Single-Objective Front Optimization: Application to RF Circuit Design

E. J. Solteiro Pires
Univ. de Trás-os-Montes
e Alto Douro, Portugal
epires@utad.pt

Luís Mendes
Inst. Politécnico de Leiria
Inst. de Telec., Portugal
lmendes@estg.ipleiria.pt

P. B. de Moura Oliveira
Univ. de Trás-os-Montes
e Alto Douro, Portugal
oliveira@utad.pt

J. A. Tenreiro Machado
Inst. Superior de Engenharia
do Porto, Portugal
jtm@isep.ipp.pt

N. M. Fonseca Ferreira
Inst. Superior de Engenharia
de Coimbra, Portugal
fnunomig@isec.pt

João Vaz, Maria Rosário
Inst. Superior Técnico
Inst. de Telec., Portugal
{joaovaz,mrosario}@ist.utl.pt

ABSTRACT

This paper proposes a new algorithm which promotes well distributed non-dominated fronts in the parameters space when a single-objective function is optimized. This algorithm is based on ϵ-dominance concept and maxmin sorting scheme. Besides that, the paper also presents the results of the algorithm when it is used in the automated synthesis of optimum performance CMOS radio-frequency and microwave binary-weighted differential switched capacitor arrays (RFDSCAs). The genetic synthesis tool optimizes a fitness function which is based on the performance parameter of the RFDSCAs. To validate the proposed design methodology, a CMOS RFDSCA is synthesized, using a 0.25 μm BiCMOS technology.

Categories and Subject Descriptors

I.0 [**General**]: [Genetic Algorithms, Multi-Objective]

General Terms

Algorithms, Design

1. SINGLE-OBJECTIVE OPTIMIZATION

In multi-objective problems there are many approaches to find the Pareto front. One of them is based on ϵ-dominance concept [1]. This technique is used to get a set of solutions with good spread and diversity over the objective space. The proposed algorithm uses this concept and the maximin technique [2] as the main ideas to achieve good diversity in the parameter space. Initially, the objective space is divided in several ranks, being each one characterized with a ϵ-distance (Figure 1(a)). Inside each rank, all the solutions have the same preference, even when their objective values are different. In the next algorithm iteration, the best n distributed solutions are selected from a set with m solutions ($m > n$). Therefore, to select a population of pop_{dim} solutions, the algorithm begins by selecting the solutions with lower rank (rank 1 of Figure 1(a)) until the last allowed rank being considered has more solutions than the remaining slots in the new population. In this case we select the best dispersed solutions of that rank based on the maximin scheme.

The main concept behind the maximin sorting scheme is to select the solutions in order to decrease the large gap areas existing

GECCO'08, July 12–16, 2008, Atlanta, Georgia, USA.
ACM 978-1-60558-130-9/08/07.

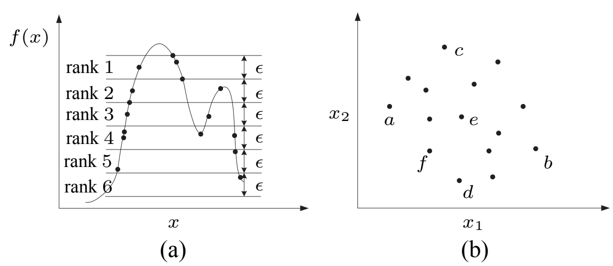


Figure 1: (a) Problem with one objective and one parameter $\{x\}$. (b) Solutions in a bidimensional parameter space $\{x_1, x_2\}$.

in the already selected population. For example, consider the solutions of one rank depicted in figure 1(b). In this case two parameters $\{x_1, x_2\}$ are considered. Initially the two extreme solutions of each parameter are selected, $\{a, b\}$ and $\{d, c\}$ for x_1 and x_2, respectively. Through this selection the set $S \equiv \{a, b, c, d\}$ is initialized. Then, solution e is selected because it has the larger distance to the set S. After that, solution f is selected for inclusion into the set $S \equiv \{a, b, c, d, e\}$, for the same reasons. The process is repeated until S population is completed.

