AN OPTIMIZATION-BASED TOOL FOR CIRCUIT LEVEL SYNTHESIS ANALOG INTEGRATED CIRCUITS

Lucas Compassi Severo, Alessandro Girardi

Federal University of Pampa – UNIPAMPA
Campus Alegrete
Av. Tiaraju, 810, Alegrete – RS – Brazil
lucascs.eletrica@gmail.com, alessandro.girardi@unipampa.edu.br

ABSTRACT

The goal of this paper is to present a tool for automatic sizing of analog basic integrated blocks using heuristics of non-linear optimization and an external electrical simulator. The methodology is based on the minimization of a cost function and constraint parameters related to circuit electrical characteristics. The methodology was implemented in Matlab using two heuristics of non-linear optimization, Simulated Annealing (SA) and Genetic Algorithms (GA). As electrical simulations we used the Smash simulator and the ACM MOSFET model. As design example, this paper shows the application of this methodology for the design of an active load differential amplifier using AMS 0.35µm technology.

1. INTRODUCTION

The design automation of analog integrated circuits can be very useful in microelectronics, because it provides an efficient search for the circuit variables, among a set of design constraints, to make it more efficient as possible. Several works have been done in this theme, aiming the development of tools for Analog Design Automation (ADA) with the goal of automation of time-consuming tasks and complex searches in highly non-linear design spaces [1, 2]. However, as far as we know, there is not a commercial tool capable to perform the synthesis of analog circuits with optimum results in a feasible time.

An important improvement in the analog design could be the automation of some design stages, such as system level, circuit level and layout level [3], maintaining the interaction with the human designer. The large number of design variables and the consequent large design space turn this task extremely difficult to perform even for most advanced computational systems. Therefore, it is mandatory the use of artificial intelligence with great computational power to solve these problems.

The automatic design optimized-based is divided in two types – based on equation model or electrical simulation. In the electrical simulation approach the algorithm is based on the result of the electrical simulation of analog block provided by an external electrical simulator.

In this context, we propose an automatic synthesis procedure for basic analog building blocks which is capable to size transistors width (W) and length (L) with efficient time and ordinary computational resources.

The proposed synthesis procedure provides some options for the designer about the form of design automation, allowing the choice between optimization heuristics and external electrical simulators. In this context, we used as optimization heuristic Simulated Annealing (SA) and Genetic Algorithms (GA), and as external electrical simulator the Smash® Simulator.

This work is organized as follows: section 2 shows the description of the proposed methodology; section 3 presents the application of the methodology in the design of a specific analog block - the differential amplifier - with circuit description and comparison of final results; finally, section 4 shows the conclusion.

2. A METHOLOGY FOR ANALOG DESIGN AUTOMATION (ADA)

The proposed methodology has the goal to provide a tool for circuit level analog design automation (ADA) [4]. It is based on the automatic sizing of MOSFETs transistor for analog basic block based on a set of specifications. The automatic sizing is made for a heuristic for a non-linear optimization using results of electrical simulation provided by an external electrical simulator. The tool diagram is shown in the Figure 1. The tool has as input the specifications of analog integrated circuit (netlist and parameter model, for example), the optimization heuristics, the external electrical simulator and the technology model that will be used.

The optimization heuristic is some meta-heuristics for non-linear optimization. The goal of optimization heuristic is to reduce a cost function for a given analog basic block. This function is based on some design constraints, as power dissipation and silicon area, for example. We used in this paper Simulated Annealing and Genetic Algorithms as optimization heuristics.

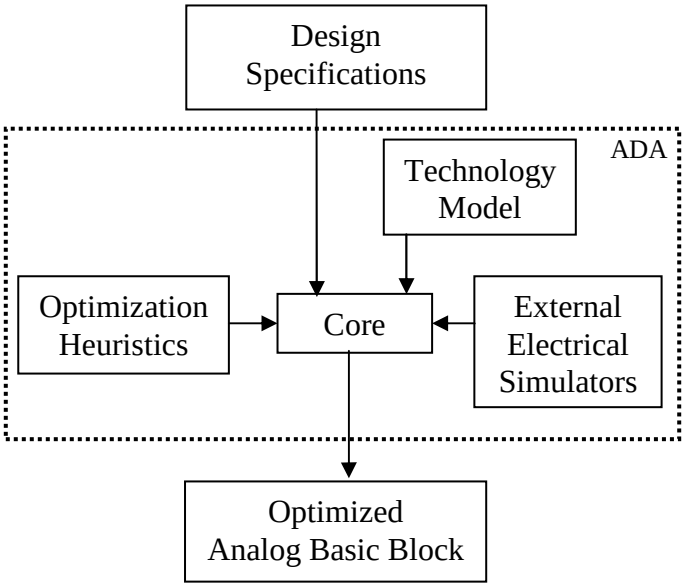

Figure 1 – Diagram of ADA tool

Simulated Annealing is a meta-heuristic for non-linear optimizations and it is inspired on the analogy of a thermodynamic principle to simulate the cooling of a heated set of atoms. This algorithm receives as input an initial solution and generate solutions in a random form. A strategy for escaping from local minima is based on the probability to accept a worst solution. This probability is dependent of a temperature parameter [5].

