

Design of Radio-Frequency Integrated CMOS Discrete Tuning Varactors Using the Particle Swarm Optimization Algorithm

E.J. Solteiro Pires¹, Luís Mendes^{2,*}, P.B. de Moura Oliveira¹,
J.A. Tenreiro Machado³, João C. Vaz^{4,*}, and Maria J. Rosário^{4,*}

¹ Centro de Investigação e de Tecnologias Agro-Ambientais e Biológicas,
Universidade de Trás-os-Montes e Alto Douro, Portugal

epires@utad.pt, oliveira@utad.pt

² Escola Superior de Tecnologia e Gestão, Instituto Politécnico de Leiria, Portugal
Instituto de Telecomunicações, Portugal

lmendes@estg.ipleiria.pt

³ Inst. Sup. de Engenharia do Porto, Instituto Politécnico do Porto, Portugal

jtm@isep.ipp.pt

⁴ Instituto Superior Técnico, Universidade Técnica de Lisboa, Portugal
Instituto de Telecomunicações, Portugal

joaovaz@ist.utl.pt, mrosario@lx.it.pt

Abstract. This paper presents an automated design procedure of radio-frequency integrated CMOS discrete tuning varactors (RFDTV). This new method use the maximin and the particle swarm optimization (PSO) algorithms to promote well distributed non-dominated fronts in the parameters space when a single-objective function is optimized. The fitness function used in the search tool is proportional to the RFDTV quality factor. The outcome of the automated design method comprises a set of RFDTV circuits, all having the same maximum performance. Each solution, which corresponds to one RFDTV circuit, is defined by the number of cells and by the circuit components values. This approach allows the designer to choose among several possible circuits the one that is easier to implement in a given CMOS process. To validate the effectiveness of the synthesis procedure proposed in this paper (PSO-method) comparisons with a design method based on genetic algorithms (GA-method) are presented. A 0.18 μm CMOS radio-frequency binary-weighted differential switched capacitor array (RFDSCA) was designed and implemented (the RFDSCA is one of the possible topologies of the RFDTVs). The results show that both design methods are in very good agreement. However, the PSO technique outperforms the GA-method in the design procedure run time taken to accomplish the same performance results.

Keywords: Particle swarm optimization, analog circuit design, automated circuit synthesis and radio-frequency integrated circuits.

* The authors would like to thank to *Fundação para a Ciência e a Tecnologia* for funding the project PTDC/EEA-ELC/64368/2006.

1 Introduction

The radio-frequency (RF) integrated CMOS discrete tuning varactor (RFDTV) is one of the key circuits in the implementation of broad-tuned inductance-capacitance resonant circuits, which are largely used in wireless transceivers. One of the approaches to design RFDTVs is based on RF binary-weighted switched capacitors arrays (RFSCAs) [1], which uses high quality factor capacitors and RF switches. Nowadays, this kind of circuits is widely used, *e.g.* in low-phase-noise ultra-wide-band voltage controlled oscillators [2], in low-noise fast-settling frequency synthesizers [3], in process dispersion compensation techniques [4] and in adaptive impedance matching circuits [5]. Furthermore, the RFSCAs have a great usage potential in reconfigurable or adaptive RF circuits for multi-mode, multi-band and multi-standard wireless transceivers. In the last years, the growing demand for full-integrated CMOS high performance wireless systems has spurred the researchers worldwide to develop new innovative techniques to optimize and automate the design of RF circuits [6,7,8,9,10,11].

Since 1950 that several numerical methods based on phenomena existing in nature have been proposed with the aim of finding a solution or solutions to optimization problems. One of them is the genetic algorithm (GA) which is inspired on organic evolution. This algorithm uses some concepts from evolutionary biology such as inheritance, selection, crossover and mutation. Another method, in this case inspired on social-psychological behaviors, is the particle swarm optimization (PSO) algorithm. This technique is typically modeled by particles that are characterized by hyperspace dynamic equations (position and velocity). In the computation field, there are single-objective problems that can be solved in a myriad of different ways, each one characterized with similar maximum fitness values. Therefore, achieving a well-spread and non-dominated parameter front is of paramount importance to the decision maker, since he can choose from a set of optimum solutions the best suited one to be developed or implemented in the problem context.

