Optimizing Current Conveyors by Evolutionary Algorithms Including Differential Evolution

I. Guerra-Gómez, E. Tlelo-Cuautle
INAOE, Department of Electronics
Luis Enrique Erro No. 1.
Tonantzintla, Puebla. 72840 MEXICO
Email: ivickguerra@yahoo.com.mx

T. McConaghy
Solido Design Automation Inc.
101 - 116 Research Drive
Saskatoon, SK, Canada

G. Gielen
Katholieke Universiteit Leuven
ESAT-MICAS
Kasteelpark Arenberg 10
B-3001 Heverlee - Belgium

Abstract—An optimization system based on two multi-objective evolutionary algorithms NSGA-II and MOEA/D is presented for sizing analog integrated circuits. The proposed system uses HSPICE as circuit evaluator and its usefulness is highlighted by sizing mixed-mode analog circuits composed of unity-gain cells. That way, the circuits are optimized in voltage- and current-mode in three parameters: gain, bandwidth and offset, and by including as constraint that all transistors are in saturation operation. Comparison of performances between NSGA-II and MOEA/D with and without differential evolution (DE) as genetic operator in the optimization of two current conveyors are given by using standard CMOS integrated circuit technology.

II. MULTI-OBJECTIVE OPTIMIZATION

NSGA-II and MOEA/D works with multi-objective optimization by minimizing (or maximizing) a problem of the form [10]:

\[
\text{minimize} \quad F(x) = (f_1(x), f_2(x), \ldots, f_m(x))^T \\
\text{subject to} \quad h_k(x) \geq 0, \quad k = 1 \ldots p,
\]

where \( X \subset \mathbb{R}^n \) is the decision space for the variables, \( x = (x_1, \ldots, x_n) \) is called the decision vector and \( n \) is the number of variables. \( F(x) \) is the objective vector, \( f_j(x) : \mathbb{R}^n \rightarrow \mathbb{R}, \quad j = 1 \ldots m (m \geq 2) \) are objective functions and \( h_k(x), k = 1 \ldots p \) are performance constraints. Very often, since the objectives in (1) contradict each other, no point in \( X \) minimizes all the objectives simultaneously. The best tradeoffs among the objectives can be defined in terms of Pareto optimality [4].

A. NSGA-II Algorithm

This is an algorithm based on Pareto ranking. First it is necessary to build two populations \((P_0, Q_0)\) each one of size \( N \). The NSGA-II procedure in each generation consists of rebuilding the current \( t \) population \((R_t)\) from the two original populations \((P_t, Q_t)\).

Next, through a nondominated sorting procedure all solutions in \( R_t \) are ranked and classified in a family of sub-fronts [2]. In the next step, it is necessary to create from the current population \( R_t \) (previously ranked and ordered by sub-front number) a new offspring \((P_{t+1})\), the objective will be to choose from a population of size \( 2N, N \) solutions which belong to the first sub-fronts. In this manner, the last sub-front could be greater than necessary, then a measure \((\text{distance})\) preserving diversity is used by selecting the solutions that are far from the rest.

B. MOEA/D

The basic idea of MOEA/D is the decomposition of a multiobjective problem in scalar optimization subproblems by a weights vector [5]. This vector associates a weight \((\lambda)\) for each subproblem that is considered as a single solution in the
population and is going to try to improve by itself and to its nearby (neighborhoods).

In each generation there is a population of $N$ solutions $x^1, x^2, \ldots, x^n \in X$ where $x^i = (x^i_1, \ldots, x^i_n)$ is the current solution to the $i_{th}$ subproblem. After the initialization of the parameters the first step in MOEA/D is related to define the $N$ spread weights vector (to each solution corresponds one $\lambda_i$). Therefore, it is possible to define a number ($T$) of neighborhoods for each $\lambda_i$. One way can be by using a parameter $H$ in a sequence, as described by $\{0/H, 1/H, \ldots, H/H\}$.

In the procedure it is necessary to generate a new solution $y$ which will be compared with all its neighborhood by applying a decomposition approach ($g(x^i | \lambda_i, z^*_{j})$) such as the Tchebycheff Approach and each neighbor worse than this new solution will be replaced by it in an external population (EP) which is used to store non-dominated solutions.

In the Tchebycheff Approach, the scalar optimization problem is described by $g(x^i | \lambda_i, z^*) = \max \{\lambda_i | f(x^i) - z^*_j\}$, where $1 \leq i \leq N$. $1 \leq j \leq m$ and $z^* = [z^*_1, z^*_2, \ldots, z^*_m]^T$ are the best current objective functions found [10].

