Context-dependent transformation of Pareto-optimal performance fronts of operational amplifiers

Elisenda Roca, Manuel Velasco-Jiménez, Rafael Castro-López and Francisco V. Fernández

Institute of Microelectronics of Seville, IMSE-CNM, CSIC and University of Seville.

Avda. Americo Vespucio s/n, Isla de la Cartuja, 41092 Seville, Spain.

Phone: +34 954 46 66 66

FAX: +34 954 46 66 00

Corresponding Author:

Elisenda Roca, eli@imse-cnm.csic.es

Abstract:

The use of Pareto-optimal performance fronts in emerging design methodologies for analog integrated circuits is a keystone to overcome the limitations of traditional design methodologies. However, most techniques to generate the fronts reported so far neglect the effect that the surrounding circuitry (such as the load impedance) has on the Pareto-front, thereby making it only realistic for the context where the front was generated. This strongly limits the use of the Pareto front because of the strong dependence between the key performances of an analog circuit and its surrounding circuitry, but, more importantly, because this circuitry remains unknown until the Pareto-optimal front is being used. Since performance front generation is a costly process, this paper proposes that performance fronts for a new context of use of a given circuit can be obtained from fronts that were previously generated under some different conditions. Towards this goal, a transformation methodology for performance objectives of operational amplifiers has been developed. Experimental results for a folded-cascode and a Miller-compensated operational amplifiers show that this is a promising approach to reuse the fronts in multiple contexts.

Keywords: analog integrated circuits, design automation, circuit sizing, simulation-based optimization, multi-objective optimization

1 Introduction

Electronic design automation of analog integrated circuits still lags behind its digital counterpart. Beyond any single reason, the inherent complexity of designing the simplest of the analog systems (the many non-ideal effects, the larger sensitivity to noise, etc.) has hindered the evolution of systematic design methodologies.

Key tasks in any analog design automation flow are topology selection and topology sizing [1],[2]. Given a set of design specifications, the first task refers to the selection of an appropriate topology, among a set of functionally equivalent ones, that can meet the specifications. Once a topology has been selected, given as an interconnection of sub-blocks, the second task refers to the determination of sub-block specifications for which the higher-level specifications can be achieved. At the transistor level, topology sizing reduces to setting the device sizes and bias voltages and currents.

In the past couple of decades most approaches to analog sizing have been built around optimization-based approaches: some iterative optimization algorithm with a performance evaluator in the loop [3]-[9]. The sizing problem has usually been formulated as:

$$\begin{aligned} &\min_{x} \ \boldsymbol{f}(\boldsymbol{x}) \\ &\text{subject to } \boldsymbol{g}(\boldsymbol{x}) \geq 0 \qquad \boldsymbol{X}_L < \boldsymbol{x} < \boldsymbol{X}_H \end{aligned} \tag{1}$$

In this equation, the vector $\boldsymbol{f}(\boldsymbol{x})$ is the set of objective functions to be minimized. Vector $\boldsymbol{x}$ corresponds to the design variables, and $\boldsymbol{X}_L$ and $\boldsymbol{X}_H$ are their lower and upper bounds, respectively. The vector $\boldsymbol{g}(\boldsymbol{x}) \geq 0$ corresponds to the design constraints and delimits the feasible performance region.

If vector $\boldsymbol{f}(\boldsymbol{x})$ contains only one objective, the design problem can be solved by using any single-objective optimization algorithm, e.g., simulated annealing [10], genetic algorithms [11], differential evolution [12], etc. If vector $\boldsymbol{f}(\boldsymbol{x})$ contains more than one objective, there are usually multiple solutions of (1), each one corresponding to a different set of values of the objectives. In the majority of reported approaches, the sizing problem has been solved by applying a single-objective optimization algorithm. For the case of several objective functions, this entails the transformation of the multi-objective optimization problem into a single-objective optimization problem, usually by applying a weighted sum of the objective functions. This requires selecting a priori one solution among the multiple ones mentioned above.

More recently, a wave of innovation has arisen with the development and application of multi-objective optimization algorithms to this kind of problems. Several algorithms have been reported (e.g., NSGAII [13], SPEA2 [14], MOEA/D [15], etc.) and successfully applied to design automation problems [16]-[22]. The outcome of the application of these algorithms to problems with several mutually conflicting objectives is a Pareto-optimal performance front, which shows the best trade-offs among the objective functions. In a Pareto-optimal front, one performance objective cannot be improved without worsening at least one of the other objectives. Pareto fronts allow, for instance, the selection of a design solution a posteriori, once the complete set of solutions has been generated. Besides, new possibilities arise. For instance, they can be used as a resource for topology selection. Since the best trade-offs among solutions are available, it can be immediately known if a given topology can meet a given set of specifications. Performance fronts have been even proposed as a basic ingredient of hierarchical multi-objective bottom-up synthesis methodologies [21],[22].

Most popular algorithms for performance front generation are multi-objective evolutionary algorithms [23],[24], where a population of candidate solutions (typically in the order of some hundreds to a few thousands) evolves along a number of generations (typically in the order of some hundreds). Therefore, the optimization algorithm requires a number of executions of the performance evaluator in the order of thousands to some hundred thousands. High accuracy is usually wished and, hence, the performance evaluator is relatively costly: electrical simulation (e.g., HSPICE [25]) for cell-level sizing or detailed behavioral simulation at higher levels [26]. It is easy to conclude that the generation of a performance front is a costly process that may take several hours of computation time. The reason such a high computational time may be considered acceptable is that, unlike single-objective design problems, the outcome of multi-objective optimization algorithms can be used for a wide range of problems. However, in this reasoning a fundamental issue has been ignored in the literature. The objective functions cannot be selected arbitrarily. They must correspond to the typical performances that the designer is interested in, and the ultimate value of many commonly used performance characteristics of analog circuits do not only depend on the block itself, but, also, on its surrounding circuitry, i.e., the generated Pareto-front depends on the context where the analog building block is being used.

To address this context dependency of the fronts, a transformation procedure is proposed in this paper [1]. The paper is structured as follows. Basic concepts of multi-objective optimization and generation of Pareto-optimal performance fronts are introduced in Section 2. Section 3 details the practical problems that motivate this work and Section 4 proposes a transformation procedure that solves the previous problems. Section 5 illustrates this methodology with two examples, a folded-cascode operational amplifier and a Miller operational amplifier, and different number of objectives. Finally, conclusions are presented in Section 6.

2 Generation of Pareto-optimal performance fronts

As stated in equation (1), the sizing problem is formulated as the maximization or minimization of a set of *n* design objectives, $\boldsymbol{f}(\boldsymbol{x}) = \{ f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \cdots, f_n(\boldsymbol{x}) \}$, where $\boldsymbol{x}$ is the vector of design variables (e.g., device sizes, performance characteristics of sub-blocks at intermediate hierarchical levels), and each $f_i(\boldsymbol{x})$ is a performance characteristic of the block (such as dc gain), subject to some constraints (e.g., slew rate larger than a certain value for an operational amplifier).