In this paper we used the *Asamin* implementation for Matlab, developed by Shinichi Sakata [6].

Genetic algorithms are also a meta-heuristic for non-linear optimization and explore the analogy with biologic evolution theories. It is a non-deterministic algorithm and it works with a variety of solutions (population), simultaneously. The size of the population is defined in order to maintain an acceptable diversity considering an efficient optimization time. Each possible solution of population is denominated chromosome, which is a chain of characters (gens) that represent the circuit variables. The algorithm creates initial solutions randomly and generates new solutions based on the recombination and mutation parameters. So, it is not necessary an initial solution [7].

We used in this work the *GAOT* implementation (Genetic Algorithms Optimization Toolbox) for Matlab developed by Christopher R. Houck et al [8].

In this work we used the ACM model for AMS035 technology. The ACM model includes a reduced set of parameters and is continuous in all regions of operation. In this case, these characteristics provide an efficient search in the design space [9].

As external electrical simulator we used Smash® Simulator.

3. DESIGN EXAMPLE – DIFFERENTIAL AMPLIFIER

As an application for the proposed tool, we implemented a design example using a CMOS differential amplifier as analog basic block. The differential amplifier is one of the most versatile circuits in analog design. It is compatible with ordinary CMOS integrated-circuit technology and serves as input stage for op amps [10]. Its basic function is to amplify the difference between the input voltages. The circuit for a differential amplifier with active load is basically composed by a load current mirror (M3 and M4), a source-coupled differential pair (M1 and M2) and a reference current mirror (M5 and M6), shown in fig.2. The main electrical parameters of the circuit are low-frequency voltage gain (Av0), gain-bandwidth product (GBW), slew-rate (SR), input common-mode range (ICMR), dissipated power (Pdiss) and area (A), among others.

The low-frequency gain is the relationship between output and input voltages, defined as:

$$Av_0 = \frac{g_{m1}}{g_{ds2} + g_{ds4}} \quad (1)$$

where g_{m1} is the gate transconductance of transistor M1 and g_{ds2} and g_{ds4} are the output conductance of M2 and M4, respectively.

The slew rate (SR) is the maximum output-voltage rate, either positive or negative, given by:

$$SR = \frac{I_{ref}}{C_1} \quad (2)$$

Here, I_{ref} is the current source of circuit C1 is the total output capacitance. This capacitance is estimated as the sum of the load capacitance CL and drain capacitance of M2 and M4. Input common-mode range (ICMR) is the maximum and minimum input common-mode voltage, defined as:

$$ICMR^- = v_{DS5(sat)} + v_{GS1} + v_{SS} \quad (3)$$

$$ICMR^+ = v_{DD} - v_{GS3} + v_{TN1} \quad (4)$$

In this case, $v_{DS5(sat)}$ is the saturation voltage of transistor M5, v_{GS1} and v_{GS3} are gate-source voltages of M1 and M3, respectively, v_{DD} and v_{SS} are voltage sources of the circuit and v_{TN1} is the threshold voltage of M1. The v_{DD} and v_{SS} source voltages are defined in this example as -1.65V and 1.65V, respectively.

The gain-bandwidth product is given by:

$$GBW = \frac{g_{m1}}{C_1} \quad (5)$$

The cost function for the circuit, in this case, is related to the power dissipation, defined as:

$$f = \frac{(v_{DD} + |v_{SS}|).I_{ref}}{P_0} + R \quad (6)$$

where R is a penalty constraint function, which will be a large value if the constraints are not met, and zero if all constraints are met. P_0 is the reference power dissipation for normalization purpose.