Pires *et al.* [12] proposed a single-objective multi-parameters optimization algorithm based on GAs (GA-method) to design maximum performance differential RFSCAs (RFDSCAs). Although the circuits obtained with this automated synthesis method present good performance, the computational time is too high. In order to improve the execution time, a new method based on a particle swarm optimization (PSO) algorithm (PSO-method) was developed and used to design RFDSCAs. As in the GA-method, this new algorithm uses the maximin concept [13] together with the ϵ -dominance [14] in order to achieve solutions diversity in the parameters space. Although the circuits found by the two synthesis methods have similar characteristics and performances the time taken by the PSO-method is substantially lower than the one of the GA-method (due to its simplicity). Section 2 presents the RFDSCA circuit and its model. Besides that, the initial restrictions and objective function for this type of RFDTVs circuits are also defined. Section 3 presents and describes the algorithm associated to the PSO-method and section 5 discusses and compares the results obtained by the two design methods. Finally, section 6 outlines the main conclusions.

2 RFDSCA Circuit and Behavioral Model

The block diagram of a RFDSCA is shown in figure 1(a). It consists of N cells, each one constituted by two cell capacitors and a cell switch. The first cell, named reference cell, is comprised by two reference capacitors, both with value C , and a reference switch. This switch is formed by placing in parallel M basic switches (BS), as illustrated in figure 1(b). For the other cells, the capacitors values are equal to $2^{i-1} \times C$ and the number of reference switches for each cell switch is given by 2^{i-1} , where i is the cell number [15,16]. All the switches are implemented by NMOS single-finger transistors.

The RFDSCA can be modeled as a non-ideal controlled capacitor, where its equivalent capacitance, C_{RFDSCA} , and quality factor, Q_{RFDSCA} , are expressed by (1) and (2), respectively. In these equations, f is the operating frequency, D represents the control word (decimal representation of the control binary word, $D = b_N \cdot 2^{N-1} + \dots + b_2 \cdot 2^1 + b_1 \cdot 2^0$) and $D_{\text{max}} = 2^N - 1$ corresponds to the maximum value of D . The RFDSCA model is based on 6 design parameters and 2 independent variables (D and f). The design parameters that can be optimized to obtain a RFDSCA with optimum performance are N , M and C , since $R_{\text{BS-ON}}$, $R_{\text{BS-OFF}}$ and $C_{\text{BS-OFF}}$ are defined by the integration technology and bias conditions.

$$C_{\text{RFDSCA}}(D) = \frac{D \frac{C}{2} + D_{\text{max}} M C_{\text{BS-OFF}}}{1 + \frac{2 M C_{\text{BS-OFF}}}{C}} \tag{1}$$

$$Q_{\text{RFDSCA}}(D, f) = \frac{\left(1 + \frac{2 M C_{\text{BS-OFF}}}{C}\right) \left(1 + \frac{D}{D_{\text{max}}} \frac{C}{2 M C_{\text{BS-OFF}}}\right)}{\frac{2 \pi f R_{\text{BS-OFF}} C_{\text{BS-OFF}}}{1 + \frac{D}{D_{\text{max}}} \left[\left(\frac{C}{2 M C_{\text{BS-OFF}}}\right)^2 \frac{R_{\text{BS-ON}}}{R_{\text{BS-OFF}}} - 1\right]}} \tag{2}$$

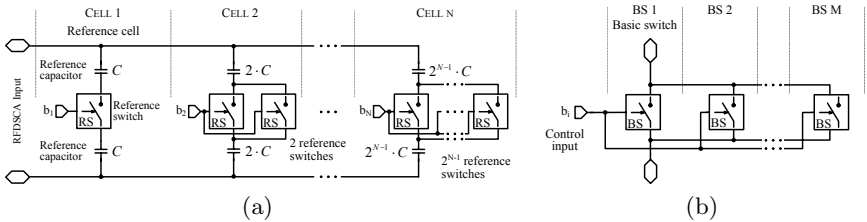


Fig. 1. (a) Binary-weighted RFDSCA topology and (b) reference switch structure

3 Modified Particle Swarm Optimization Algorithm

The PSO algorithm was originally proposed by James Kennedy and Russell C. Eberhart [17]. This optimization technique is inspired in the way swarms

(*e.g.* flocks of birds, schools of fishes, herds, *etc.*) elements move in a synchronized way as a defensive tactic. An analogy is established between a particle and an element of swarm. The particle movement is characterized by two vectors representing its current position x and velocity v .