In the single-objective case, design candidates, each one represented by a vector of design variables $\boldsymbol{x}$, can be easily ordered by considering the objective function value $f(\boldsymbol{x})$. However, in the multi-objective case, a different ordering method is required: *Pareto dominance*. A design point, $\boldsymbol{a} \in \boldsymbol{X}$, is said to *dominate* another design point, $\boldsymbol{b} \in \boldsymbol{X}$, (noted as $\boldsymbol{a} \prec \boldsymbol{b}$) if $\boldsymbol{f}(\boldsymbol{a}) \leq \boldsymbol{f}(\boldsymbol{b})$ and $f_i(\boldsymbol{a}) < f_i(\boldsymbol{b})$ for at least one function i [2]. The design point $\boldsymbol{a}$ is said to be *non-dominated* if there is no other design point that dominates it. The non-dominated set of the entire feasible search space is known as the *Pareto-optimal front*. All these concepts are illustrated in Figure 1 for a 2-dimensional objective space.

---

[1] A preliminary version of this paper appeared at the XI Int. Workshop on Symbolic and Numerical Methods, Modeling and Applications to Circuit Design [27].

[2] This formulation is valid for minimization problems. A simple change of sign applies for maximization.

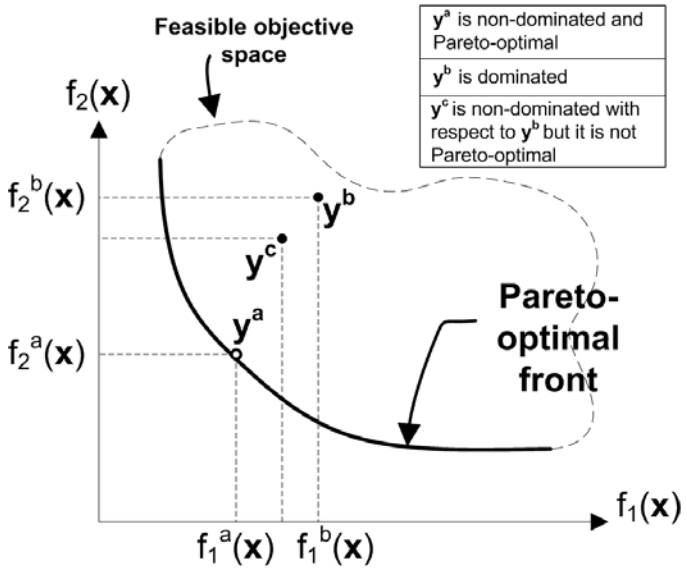

Figure 1. Illustrating Pareto dominance and Pareto-optimal front concepts for a 2-dimensional performance space.

The computation of an approximation to the ideal Pareto-optimal front is typically performed by using multi-objective evolutionary algorithms [23]. Ordering of solutions according to the Pareto dominance concept requires a performance evaluator, which assesses objective and constraint values for each design solution, typically named individual in the literature on evolutionary algorithms. These multi-objective evolutionary algorithms typically start with a random population of individuals that, after being evaluated by the performance evaluator, is modified in such a way that after *n* iterations (called generations) a population of non-dominated individuals is obtained: the Pareto-optimal front. Being of stochastic nature, the computational cost due to the high number of required performance evaluations is the main drawback of these algorithms. Efficiency, convergence to the ideal Pareto-optimal front, and diversity of solutions are areas of intense and current research. An illustrative example of recent research in these areas is the definition of quality metrics that, properly tracked, can control when the optimization algorithm should be stopped [28].

In order to illustrate the potential of Pareto-optimal fronts, let us consider the folded-cascode operational amplifier of Figure 2, where a capacitive load of 1pF has been considered at the output. The circuit is optimized to obtain the maximum values of two conflicting objectives: the dc gain, A_0, and the unity-gain frequency, f_u. The design variables are the width and length of the transistors, as well as the bias currents. Different constraints are imposed in order to obtain correct and useful sized circuits; for example, dc gain is set to be larger than 20dB, phase margin is set to lie between 10^o and 90^o, and transistors are enforced to operate in the saturation region. The Pareto-optimal front was generated by coupling the electrical simulator HSPICE to the multi-objective evolutionary optimization algorithm NSGAII. The population size and number of generations were set at 1500 and 100, respectively. The generation of this Pareto-optimal performance front took 1 hour of CPU time on a 2.2-GHz processor. The result is a 1500-points approximation to the 2-dimensional Pareto front, which can be seen as black points in Figure 3. Each point represents a sized circuit showing a best trade-off among the two performances being considered. To illustrate the evolution of the population of design solutions towards the Pareto-optimal front, Figure 3 shows the population after 8 generations (red points), 25 generations (blue points) and 100 generations (black points).

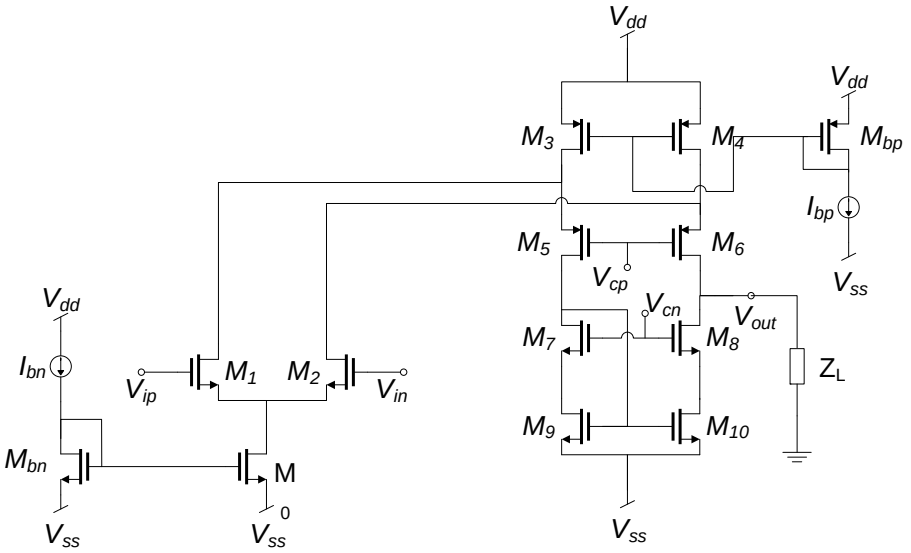

Figure 2. Folded-cascode amplifier.