The maximin sorting scheme is depicted in Algorithm 1. In each generation the new offsprings (set D) are merged with their progenitors (set P), according with the Algorithm 1, resulting in the new set R (line 1). After that, the algorithm may select, for each one of the optimization parameters (n_{par}), the extreme solutions (getMin and getMax functions) from rank 1 (lines 5-7) and introduces them into the final population (set S). Then the individuals of lower rank are removed (getRankMin function) from the auxiliary population A and inserted into the set S until the number of solutions of the current rank surpass the allowed number of solutions of set S (lines 9-12). Next, the distance squared, c_{a_j} (1a), between each rank solution, a_j, and the solutions already selected, s_i, is evaluated. Then, the solution a_j, whose squared distance to the set S is the larger (k solution), is selected (1b) (getMaxCi function). Each time a solution enters into the set S, the cost c_{a_l} of the set A is reevaluated (lines 19-21). This process ends when the set S is completed.

$$c_{a_j} = \min_{s_i \in S, a_j \in A} \parallel a_j - s_i \parallel^2 \tag{1a}$$

$$S = S \cup \{a_j : c_{a_j} = \max_{a_i \in A} c_{a_i}\} \tag{1b}$$

Algorithm 1 : Single-objective maximin algorithm.

$R = P \cup D$
$S = \emptyset$
$A =$ getRankMin(R)
if $\#A > pop_{\text{dim}}$ **then**
  for $i = 1$ **to** $n$par **do**
    $S = S \cup$ getMin$(A, i) \cup$ getMax(A, i)
  end for
end if
while $\#S + \#A \leq pop_{\text{dim}}$ **do**
  $S = S \cup A$
  $A =$ getRankMin(R)
end while
for $j = 1$ **to** $\#A$ **do**
  $c_{a_j} = \min\limits_{s_i \in S}\{\|a_j - s_i\|^2\}$
end for
while $\#S < pop_{\text{dim}}$ **do**
  $k =$ getMaxCi(A)
  $S = S \cup k$
  for $l = 1$ **to** $\#A$ **do**
    $c_{a_l} = \min\{\|a_l - k\|^2, a_l\}$
  end for
end while

2. CIRCUIT DESIGN AND RESULTS

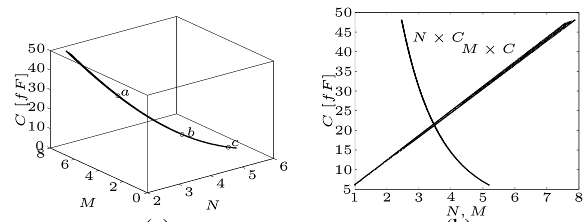
To access the algorithm performance, an application example in the area of RF integrated circuits is used. In this sense, the optimization procedure is employed to automate the design of a RFDSCA circuit with the goal of finding the component values that maximize its performance. The circuit is developed in a 0.25 μm BiCMOS technology and it is intended to be a cell of a ku-band voltage controlled oscillator (VCO) [3].

To design high performance RFDSCAs it is necessary to obtain the components values that maximize the RFDSCA quality factor (Q_{RFDSCA}). Because of that, the fitness function used in the automated RFDSCA circuit design algorithm ($f_v = 2\pi f Q_{\text{RFDSCA}}$) is the one defined in (2). In this equation the number of cells N, the two reference capacitors, both with value C, and the number of basic switches M are the optimization parameters. The technological parameters are the ON and OFF resistances of the basic switch ($R_{\text{BS-\{ON, OFF\}}}$, respectively) and its OFF capacitance ($C_{\text{BS-OFF}}$). Finally, the independent variable is defined by the control word D.

To fulfill the required VCO tuning range, the RFDSCA capacitance must present, at least, a minimum capacitance of 72 fF, a maximum capacitance of 108 fF and a maximum tuning step of 8 fF. These three capacitance constraints are evaluated from the RFDSCA capacitance equation (defined in (3)).