Since 1995 the PSO algorithm has evolved greatly. Several techniques have been proposed with the aim to refine and complement the original algorithm. Some of them have improved the tuning parameters [18] and others use the PSO algorithm with evolutionary techniques [19] to obtain better performances.

In this study the standard PSO algorithm, represented by the pseudo-code of figure 2, was first considered. The basic algorithm begins by initializing the swarm randomly in the search space (line 1). In each algorithm iteration (lines 2 to 8) the particle position is changed by adding a new velocity (line 4 and 5). This new velocity vector is evaluated by summing an increment, which is a function of the cognitive knowledge and the social knowledge components, to the original velocity. The particle cognitive knowledge is equal to the difference between the current position x_t and its best position so far b and the social knowledge of each particle is given by the difference between x_t and the best swarm global position achieved so far g . The cognitive and social knowledge factors are multiplied by a randomly generated constant ϕ_1 and ϕ_2 , respectively (line 4). The velocity of particles are restricted in order to keep velocities from exploding.

Since standard PSO algorithm is inadequate to find several optimum solutions, some modifications were performed in it. The proposed algorithm (PSO-method), defined by the pseudo-code of figure 3, uses the ϵ -dominance and the maximin selection techniques, both adopted from multi-objective optimization problems. The ϵ -dominance concept allows that somewhat inferior fitness values remain in the population as non-dominated ones [14] and the maximin technique is mainly used to obtain a set of points from a continuous front instead of finding the peaks of modality functions. This approach allows that the PSO-method find a set of solutions with good spread and diversity over the parameters space.

The maximin technique used in the modified PSO algorithm can be summarized as follows: initially, the algorithm split the objective space using virtual horizontal parallel hyperplanes (straight lines or planes for a 1-D or 2-D dimensional

```

1: Initialize swarm
2: while !stopping criteria do
3:   for  $l = 1$  to #swarm do
4:      $v_{t+1} = \omega \cdot v_t + \phi_1 \cdot (b - x_t) + \phi_2 \cdot (g - x_t)$ 
5:      $x_{t+1} = x_t + v_{t+1}$ 
6:      $x_{t+1} \in [-x_{\min}, +x_{\min}]$ 
7:   end for
8: end while

```

Fig. 2. Pseudo-code of the standard PSO algorithm

```

1: Initialize swarm
2: while !stopping criteria do
3:   for  $l = 1$  to #swarm do
4:      $v_{t+1} = \omega \cdot v_t + \phi_1 \cdot (b - x_t) + \phi_2 \cdot (g - x_t)$ 
5:      $x_{t+1} = x_t + v_{t+1}$ 
6:      $x_{t+1} \in [-x_{\min}, +x_{\min}]$ 
7:   end for
8:    $Q(t+1) = P(t+1)$ 
9:    $Arc(t+1) = \text{maximin}(Arc(t) \cup Q(t+1))$ 
10:   $P(t+1) = Arc(t+1)$ 
11:   $t = t + 1$ 
12: end while

```

Fig. 3. Pseudo-code of the modified PSO algorithm (include the maximin technique)

parameter space, respectively), that are separated from each other by a ϵ -distance. Two consecutive hyperplanes define a ϵ -rank, where all the solutions have the same preference, even if their objective values are different. After the division of the objective space, the algorithm select the solutions, starting from the better ϵ -ranks, until a ϵ -rank is found with more solutions than the empty slots available in the new population. However, if the best ϵ -rank has more solutions than the number of individuals of the new population, then the solutions of the other ϵ -ranks are not considered. In both cases, the best distributed solutions in the parameter space are selected according to the maximin selection [12]. The main concept behind the maximin sorting scheme is to select the solutions in order to decrease the large gap areas where no solution exists in the already selected population. As shown in figure 3, the maximin algorithm is called (line 9) at the end of each iteration (line 2 to 12) of the modified PSO algorithm. Besides that, to ensure diversity along the iterations, the algorithm uses an archive (line 10) to store the solutions most dispersed in the parameters space.