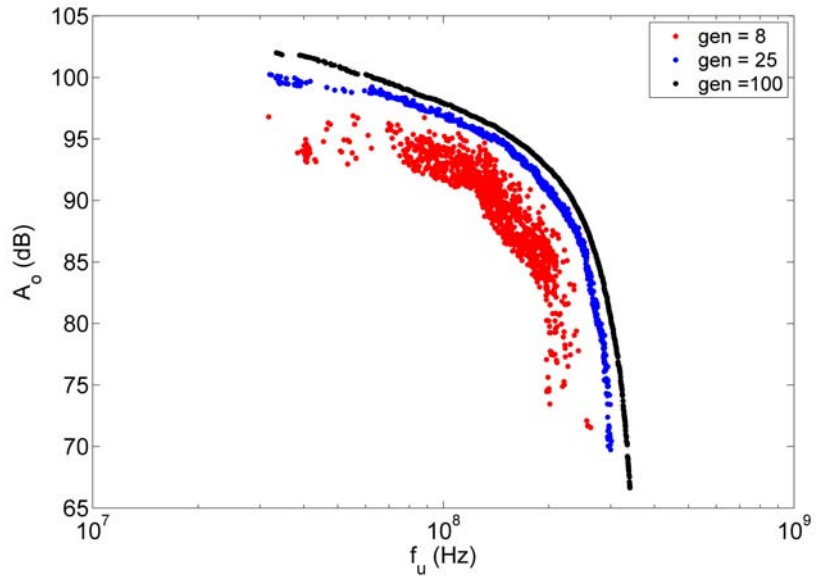

Figure 3. Illustrating the Pareto-optimal front generation for the folded-cascode amplifier of Figure 2 where dc gain and unity gain frequency have been maximized.

3 Potential problems in using performance fronts

As discussed in previous sections, the generation of Pareto-optimal performance fronts may be a computationally expensive process. Yet, their great potential benefit is that multiple performance trade-offs are available. Either if they are used interactively or within some design automation flow, once they are generated they can be reused many times by selecting the design solution with the appropriate performance trade-off in each case. However, there is a fundamental flaw in this reasoning: some circuit performances do not depend only on the circuit itself but also on the other circuitry to which it is connected. Consider, for instance, the load impedance that a typical analog block such as the Miller-compensated operational amplifier in Figure 4 has to drive. Although,

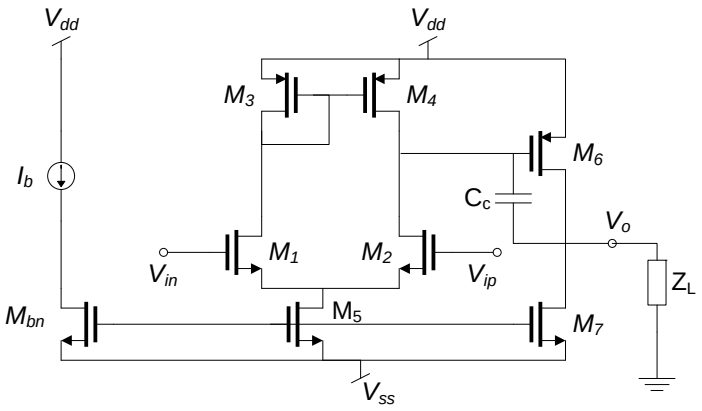

Figure 4 Miller two-stage operational amplifier.

usually, a higher number of performance objectives has to be considered, for illustration's sake we will consider only two: dc gain and unity-gain frequency. The plot with black circles in Figure 5 shows the performance front of the Miller opamp exhibiting the dc gain vs. unity-gain frequency trade-off for a phase margin constraint PM>60º and a capacitive load of 1pF. If the load impedance is changed to the parallel connection of a 2-pF capacitance and a 10-kΩ resistance, the performance front obtained is the one shown in the same figure with grey circles. Therefore, it becomes clear that the information that the performance front offers about the trade-offs that a topology can achieve, depends on the context where the circuit is going to be used. The conclusions drawn from the performance front obtained for some given conditions cannot be thus directly extrapolated to some other conditions.

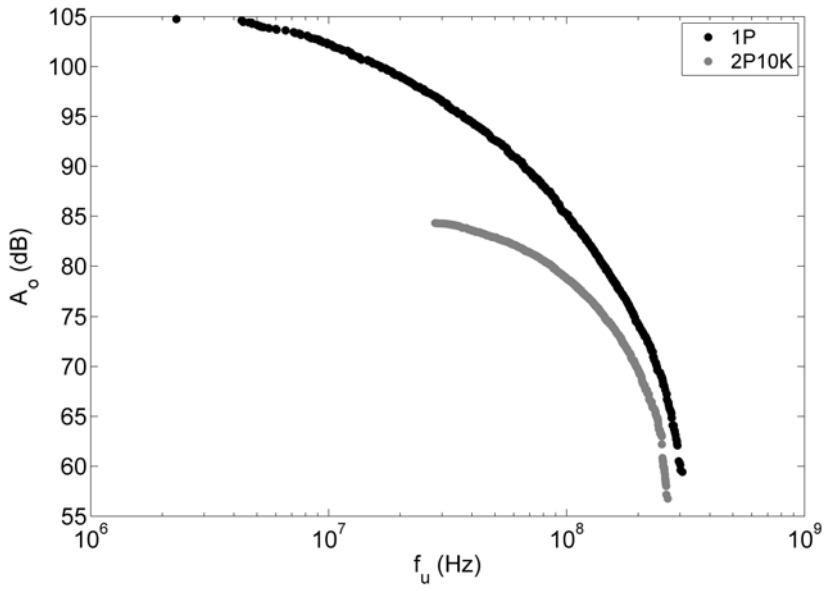

Figure 5. Dc gain vs. unity-gain frequency trade-off of the Miller operational amplifier for a load impedance equal to 1pF (black circles) and the parallel connection of a 2-pF capacitance and a 10-kΩ resistance.

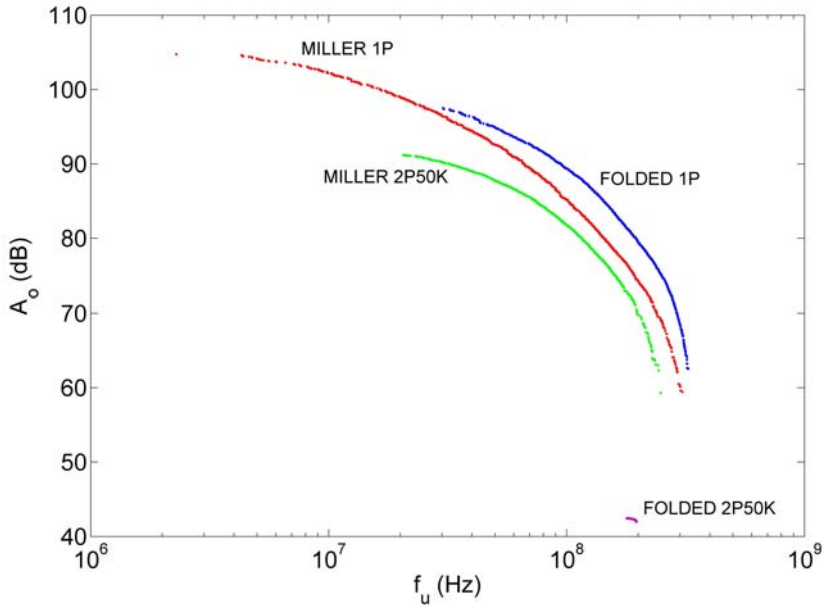

Figure 6. Dc gain vs. unity-gain frequency trade-off of the Miller and folded-cascode operational amplifiers for a load impedance equal to 1pF and the parallel connection of a 2-pF capacitance and a 10-kΩ resistance.