C. Differential Evolution

Differential evolution (DE) consists of randomly choosing three solutions $[8]; x^a, x^b$ and $x^c$ from $\{x^1, x^2, \ldots, x^N\}$. A new solution $x_{new} = x^a_{new}, x^b_{new}, \ldots, x^c_{new}$ is generated as:

$$x_{i_{new}} = x_i^a + R \cdot (x_i^b - x_i^c), \quad i = 1, 2, \ldots, n.$$

where $R$ is a constant factor which controls the amplification of the differential variation, in this work it is $R = 0.5$.

III. PROPOSED SYSTEM

The proposed optimization system works on a MATLAB code and the circuit simulations are made with HSPICE by modifying each transistor width ($W$), and recollecting results from the output listing. The evaluations are made by using the .MEAS statement which prints user-defined electrical specifications of a circuit, and the results could be manipulated in a post-processing step [11]. The library shown in Table I is included to perform AC and DC analysis. With the aim to compare the behavior of both optimization methods, two genetic operators are used: one-point cross-over (CR) and DE.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Transistor</th>
</tr>
</thead>
<tbody>
<tr>
<td>W1</td>
<td>$M1, \ldots, M4$</td>
</tr>
<tr>
<td>W2</td>
<td>$MB1, \ldots, MB4$</td>
</tr>
<tr>
<td>W3</td>
<td>$M5, M6$</td>
</tr>
<tr>
<td>W4</td>
<td>$M7, M8$</td>
</tr>
<tr>
<td>W5</td>
<td>$M9$</td>
</tr>
</tbody>
</table>

V. RESULTS

The circuits are optimized in 6 runs (in a dual processor 2GHz, RAM 2GHz) with both EA’s methods and it was selected an H value of 11, then the population length for both methods is 78. The objectives to optimize are: gain, offset and bandwidth (BW) for voltage and current mode. Gain is the relation between the Y to X voltage transfer, and the X to Z current transfer. Offset is a voltage or current value between Y-X or X-Z, and BW is always expressed in Hertz. For voltage and current optimization of these circuits is desired a gain closer to unity, a minimum offset and a maximum BW.

Figures 2 - 5 depict the last non-dominated solutions for voltage and current optimization for the NSGA-II and MOEA/D approaches. As special case, Figure 2(c) compared with 2(d) shows that DE improves the convergence to a Pareto front. In similar cases are Fig. 3(c) compared with Fig. 3(d) and Fig. 5(d) compared with Fig. 5(e).
Tables III and IV show the results of CPU time per generation, gain, offset and BW results. For CCII+ Voltage optimization, NSGA-II(CR) and MOEA/D(DE) found the best values, but for the rest of the results always NSGA-II(DE) and MOEA/D(DE) achieve the best values while CPU time is preserved.

Figure 6 shows the CCII+ current optimization for BW by using MOEA/D along the 80 generations with CR and DE; it is possible to see how after applying DE all the runs converged at the same value. Figure 7 shows the same behavior, but this time with the CCII+ voltage optimization for Gain by using NSGA-II; after applying DE all the runs converged and in this case, the best value is reached in less generations with DE.

Table V displays the dominance percentage which shows how much the methods in columns are dominated by the methods in rows. For instance, NSGA-II(CR) is dominated 39.87% by NSGA-II(DE) for CCII+ current optimization. And MOEA/D(CR) is dominated 10% by MOEA/D(DE) for CCII+ voltage optimization. Generally NSGA-II(DE) has similar or large dominance percentage on NSGA-II(CR), and MOEA/D(DE) has similar or large dominance percentage on MOEA/D(CR).

VI. CONCLUSION

This work shows the performance of NSGA-II and MOEA/D by applying DE as genetic operator for the optimization of two current conveyors. Gain, band width and offset were the objectives optimized for voltage and current mode ensuring saturation condition.

The discussed results present evidence that in most cases DE improves solution for both EAs. On one hand, DE enhances the convergence to the Pareto front and reduces the objective scattering among different runs. On the other hand, DE preserves the same time efficiency and increases the dominance on NSGA-II and MOEA/D compared with CR.

In this manner we conclude in the usefulness of DE to improve circuit optimization due to the performances enhance-

VII. ACKNOWLEDGMENT

This work is partially supported by CONACyT/Mexico under the grants numbers 48396-Y and 81604-R.
REFERENCES