4 Automated Synthesis Procedure

To automate the RFDSAs design using the algorithm described in section 3, it is necessary to define the restrictions and the fitness function appropriated for these type of circuits. The design restrictions, which are related to the initial RFDSA specifications, are defined in equation (3).

$$C_{\text{RFDSA-MAX}} \geq C_{\text{MAX}}, \quad C_{\text{RFDSA-MIN}} \leq C_{\text{MIN}}, \quad \Delta C_{\text{RFDSA}} \leq \Delta C \quad (3)$$

The three inequations show that the minimum value for the RFDSA maximum capacitance ($C_{\text{RFDSA-MAX}}$) must be greater or equal than the desired maximum capacitance (C_{MAX}), the maximum value for the RFDSA minimum capacitance ($C_{\text{RFDSA-MIN}}$) must be less or equal to the required minimum capacitance (C_{MIN}) and the maximum value for the RFDSA capacitive tuning step (ΔC_{RFDSA}) must be less or equal than the initially specified capacitive tuning step (ΔC). To achieve high performance RFDSAs, it is necessary to determine the number of cells and components values and sizes that maximizes the RFDSA quality factor. Therefore, the chosen optimization objective function is given by expression (4).

$$f_v = 2\pi \cdot f \cdot Q_{\text{RFDSA}} \quad (4)$$

The optimization algorithm uses a swarm size of 10^3 potential solutions, each one represented by the optimization parameters N , M and C . These floating point values are randomly initialized in an appropriate range ($N = 1 \dots 64$, $M = 1 \dots 64$ and $C = 1\text{fF} \dots 1\text{pF}$). The search is then carried out with this population over 10^5 iterations. The fitness value, f_v , is given by (4) if the solution verifies the restrictions, otherwise takes a negative value, proportional to the distance to the feasible decision region, if at least one restriction is not satisfied. Parameters ϕ_1 and ϕ_2 are uniformly generated in the interval $[0, 1]$. The weight coefficient is $\omega = 0.7$ and the height of each rank is $\epsilon = 10^{11}$.

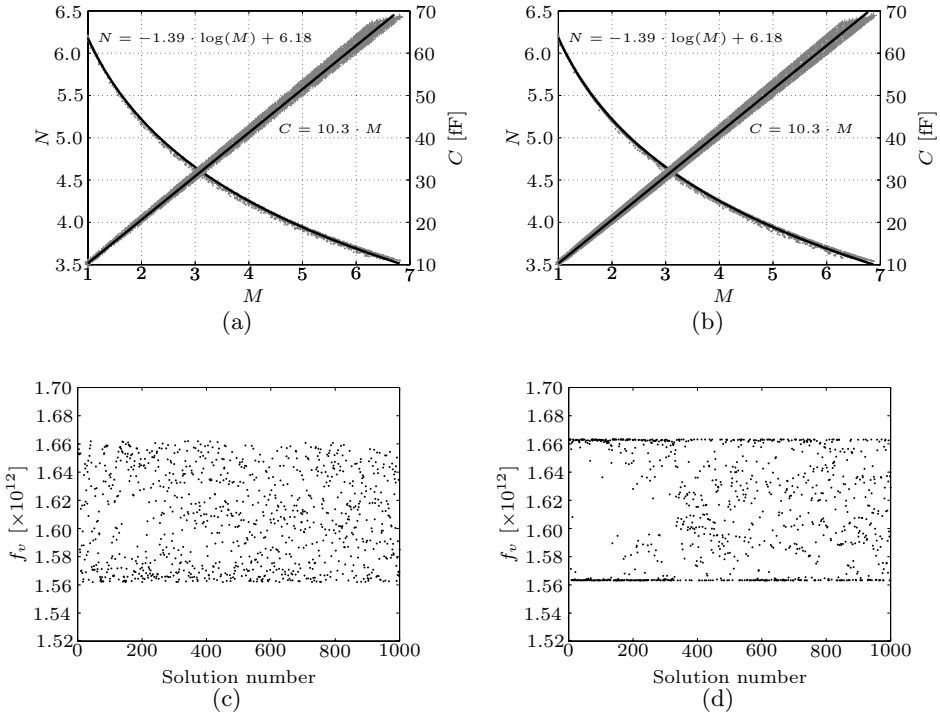


Fig. 4. Non-dominated solutions of the (a) PSO-method and (b) GA-method. Fitness values for each solution obtained with the (c) PSO-method and (d) GA-method.