As discussed in Section 1, Pareto-optimal performance fronts of different topologies can be used to select the most appropriate architecture for some given specifications. For instance, Figure 6 shows the best dc gain vs unity-gain frequency trade-off that can be obtained for the Miller-compensated operational amplifier (red dots) in Figure 4 and the folded-cascode operational amplifier (blue dots) in Figure 2, respectively, when both are loaded with a 1-pF capacitance. According to these performance fronts, and assuming that only these two performances are of interest, there is a range of performance values for which a Miller operational amplifier would be preferable. A folded-cascode opamp, on the other hand, would be the preferred option for other performance values. Assume now that the amplifier load impedance is given by the parallel connection of a 2-pF capacitance and a 50-kΩ resistance. In this case, the Pareto-optimal performance front of the Miller amplifier is plotted with green dots and that of the folded-cascode amplifier is plotted in magenta dots. In this second case, all points of the folded-cascode amplifier front are dominated by points of the Miller amplifier front, hence, the Miller amplifier would be the preferred option whatever the required performances are. Therefore, these experiments exemplify that knowledge of the performance fronts at the right load conditions is also essential for proper selection of circuit topology.

4 Methodology for context-dependent transformation of Pareto-optimal performance fronts of operational amplifiers

Some of the performances usually considered in an optimization process of an operational amplifier depend on the load conditions, for instance dc gain A_0, unity-gain frequency f_u, or phase margin *PM*. However, the load conditions are not known until the design process is being performed, that is, until any other circuitry around the amplifier is known. Since the Pareto front generation process is a lengthy process, the objective of our research is to generate trade-off

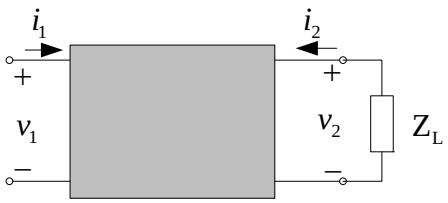

Figure 7. Two-port with arbitrary load.

information a priori and easily and efficiently transform this information into the Pareto fronts of the performances when the load conditions are known.

The operational amplifier can be viewed as a two-port, like the one shown in Figure 7. Voltage v_1 and current i_1 represent the differential input voltage and current, respectively. Voltage v_2 and current i_2 represent the output voltage and current, respectively. A priori, any two-port matrix characterization of the amplifier can be considered. However, the two-port matrix parameters must be selected intelligently, so that a load-independent characterization of the operational amplifier can be obtained more easily, according to the performances of interest of the block. Let us consider the hybrid-2 parameters [29] to characterize the two-port:

$$\begin{aligned} i_1 &= h'_{11} \cdot v_1 + h'_{12} \cdot i_2 \\ v_2 &= h'_{21} \cdot v_1 + h'_{22} \cdot i_2 \end{aligned} \tag{2}$$

In this equation, parameter h'_{11} represents the input impedance, h'_{12} is the inverse current gain, h'_{21} represents the voltage gain of the amplifier without any load and h'_{22} is the output impedance. The voltage gain and output impedance when a certain load Z_L is added can be obtained from (2) and the constitutive equation:

$$v_2 = -Z_L \cdot i_2 \tag{3}$$

yielding the following expressions:

$$\begin{aligned} A_v(s) &= \frac{h'_{21}(s)}{1 + \frac{h'_{22}(s)}{Z_L(s)}} \\ Z_o(s) &= h'_{22}(s) \end{aligned} \tag{4}$$

Equation (4) allows to obtain the hybrid-2 parameters h'_{21} and h'_{22} from the voltage gain, $A_v(s)$, and output impedance, $Z_o(s)$, for some known load conditions, and vice versa, to obtain the voltage gain $A_v(s)$ and output impedance $Z_o(s)$ for some load from the hybrid-2 parameters h'_{21} and h'_{22}. Moreover, a particular case is when $Z_L \to \infty$. In this case, h'_{21} is identical to $A_v(s)$. This equation is the key to developing the methodology that is proposed in this paper to transform a Pareto front for some known load conditions to arbitrary new load conditions.

Let us assume that we wish to generate the Pareto-optimal performance front for the output impedance, dc voltage gain, unity-gain frequency, and phase margin for some arbitrary load conditions. The generation methodology would proceed as follows:

1) Generate the Pareto-optimal front of the performances of interest for some known load conditions by using a multi-objective optimization algorithm with a nested electrical simulator as performance evaluator. The performances above (output impedance, gain, unity-gain frequency

and phase margin) can be easily obtained by processing the output of the AC circuit analysis performed by the electrical simulator.

2) For each sample or individual of this Pareto front, store pole and zero locations of the output impedance $Z_o(s)$ and the voltage gain $A_v(s)$, both being frequency dependent functions, as well as their dc values. This information can be retrieved from common electrical simulators.

3) Use (4) to extract the hybrid parameters $h'_{21}(s)$ and $h'_{22}(s)$ for each sample; this means extracting dc values and both, poles and zeros. Note that, from basic circuit theory, the poles of $h'_{21}(s)$ and $h'_{22}(s)$ are identical.

4) Apply (4) to obtain the voltage gain $A_v(s)$ for the new arbitrary load conditions (new $Z_L(s)$) from the previously calculated hybrid parameters.

5) Obtain the performance parameters (dc gain, unity-gain frequency and phase margin) by simple processing of the magnitude and phase of the network function $A_v(s)$.

Note that this procedure can be applied to any initial known load conditions.

The first step of this methodology has the heaviest computational effort by far, but note that the results of step 3 are independent of the application, i.e., independent of the final load conditions. Therefore, the first three steps can be performed beforehand, as only the circuit topology and the performances of interest are required. The results can be stored and used whenever and wherever necessary.

5 Experimental Results

5.1 Folded-cascode operational amplifier

The proposed methodology will be first applied to a single-stage operational amplifier, the folded-cascode amplifier in Figure 2. In this case, the optimization process will be carried out for three different performances: dc gain, A_0, unity-gain frequency, f_u, and phase margin, *PM*. According to step 1 of the methodology in Section 4, a Pareto-optimal front with a capacitive load of 1pF was first generated for the three objectives. The design variables in this case are the width and length of the transistors, as well as the bias currents. Several constraints are imposed in order to obtain valid circuits, while at the same time explore all the feasibility range of the performances (for example, dc gain is set to be larger than 20dB, and phase margin 90º >*PM* >10º). The Pareto front was generated by using the electrical simulator HSPICE as performance evaluator within the multi-objective evolutionary optimization algorithm NSGAII [3]. The population size and the number of generations were 1500 and 100, respectively. The generation of this Pareto front took 1 hour of CPU time on a 2.2 GHz processor. The result is 1500 sample points of the 3-dimensional Pareto front, which can be seen in Figure 8 (2 dimensional projections of this front are included in Figure

[3] Although all experimental results in this paper have been obtained by applying the multi-objective optimization algorithm NSGAII, it should be highlighted that the transformation procedure introduced in Section 4 is completely independent of the optimization algorithm used to obtain the initial Pareto-optimal front.