The algorithm uses 10^3 potential solutions, each one represented by N, M and C. These floating point values are randomly initialized in an appropriate range ($N =$ 1-64, $M =$ 1-64 and $C = 1$ fF-1 pF). The search is then carried out with this population over 10^7 generations. The fitness value, f_v, is given by (2) if the solution verifies the capacitance constraints, otherwise takes a negative value, proportional to the distance to the feasible decision region, if at least one restriction is not satisfied. The successive generations of new solutions are reproduced based on a linear ranking scheme and simulated binary crossover [4]. Finally, when mutation occurs the operator replaces the value of one optimization parameter according to a uniform distribution function. The uniform function varies in the range $[-U, U]$ where $U = \{0.2, 0.2, 0.4 \text{ fF}\}$ for N, M and C, respectively. The crossover and mutation probabilities are 0.6 and 0.05, respectively. The height of each rank is $\epsilon = 10^{11}$.

$$f_v(N, M, C) = \frac{\dfrac{\left(1+\frac{2MC_{\text{BS-OFF}}}{C}\right)\left(1+\frac{D}{D_{\max}}\frac{C}{2MC_{\text{BS-OFF}}}\right)}{R_{\text{BS-OFF}}C_{\text{BS-OFF}}}}{1+\frac{D}{D_{\max}}\left[\left(\frac{C}{2MC_{\text{BS-OFF}}}+1\right)^2\frac{R_{\text{BS-ON}}}{R_{\text{BS-OFF}}}-1\right]} \quad (2)$$



(a) (b)

Figure 2: (a) Optimal parameters front in the $N \times M \times C$ space. (b) Projections of the front in the $N \times C$ and $M \times C$ planes.

$$C_{\text{RFDSCA}}(N, M, C) = \frac{D\frac{C}{2} + D_{\max}\, MC_{\text{BS-OFF}}}{1+\frac{2MC_{\text{BS-OFF}}}{C}} \quad (3)$$

The RFDSCA circuits with optimum performance obtained by the genetic algorithm (GA) can be directly defined by the dot-points of figures 2(a) and 2(b). Figure 2(a) shows the optimum solution front in a 3D space and Figure 2(b) the corresponding projections in $N \times C$ and $M \times C$ planes. These two figures clearly show that the algorithm finds a front with good diversity. An important aspect about the algorithm performance is its convergence to the optimum solutions. The algorithm convergence ability is very good since all the solutions are in the same rank, where the maximum variation between the best and worst fitness solutions are lower than 2.25%. These charts reveal that the circuit can have several possible implementations with the same performance (the problem has not a single isolated optimum point, but, in fact an optimal front).

To infer the validity of the synthesis results, three RFDSCAs were implemented (defined by a, b and c in Figure 2(a)) and simulated on SpectreRF. The simulation results obtained on SpectreRF are very similar to the ones obtained with the proposed algorithm.

3. CONCLUSIONS

A synthesis procedure to automate the design of RFDSCAs is presented in this paper. The synthesis is carried on by a GA that promotes the distribution of the solution along the parameters space in order to give several optimal solutions. This method is based on closed-form symbolic mathematical expressions of the input impedance and quality factor of the RFDSCA. To verify the proposed synthesis method, three RFDSCAs were designed using three different solutions provided by the GA for the same design constraints. The results show that the synthesis and simulated outcomes are in very good agreement and also demonstrate that the GA is able to reach optimal solutions regarding the optimization objective. Moreover, the GA obtains a set of solutions along the optimal front in one run of the algorithm.

4. REFERENCES

[1] LAUMANNS, M. *et. al* Archiving with guaranteed convergence and diversity in multi-objective optimization. In *Proc. of the GECCO* (2002), pp. 439–447.

[2] PIRES, *et. al* Multi-Objective maximin sorting scheme. In *EMO* (2005), Springer-Verlag, LNCS Vol. 3410, pp. 165–175.

[3] MENDES, *et. al* A high-performance digitally controlled LC oscillator for ku-band applications. In *IEEE Int. Conf. on Electronics, Circuits and Systems* (Morroco, Dec. 2007).

[4] DEB, K. *Multi-Objective Optimization Using Evolutionary Algorithms*. John Wiley & Sons, LTD, 2001.