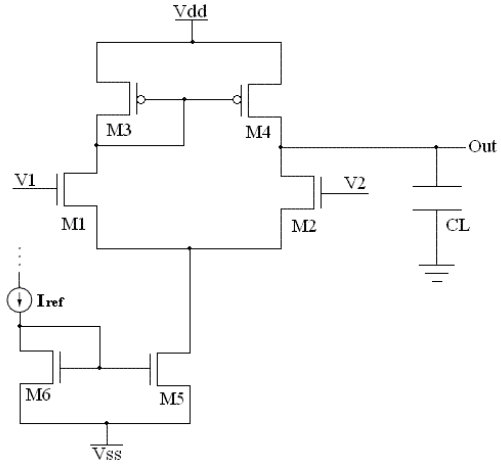

Fig.2 - Schematics of a differential amplifier

The optimization was executed in a computer with Intel dual core 1.73GHz processor and 2GB of memory. The optimization using Simulated Annealing (SA) is executed and the Figure 3 show the evolution for cost function and the Table 1 show the results. The comparison between the initial values and the final values are shown in Table 2, where it is possible to see that this heuristic is dependent of good initial values as input. So, these initial values are obtained by manual calculation based on the design equations of the circuit. In the Table 1 it is possible observe that all values of required specifications are reached.

Table 1: Specifications required and designed for the optimization using Simulated Annealing.

Specification	Required	Designed
Av0	55dB	55,07 dB
SR	5 V/µs	8,09 V/µs
ICMR+	0.7V	1,11 V
ICMR-	-0.7V	-0,83 V
GBW	1MHZ	7,00 MHz

Table 2: Initial and final values for the optimization using Simulated Annealing.

Variable	Initial value	Final value
W(M1, M2)	20 µm	45,76 µm
W(M3, M4)	30 µm	7,87 µm
W(M5, M6)	100 µm	77,03 µm
L(M1, M2)	10 µm	1,52 µm
L(M3, M4)	15 µm	1,55 µm
L(M5, M6)	5 µm	23,39 µm
Iref	100 µA	50,50 µA

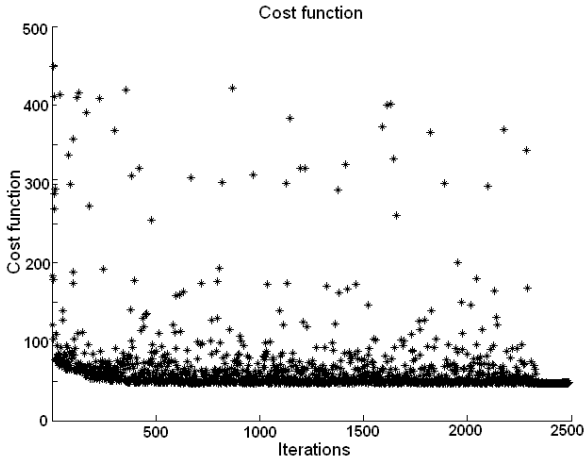

Figure 3: Evolution of cost function for optimization using SA.

For the genetic algorithm (GA) we made optimizations with three different populations: 10, 100 and 1000 individuals. The choice of population size is defined based on the execution time and on the quality of the solution. Table 3 shows the results for the genetic algorithms using the same specifications used in the simulated annealing optimization. Considering power dissipation (the goal), it is possible to verify that the best results are reached by a population of 1000 individuals, at the expense of larger optimization time, the Figure 4 show the evolution of cost function. Table 4 shows the initial and final values of optimization using Genetic Algorithms. We can notice that with this heuristic the initial values are not relevant, because all values are generated randomly.

Table 3: Specifications achieved using the Genetic Algorithm with three population sizes.

Specification	Pop=10	Pop=100	Pop=1000
GBW	3.91MHz	949kHz	5.95MHz
SR	5.12V/µs	5.00 V/µs	5.00 V/µs
Av0	61.18 dB	62.29dB	60dB
ICMR-	-1.02V	-0.94V	-0.70V
ICMR+	1.07V	0.81V	1.03V
Power dissipation	148.33µW	148.80µW	139.46µW
Time	22min	19min	25min
Generations	2524	2354	2026

Table 4: Initial and final values of optimization using GA.

Variable	Initial value	Optimized value
W(M1 e M2)	random	99.98 µm
W(M3 e M4)	random	24.17 µm
W(M5 e M6)	random	67.49 µm
L(M1 e M2)	random	1.85 µm
L(M3 e M4)	random	2.56 µm
L(M5 e M6)	random	23.96 µm
Iref	random	51.22 µA

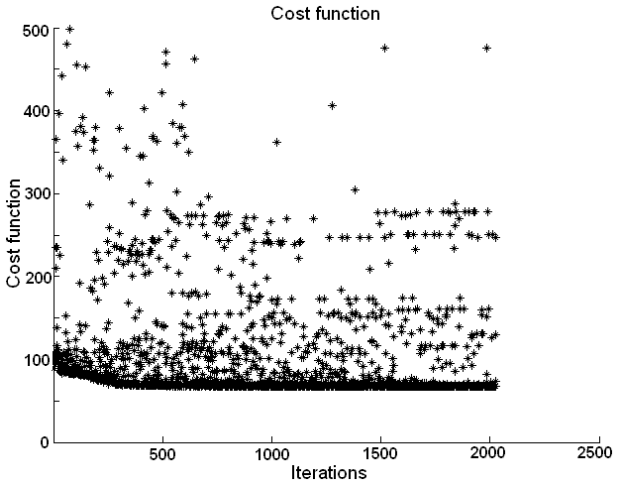

Figure 4: Evolution of cost function for optimization using GA with 1000 individuals.