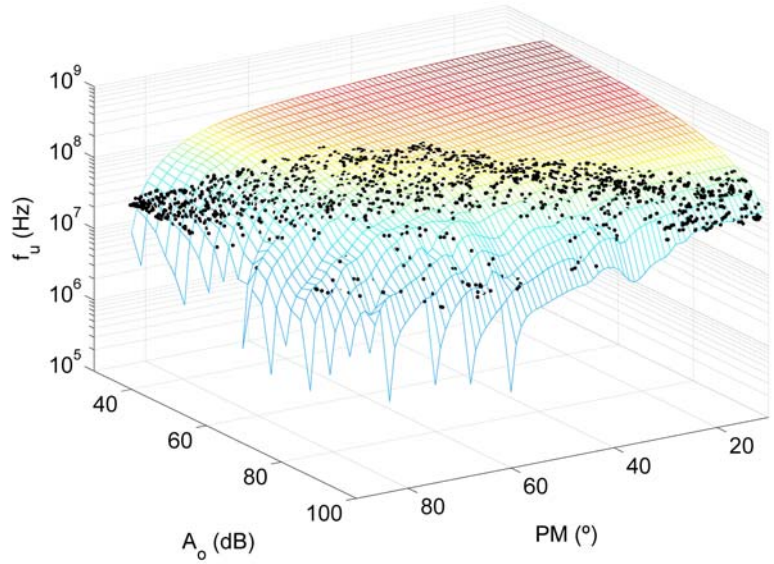

Figure 8 Pareto-optimal front of the folded-cascode amplifier with 1pF output load, generated for dc gain, unity gain frequency and phase margin.

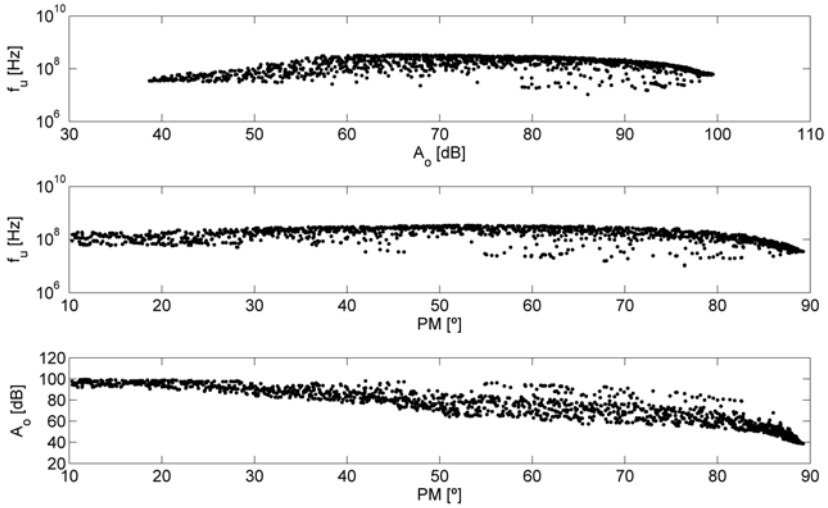

Figure 9 Projections of the POF generated for the folded-cascode amplifier with 1pF output load illustrated in Figure 8.

9). Each point represents a sized circuit showing one of the best trade-offs among the three performances considered. [4]

[4] The outcome of the optimization procedure is just the black points in Figure 8. The regression surface in these figures has just been added for better visualization in the three dimensional space, and no other use should be made of it. In particular, portions of this surface in regions with no points correspond to areas with infeasible performances.

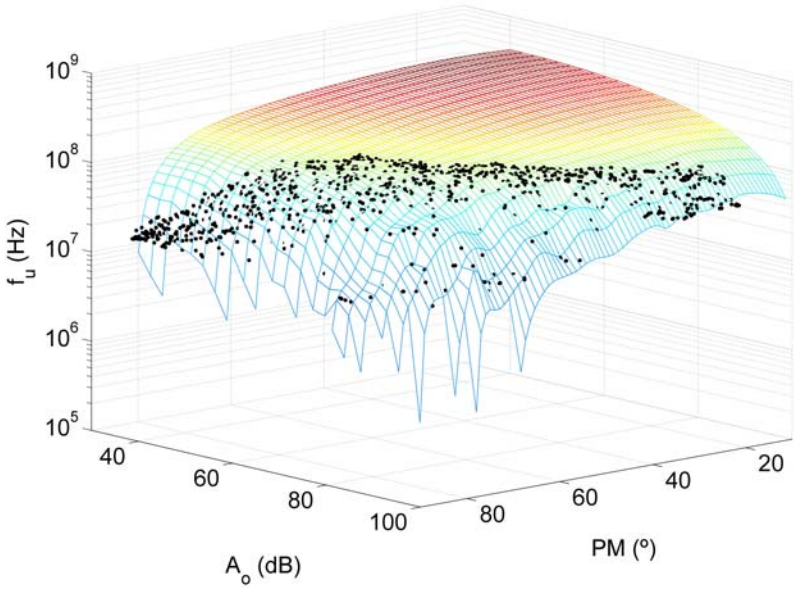

Figure 10 Pareto-optimal front of the folded-cascode amplifier obtained by transforming the Pareto front in Figure 8, generated for 1pF output load, to the new load conditions of 2pF.

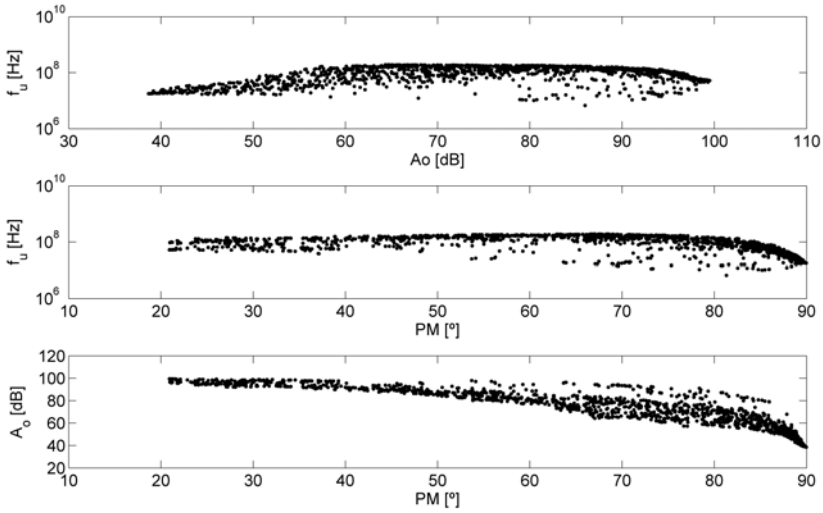

Figure 11 Projections of the transformed POF illustrated in Figure 10.

From the electrical simulation of the 1500 points of the Pareto-optimal front, the network functions $A_v(s)$ and $Z_o(s)$ of each design point can be easily obtained (step 2 in Section 4). According to step 3 in Section 4, eq. (4) is applied to each of these points, and therefore the hybrid-2 parameters h'_{21} and h'_{22} of the 1500 points can be calculated. These two steps are performed in less than 5 minutes of CPU time. All the steps performed so far are independent of the final load conditions. Therefore, although computationally costly, they can be performed long before the load is known and the other steps have to be carried out.