5 Simulation and Implementation Results

This section presents the development of a $0.18 \mu\text{m}$ CMOS RFDSOA intended for a 3 to 4 GHz voltage controlled oscillator (VCO). To fulfill the required VCO tuning range, the RFDSOA must present a $C_{\text{min}} = 153.6 \text{ fF}$, $C_{\text{max}} = 366.2 \text{ fF}$ and $\Delta C = 20 \text{ fF}$. In the design both PSO and GA methods were used with the same constraints and objective function. Moreover, the number of potential solutions, number of iterations and the height of each rank were also the same for both techniques. The two algorithms were developed in C++ and executed in a Pentium 4 at 3 GHz with 500 Mbytes of memory. This design approach allows a fair comparison between the performances of the two algorithms.

The optimum performance circuits obtained with the PSO and GA methods are given in figures 4(a) and 4(b), respectively. These curves were obtained considering $R_{\text{BS-ON}} = 60 \Omega @ V_{\text{GS}} = 1.8\text{V}$, $R_{\text{BS-OFF}} = 48 \Omega @ V_{\text{GS}} = 0\text{V}$ and $C_{\text{BS-OFF}} = 7.24 \text{ fF} @ V_{\text{GS}} = 0\text{V}$. These values were determined from the SPICE

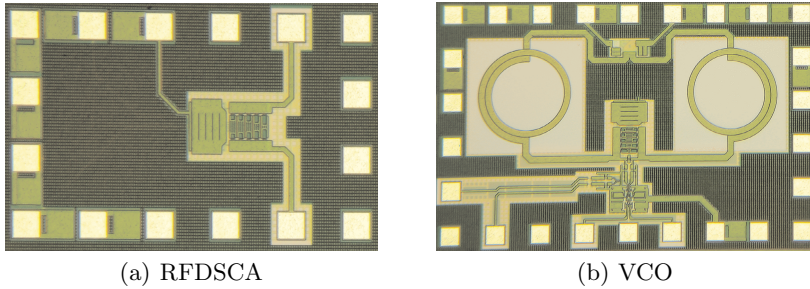


Fig. 5. Photomicrographs

models of the NMOS transistors and from the technological process parameters. The graphs 4(a) and 4(b) present the curves of the design parameters C and N versus M for the RFDSKA and for the two design procedures. These curves clearly show that both algorithms find a front with good diversity. Besides that, the two algorithms convergence ability is very good since all the solutions of each technique are in the same rank (see figures 4(c) and 4(d)). Moreover, the fitting equations of C versus M and N versus M shown in figures 4(a) and 4(b) are equal and all solutions of both techniques have similar fitness values. So, the results of figure 4 allows concluding that both techniques, in terms of the final results, are equivalent. The major difference between the PSO-method and the GA-method is in the algorithms execution time. The PSO technique takes 4h20m to perform the optimization and the GA algorithm 7h30m.

To select a RFDSKA circuit with maximum quality factor it is only necessary to choose N , M and C from the figure 4(a) or 4(b). The selected solution should simplify the RFDSKA circuit implementation. The selected circuit is characterized by $N = 4$, $M = 4$ and $C = 43.36$ fF. The photomicrographs of the implemented RFDSKA and VCO are in figure 5.

6 Conclusions

A synthesis procedure to automate the design of RFDSKAs is presented in this paper. The synthesis is carried on by a modified PSO algorithm (it includes the ϵ -dominance and maximin techniques) which promotes the distribution of a family of optimal solutions (circuits) along a non-dominated parameter front. This means that a RFDSKA circuit can be implemented with different number of cells and components values. This method is based on closed-form symbolic mathematical expressions of the input impedance and quality factor of the RFDSKA.

To verify the proposed synthesis method, the PSO results were compared with the ones obtained by a GA-method. Both algorithms reaches the same front. However, the PSO-method requires considerably less time that the GA.

Additionally, one RFDSCA circuit was implemented and its measurements are in agreement with the correspondent simulated results.