In the Table 5 a comparison between GA and SA is shown. Analyzing the results we can see that both heuristics arrived near the same goal function, with a difference of 3.37 µW. In the optimization time, the GA is about 12% faster. The gate area presented similar values for both methodologies.

Table 5: comparison of results between GA and AS optimizations.

Parameter	GA	SA
Time	25 minutes	28 minutes
W(M1, M2)	99.98 µm	45.76 µm
W(M3, M4)	24.17 µm	7.87 µm
W(M5, M6)	67.49 µm	77.03 µm
L(M1, M2)	1.85 µm	1.52 µm
L(M3, M4)	2.56 µm	1.55 µm
L(M5, M6)	23.96 µm	23.39 µm
Iref	51.22 µA	50.50 µA
Av0	59.09 dB	55.07 dB
SR	5.00 V/µs	8.09 V/µs
ICMR+	1.03	1.11 V
ICMR-	-0.70V	-0.83 V
GBW	5.95 MHz	7,00 MHz
Gate area	1863.89 µm²	1883.49 µm²
Cost (Power dissipation)	169.03 µW	165.66 µW

4. CONCLUSION

The proposed tool for analog design automation of basic analog building blocks presented good results in a reasonable computing time. It was possible to see that genetic algorithms and simulated annealing achieved similar results, generating solutions that meet all design specifications.

A good characteristic of Genetic algorithms in relation the Simulated Annealing is the fact that it is not necessary to calculate a feasible initial solution.

Electrical simulator using the ACM model implemented in this methodology guarantee the search in all regions of operation of MOSFET transistors.

As future work, we intend to develop a framework for the automatic synthesis of analog circuits. Also, we can explore the use of electrical simulators from different vendors, expand the methodology for other analog basics blocks and create a friendly interface for the human designer.

5. ACKNOWLEDGMENTS

The grant provided by FAPERGS research agency for supporting this work is gratefully acknowledged.

6. REFERENCES

[1] M. Degrauwe, O. Nys, E. Dukstra, J. Rijmenants, S. Bitz, B. L. A. G. Go®art, E. A. Vittoz, S.Cserveny, C. Meixenberger, G. V. der Stappen, and H. J. Oguey, “IDAC: An interactive design tool for analog CMOS Circuits”, IEEE Journal of Solid-State Circuits, SC-22(6):1106{1116, December 1987

[2] M. D. M. Hershenson, S. P. Boyd, and T. H. Lee, “Optimal design of a CMOS op-amp via geometric Programming”, IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems, 20(1):1{21, January 2001.

[3] A. Girardi and S. Bampi, “LIT - An Automatic Layout Generation Tool for Trapezoidal Association of Transistors for Basic Analog Building Blocks”, Design automation and test in Europe, 2003.

[4] Balkir S., Dündar G., Ögrenci, A S., Analog VLSI Design Automation, CRC Press, Florida, 2000.

[5] Vanderplaats, G. N., Numerical Optimization Techiniques for Engineering Design: with Applications, Mcgraw-Hill, New York, 1984.

[6] Sakata, S. Asamin: a Matlab gateway routine to adaptive simulated annealing (ASA). available at: http://www.econ.ubc.ca/ssakata/public_html/software. [7] P. Venkataraman, “Applied Optimization with Matlab Programming”, John wiley e sons, New York, 2002.

[8] Christopher R. Houck, Jeffery A. Joine and Michael G. Kay, “A Genetic Algorithm for Function Optimization: A Matlab Implementation”, North Carolina State University, available at: http://www.ie.ncsu.edu/mirage/GAToolB.

[9] A. I. A. Cunha, M. C. Schneider, and C. Galup-Montoro, ”An MOS transistor model for analog circuit design. IEEE Journal of Solid-State Circuits,” 33(10):1510{1519, October 1998.

[10] P. E. Allen and D. R. Holberg, CMOS Analog Circuit Design, Oxford University Press, Oxford, second Edition, 2002.