Assume that we now need to generate the Pareto front for a load of 2pF. Using the points previously stored, the Pareto front for the new load is generated by first calculating the new network functions $A_v(s)$ and $Z_o(s)$ from (4) and extracting dc gain, unity-gain frequency, phase margin and output impedance from their magnitude and phase behaviours (steps 4 and 5 described in Section 4). Figure 10 shows the 3-dimensional plot of the front and Figure 11 its 2-dimensional projections. The application of these two steps takes only 20 seconds of CPU time.

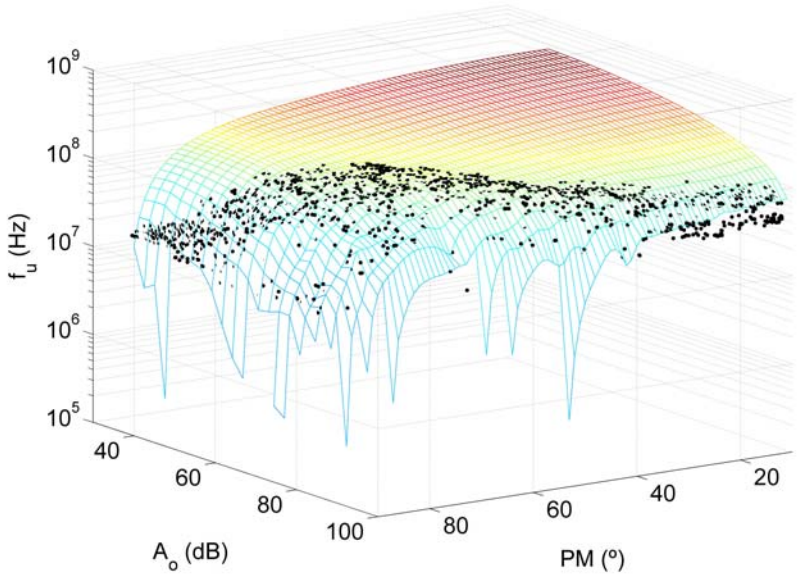

Figure 12 Pareto-optimal performance front generated by the multiobjective optimization algorithm for a folded-cascode amplifier with 2-pF output load.

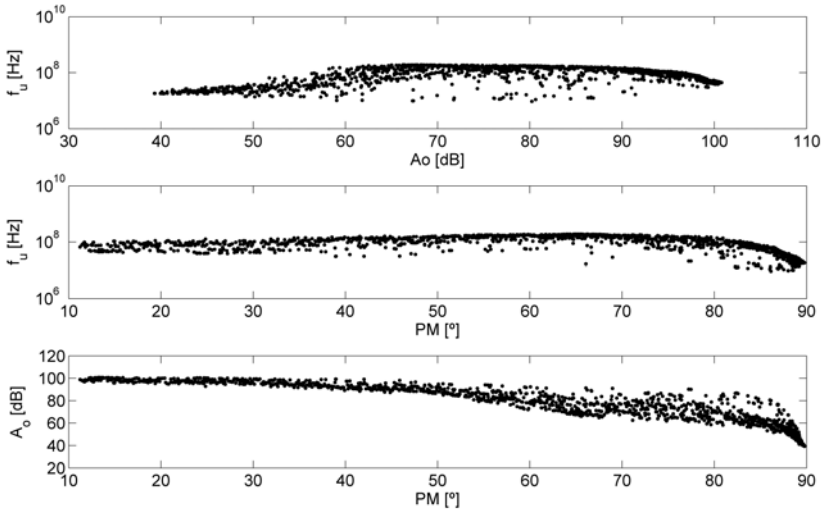

Figure 13 Projections of the POF generated for the folded-cascode amplifier with 2pF output load illustrated in Figure 12.

To assess the procedure, the Pareto front was also generated by running again the multi-objective optimizer with the electrical simulator for this new 2pF load (that is, not following the transformation procedure proposed here). A set of samples of the Pareto front with similar quality characteristics is obtained (see Figure 12 for the 3-dimensional POF and Figure 13 for 2-dimensional projections).

5.2 Miller-compensated two-stage operational amplifier

The same methodology has also been applied to the Miller operational amplifier depicted in Figure 4. The Pareto front was first generated with a capacitive load of 1pF for four objectives: dc voltage gain, A_0, unity-gain frequency, f_u, phase margin, *PM*, and output impedance, Z_o. The first three objectives were maximized and the fourth one was minimized. The design variables in this case are the width and length of the transistors, as well as the bias current and the compensation capacitor. As for the folded-cascode amplifier, different constraints are imposed to the problem: dc gain is set to be larger than 20dB, and phase margin $90^o > PM > 10^o$. Following the optimization procedure in Section 3, the Pareto front was generated by coupling the electrical simulator HSPICE to the multi-objective evolutionary optimization algorithm NSGAII. The population size was 1500 and the number of generations was 150. The generation of the Pareto front took 1 hour 36 minutes of CPU time. As in the previous experiment, most of the time was spent on the electrical simulation of all design solutions. Figure 14 shows the projections of the 1500 sample points of the 4-dimensional Pareto front on the phase margin vs. unity-gain-frequency plane, on the dc gain vs. unity-gain frequency plane, and on the dc gain vs. output impedance plane.

The network functions $A_v(s)$ and $Z_o(s)$ of each design point of the Pareto front are obtained from the circuit simulation, and the hybrid-2 parameters h'_{21} and h'_{22} of the 1500 points are calculated by applying (4) to each point. These two steps are performed in less than 5 minutes of CPU time. It should be remarked again that steps 1, 2 and 3 are independent of the final load conditions and they are performed before the load is known and the steps 4 and 5 have to be performed.

When the Pareto front for a new load condition is needed, for instance for 2pF and 50kΩ, the points previously stored with information for the hybrid-2 parameters are used to generate the new Pareto front by applying steps 4 and 5 of the transformation procedure. Figure 15 shows the three projections of the 4-dimensional front. It has to be observed again that the application of these two steps takes only 20 seconds of CPU time.

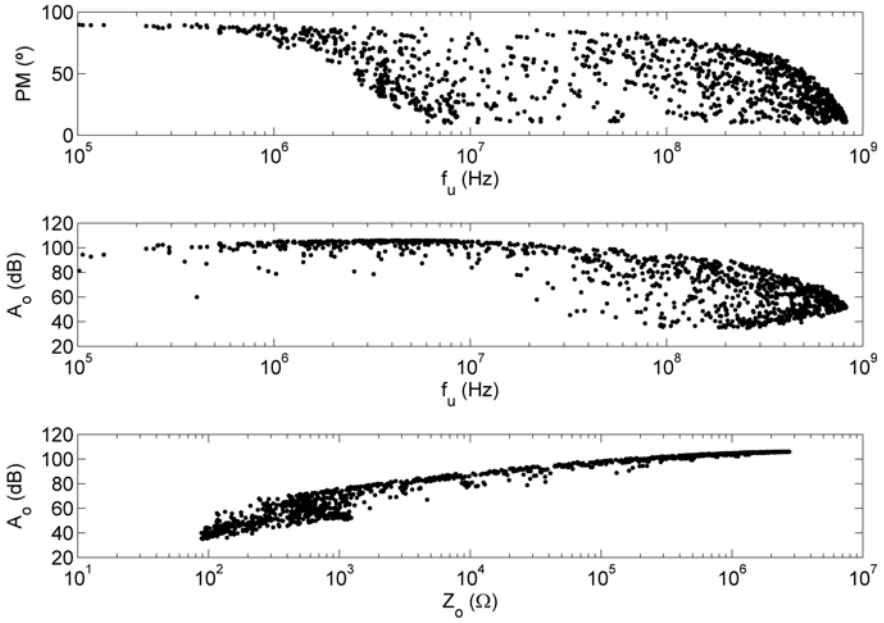

Figure 14 Projections of the POF generated by the multiobjective optimization algorithm for a capacitive load of 1pF.