References

1. Kral, A., Behbahani, F., Abidi, A.: RF-CMOS oscillators with switched tuning. In: IEEE CICC (May 1998)
2. Fard, A.: Phase noise and amplitude issues of a wide-band VCO utilizing a switched tuning resonator. In: IEEE ISCAS (May 2005)
3. Wu, S.: A low-noise fast-settling PLL frequency synthesizer for CDMA receivers. In: SoC (November 2004)
4. Ang, K., Chia, M., Li, D.: A process compensation technique for integrated VCO. In: IEEE RFIC (June 2004)
5. Sjöblom, P., Sjöland, H.: An adaptive impedance tuning CMOS circuit for ISM 2.4-GHz band. *IEEE Trans. on Circ. and Syst.* 52(6), 1115–1124 (2005)
6. Somani, A., Chakrabarti, P.P., Patra, A.: An evolutionary algorithm-based approach to automated design of analog and RF circuits using adaptive normalized cost functions. *IEEE Trans. on Evol. Comp.* (in press)
7. Kaitharam, S., Rajagopal, C., Nunez, A.: SRFCC: synthesis of RF CMOS circuits. In: IEEE MWSCAS (August 2002)
8. Rajagopal, C., Sridhar, K., Nunez, A.: RF CMOS circuit optimizing procedure and synthesis tool. In: ACM VLSI (April 2003)
9. Konstantopoulos, G., Papathanasiou, K., Samelis, A.: Optimization of RF circuits by expert system monitored genetic computation. In: IEEE ISCAS (May 2006)
10. Chu, M., Allstot, D.J., Huard, J.M., Wong, K.Y.: NSGA-based parasitic-aware optimization of a 5 GHz low-noise VCO. In: IEEE ASP DAC (January 2004)
11. Sen, P., Pratap, R.J., Mukhopadhyay, R., Sarkar, S., Lee, C.H., Pinel, S., May, G.S., Laskar, J.: Neuro-genetic design centering of millimeter wave oscillators. In: IEEE SiRF (January 2006)
12. Solteiro Pires, E.J., Mendes, L., de Moura Oliveira, P.B., Machado, J.A.T., Ferreira, N.M.F., Vaz, J., Rosário, M.: Single-objective front optimization: application to RF circuit design. In: GECCO 2008: Proc. of the 10th annual conf. on Genetic and evolutionary computation, USA, pp. 765–766. ACM, New York (2008)
13. Solteiro Pires, E.J., de Moura Oliveira, P.B., Machado, J.A.T.: Multi-objective MaxiMin Sorting Scheme. In: Coello Coello, C.A., Hernández Aguirre, A., Zitzler, E. (eds.) EMO 2005. LNCS, vol. 3410, pp. 165–175. Springer, Heidelberg (2005)
14. Laumanns, M., Thiele, L., Deb, K., Zitzler, E.: Archiving with guaranteed convergence and diversity in multi-objective optimization. In: Proc. of the Genetic and Evolutionary Computation Conf., pp. 439–447. Morgan Kaufmann Publishers, San Francisco (2002)
15. Mendes, L., Vaz, J.C., Rosário, M.J.: A closed-form input admittance solution for RF and microwave switched capacitor arrays. In: IEEE MWSCAS, Puerto Rico (August 2006)
16. Mendes, L., Vaz, J.C., Rosário, M.J.: Performance of Si-integrated wide-band single-ended switched capacitor arrays. In: 13th IEEE ICECS, France (December 2006)
17. Kennedy, J., Eberhart, R.C.: Particle swarm optimization. In: Proc. of the 1995 IEEE Int. Conf. on Neural Networks, Perth, Australia, vol. 4, pp. 1942–1948. IEEE Service Center, Piscataway (1995)

18. Shi, Y., Eberhart, R.C.: Empirical study of particle swarm optimization. In: Angeline, P.J., et al. (eds.) Proceedings of the Congress of Evolutionary Computation, Washington D.C, vol. 3, pp. 1945–1950. IEEE Press, Los Alamitos (1999)
19. Løvbjerg, M., Rasmussen, T.K., Krink, T.: Hybrid particle swarm optimiser with breeding and subpopulations. In: Spector, L., et al. (eds.) Proc. of the Genetic and Evolutionary Computation Conf., pp. 469–476. Morgan Kaufmann, San Francisco (2001)