computer-aided design of analog and mixed-signal integrated circuits. *Proceedings of the IEEE*, 88(12):1825–1854, December 2000.

[7] H. Gräb, S. Zizala, J. Eckmüller, and K. Antreich. The sizing rules method for analog integrated circuit design. In *IEEE/ACM International Conference on Computer Aided Design. ICCAD 2001*, pages 343–349, San Jose, CA, USA, 2001.

[8] P. R. Gray, P. J. Hurst, S. H. Lewis, and R. G. Meyer. *Analysis and design of analog integrated circuits*. John Wiley & Sons, New York, USA, 2001.

[9] M. Keramat and R. Kielbasa. OPTOMEGA: an environment for analog circuit optimization. In *ISCAS '98 Proceedings of the 1998 IEEE International Symposium on Circuits and Systems*, pages 122–125, Monterey, CA, USA, 1998.

[10] M. Krasnicki, R. Phelps, R. A. Rutenbar, and L. R. Carley. MAELSTROM: efficient simulation-based synthesis for custom analog cells. In *Proceedings 1999 Design Automation Conference*, pages 945–950, New Orleans, LA, USA, 1999.

[11] K. Krishna and S. W. Director. The linearized performance penalty (LPP) method for optimization of parametric yield and its reliability. *IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems*, 14(12):1557–1568, December 1995.

[12] R. Phelps, M. Krasnicki, R. A. Rutenbar, L. R. Carley, and J. R. Hellums. Anaconda: simulation-based synthesis of analog circuits via stochastic pattern search. *IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems*, 19(6):703–717, 2000.

[13] J. Puhan, T.Tuma, and I. Fajfar. Optimisation methods in SPICE: a comparison. In *Proceedings of European Conference on Circuit Theory and Design. ECCTD'99. Vol. 2*, pages 1279–1282, Stresa, Italy, 1999.

[14] T. Quarles, A. R. Newton, D. O. Pederson, and A. Sangiovanni-Vincentelli. *SPICE3 version 3f4 user's manual*. University of California, Berkeley, CA, USA, 1989.

[15] R. Schwencker, F. Schenkel, H. Gräb, and K. Antreich. The generalized boundary curve - a common method for automatic nominal design and centering of analog circuits. In *Proceedings Design, Automation and Test in Europe Conference and Exhibition 2000*, pages 42–47, Los Alamitos, CA, USA, 2000.

[16] A. H. Shah. Technology migration of a high performance CMOS amplifier using an automated front-to-back analog design flow. In *Design, Automation and Test in Europe Conference and Exhibition, 2002. Proceedings of Designers' Forum*, pages 224–229, Paris, France, 2002.