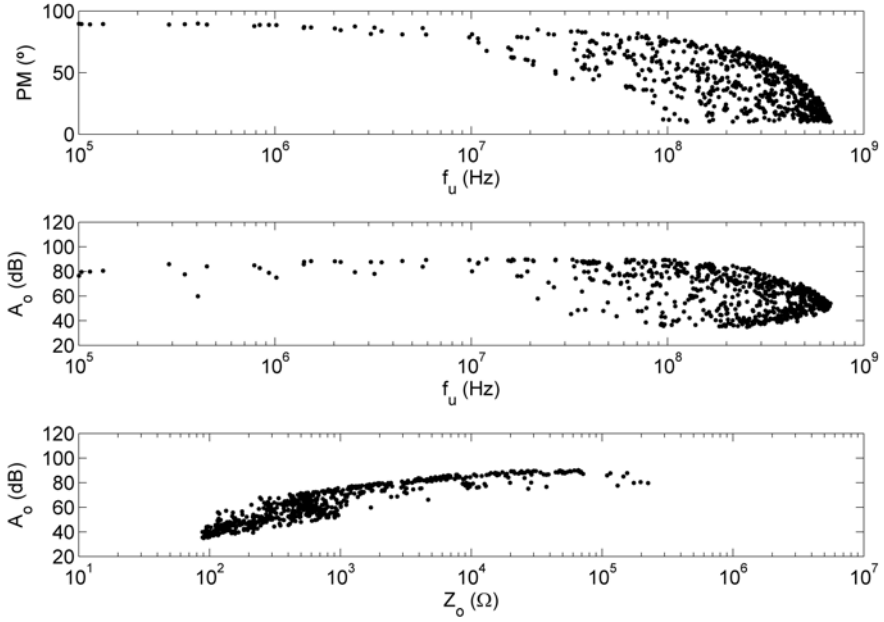

Figure 15 Projections of the POF obtained by the transformation procedure for a load of 2pF-50kΩ.

The Pareto front was also generated by running the optimizer with the electrical simulator for this new 2pF-50kΩ load in order to assess the procedure. A set of samples of the Pareto front with similar quality characteristics is obtained (see Figure 16). However, as in the generation of the initial Pareto front, 1 hour and 36 minutes of CPU is employed instead of the 20 seconds that the transformation procedure described here requires.

5.3 Discussion

From the two previous examples, we can see that the time to transform the Pareto-optimal performance front to the new load conditions, when the hybrid parameters are already calculated, is only 20 seconds for a front with 1500 points, i.e., less than 14 ms per point of the performance front. This is quite an acceptable time for interactive transformation (that is, during the design process) and use of the performance fronts. It is even acceptable for any hierarchical synthesis flow where iterative evaluations of circuit performances are necessary with different load conditions, i.e., at each iteration, the performances of the circuits of the Pareto front are transformed for the new load conditions.

The transformed Pareto fronts in Figure 10 and Figure 15 are similar to the fronts in Figure 12 and Figure 16, which have been generated by running again the complete optimization process for the final load conditions. It is obvious that the points of the transformed performance front will not coincide in general with those of the performance front obtained for the final load conditions. This is due to the fact that the samples are different because of the finite population size and the stochastic nature of the multi-objective evolutionary algorithm.

Additional differences can be observed as not all transformed points are useful, i.e., each design point changes its performances when the load conditions change. This change may shift the specifications to useless values, e.g., the phase margin of some design points may become negative

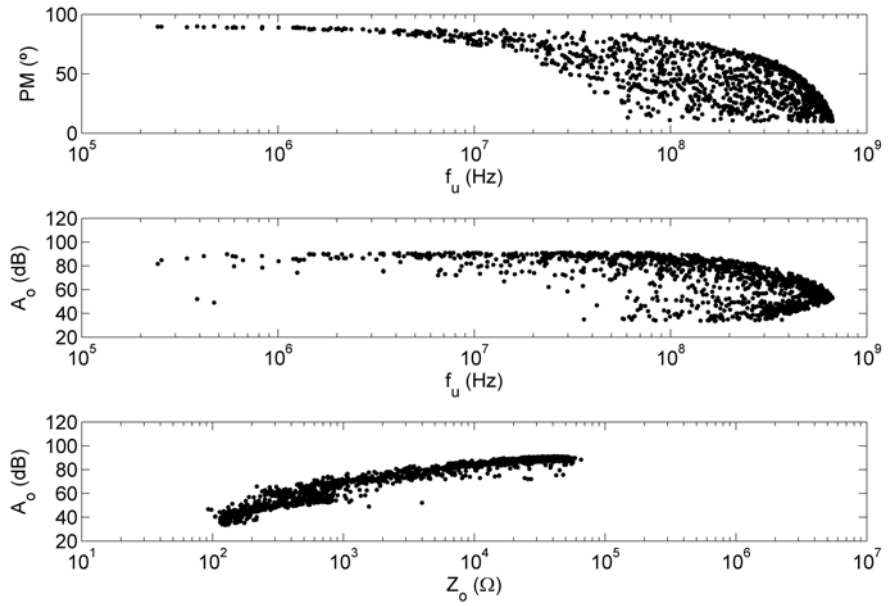

Figure 16 Projections of the POF generated by the multiobjective optimization algorithm for a load of 2pF-50kΩ.

or the dc gain may decrease too much. Therefore, the density of points in the transformed Pareto front is usually lower than in the original front.

Quality of Pareto-optimal performance fronts is usually associated to diversity (the range of performance objectives that is covered by the front and the uniform distribution of points in the front) and convergence (how close is the front to the ideal performance front). In terms of convergence, if the transformed front is compared to the performance front obtained for the final load conditions, some points are dominated by the latter and some points of the latter are dominated by those of the former. This is not strange as the performance fronts are an approximation to the ideal front (the evolutionary optimization algorithm generating the front stops after a finite number of generations). In fact, the same phenomenon can be observed if the fronts of two different executions of the optimization algorithm (of stochastic nature) for exactly the same conditions are compared.

There is some degradation in terms of diversity, usually more important as the difference between the initial and final load conditions becomes larger. This is quite logical as the transformation to new load conditions implies a displacement of the points of the Pareto front in the performance space. Multi-objective optimization algorithms are usually designed to optimize the diversity of points in the Pareto front. As the transformation procedure is non-linear with respect to the performance objectives, it is obvious that diversity will change.

6 Conclusions

This paper introduces a transformation procedure of Pareto-optimal performance fronts of operational amplifiers based on a hybrid-2 parameter characterization. This approach enables the efficient and rapid generation of performance fronts for arbitrary interconnection conditions.

Future work will address the generation of performance fronts for several initial load conditions to improve the diversity of the transformed performance fronts.

7 Acknowledgements

This research work has been supported by the TIC-2532 Project, funded by Consejería de Innovación, Ciencia y Empresa, Junta de Andalucía, and the TEC2010-14825 Project, funded by the Spanish Ministry of Science and Innovation and ERDF.

8 References

[1] Rutenbar, R., Gielen, G. & Roychowdhury, J. (2007). Hierarchical modeling, optimization, and synthesis of system-level analog and RF designs. *Proc. of the IEEE*, 95(3), 640-669.

[2] Martens, E. & Gielen, G. (2008). Classification of analog synthesis tools based on their architecture selection mechanisms. *Integration – the VLSI Journal*, 41(2), 238-252.

[3] Medeiro, F., Rodríguez-Macías, R., Fernández, F.V., Domínguez-Castro, R., Huertas J.L., & Rodríguez-Vázquez, A. (1994). Global design of analog cells using statistical optimization techniques. *Analog Integrated Circuits and Signal Processing*, 6(3), 179-195.

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

[5] Alpaydin, G., Balkir, S., & Dundar, G. (2003). An evolutionary approach to automatic synthesis of high-performance analog integrated circuits. *IEEE Trans. on Evolutionary Computation*, 7(3), 240-252.

[6] Barros, M., Guilherme, J., & Horta, N. (2010). Analog circuits optimization based on evolutionary computation techniques. *Integration – the VLSI Journal*, 43(1), 136-155.

[7] Sonmez, O., & Dundar, G. (2011). Simulation-based analog and RF circuit synthesis using a modified evolutionary strategies algorithm. *Integration – the VLSI Journal*, 44(2), 144-154.

[8] Castro-López, R., Guerra, O., Roca, E., & Fernández, F.V. (2008). An integrated layout-synthesis approach for analog ICs. *IEEE Trans. on Computer-Aided Design*, 27(7), 1179-1189.

[9] Liu, B., Fernández, F.V., Gielen, G., Castro-López, R., & Roca, E. (2009). A memetic approach to the automatic design of high-performance analog integrated circuits. *ACM Trans. on Design Automation of Electronic Systems*, 14(3).

[10] Rutenbar, R.A.(1989). Simulated annealing algorithms: an overview. *IEEE Circuits and Devices Magazine*, 5(1),19-26.

[11] Michalewicz, Z. (1996) *Genetic algorithms + data structures = evolution programs*. Springer.

[12] Price, K., Storn, R.,& Lampinen, J. (2005) *Differential evolution. A practical approach to global optimization*. Springer.

[13] Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). A fast and elitist multi-objective genetic algorithm: NSGA-II. *IEEE Trans. on Evolutionary Computation*, 6(2), 181-197.

[14] Zitzler, E., Laumanns, M., & Thiele, L. (2001). SPEA2: Improving the strength Pareto evolutionary algorithm. TIK-Report 103, Swiss Federal Institute of Technology.

[15] Zhang Q., & Li, H. (2007). MOEA/D: A multiobjective evolutionary algorithm based on decomposition. *IEEE Trans. on Evolutionary Computation*, 11(6), 712-731.

[16] De Smedt, B., & Gielen, G. (2003). WATSON: Design space boundary exploration and model generation for analog and RF IC design. *IEEE Trans. on Computer-Aided Design*, 22(2), 213-224.

[17] Chu, M., Allstot, D.J., Huard, J.M., & Wong, K.Y. (2004). NSGA-based parasitic-aware optimization of a 5GHz low-noise VCO. In *Proc. Asia and South Pacific Design Automation Conf.*, (pp. 169-174).

[18] Tiwary, S.K., Velu, S., Rutenbar, R.A., & Mukherjee, T. (2004). Pareto optimal modeling for efficient PLL optimization. In *Proc. NSTI Nanotechnology Conf.* (pp. 195-198).

[19] Stehr, G., Graeb, H.E., & Antreich, K.J. (2007). Analog performance space exploration by normal-boundary intersection and by Fourier–Motzkin elimination. *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, 26(10), 1733-1748.

[20] Castro-López, R., Roca, E. , & Fernández, F.V. (2009). Multimode pareto fronts for design of reconfigurable analogue circuits. *Electronics Letters*, 45(2), 95-96.

[21] Eeckelaert, T., McConaghy, T., & Gielen, G. (2005). Efficient multiobjective synthesis of analog circuits using hierarchical pareto-optimal performance hypersurfaces. In *Proceedings of the Design Automation and Test in Europe Conference (DATE'05) – VOLUME 2*, (pp. 1070-1075).

[22] Eeckelaert, T., Schoofs, R., Steyaert, M., Gielen, G., & Sansen, W. (2008). A continuous-time delta-sigma modulator for 802.11a/b/g WLAN implemented with a hierarchical bottom-up optimization methodology. *Analog Integrated Circuits and Signal Processing*, 55(1), 37-45.

[23] Deb, K. (2001). *Multi-objective Optimization Using Evolutionary Algorithms*. Wiley.

[24] Coello-Coello, C., Lamont, G.B., & Van Veldhuizen, D.A. (2007). *Evolutionary algorithms for solving multi-objective problems*. Second edition. Springer.

[25] HSPICE circuit simulator. http://www.synopsys.com/Tools/Verification/AMSVerification/CircuitSimulation/HSPICE/Pages/default.aspx. Accessed 15 February 2011.

[26] Ruiz-Amaya, J., de la Rosa, J.M., Fernández, F.V., Medeiro, F., del Río, R., Pérez-Verdú, B., & Rodríguez-Vázquez, A. (2005). High-Level Synthesis of Switched-Capacitor, Switched-Current and Continuous-Time ΣΔ Modulators Using SIMULINK-based Time-Domain Behavioral Models. *IEEE Trans. on Circuits and Systems, Part I*, 52(9), 1795-1810.

[27] Roca, E., Velasco-Jiménez, M., Castro-López, R., & Fernández, F.V. (2010). Context-independent performance modeling of operational amplifiers using Pareto fronts. In *Proc.of the XIth Int. Worskhop on Symbolic and Numerical Methods, Modeling and Applications to Circuit Design*, (pp.1-4).

[28] Fernández, F.V., Esteban-Muller, J., Roca, E., & Castro-López, R. (2010). Stopping criteria in evolutionary algorithms for multi-objective performance optimization of integrated inductors. In Proc. IEEE Int. Conf. on Evolutionary Computation.

[29] Chua, L.O., Desoer, C., Kuh, E.S. (1987). *Linear and Non-linear Circuits*. McGraw-Hill.