Engineering Applications of Artificial Intelligence 58 (2017) 62–78

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

An enhanced optimization kernel for analog IC design automation using the shrinking circles technique

Maryam Dehbashian, Mohammad Maymandi-Nejad*

Department of Electrical Engineering, Ferdowsi University of Mashhad, Mashhad, Iran

ARTICLE INFO

Keywords:
Analog integrated circuit sizing
Robustness
Optimization
Advanced gravitational search algorithm (AGSA)
The shrinking circles
Corners analysis

ABSTRACT

This paper presents a novel sizing automation tool to improve the efficiency and accuracy of the analog IC design process. A new technique named the "shrinking circles" is proposed to create a balance between the exploration and exploitation capabilities when the optimization algorithm is converging to a possible optimum point. With the help of the shrinking circles concept, an upgraded version of Gravitational Search Algorithm (GSA) named Advanced GSA (AGSA) is proposed to be used as an optimization kernel for our circuit sizing tool. The performance of the proposed AGSA is evaluated over 23 benchmark functions and the results are compared with the standard GSA and Clustered-GSA algorithms. A two-stage op-amp in the 0.35 μm CMOS technology is utilized to validate the performance of the proposed sizing tool. The corners analysis is also performed over the obtained solutions to guarantee their robustness against the process and environmental variations. Finally, a statistical study over the final solutions is conducted from the two aspects of quantitative and qualitative analyses.

1. Introduction

In the 21st century, the growth of the semiconductor industry is moving forward with rapidly increasing speed. According to IC Insights Inc. (McClean, 2015), it is forecast that total semiconductor unit shipments will be increased to 1,024.5 billion devices in 2017 from 32.6 billion in 1978. Moreover, the World Semiconductor Trade Statistics (WSTS) has forecast that the worldwide semiconductor market will reach to $352 billion in 2017 from $10 billion in 1980 (WSTS Inc., 2015).

In the past few years, with more development of the VLSI technologies, it has become possible to integrate analog, digital and RF circuits as a complex mixed-signal systems-on-chip (SoC). The analog circuits occupy a small area of the entire integrated circuit (IC), however, the design of analog circuits is more complex and time consuming compared to digital circuits. The design of analog section, requires highly skilled designers and for this reason, it is known as the bottleneck of IC design (Barros et al., 2010a).

Since the manual design of analog circuits is a tedious and cumbersome task, it is essential to develop Computer Aided Design (CAD) tools for increasing the productivity and quality and reducing the cost and time-to-market of final designs. The analog CAD tools typically consist of topology selection, circuit sizing, and layout synthesis parts. Due to the significant importance of the circuit sizing, the present study concentrates on this area assuming that the designer is able to provide the circuit topology. As shown in Fig. 1, the analog IC sizing tools include two main sections, i.e., synthesis and optimization which are linked together.

Although many research efforts on the development of analog CAD tools, especially circuit sizing methods, have even been made (Alpaydin et al., 2003; Antao and Brodersen, 1995; Barros et al., 2010b; Degrauwe et al., 1987; Koh et al., 1990; Liu et al., 2014; Lourenco et al., 2015; Ochotta et al., 1996; Phelps et al., 2000), the automation of analog circuits design is far behind that of digital circuits regarding the two aspects of technique and methodology, and hence, only a few number of them have been commercialized (del Mar Hershenson et al., 1998; EEDesign, 2009; Synopsys Inc., 2009; AIDAsoft, 2015; Muneda, 2015).

To propose an efficient tool for analog circuit sizing, the existing challenges in this area should be first investigated. The technological advances in the IC fabrication have resulted in very accurate design specifications and highly scaled-down device sizes for modern analog circuits. Thus, one of the challenges is that the existing optimization techniques are not still powerful enough for high-performance analog IC sizing (Liu et al., 2014).

Most of analog IC sizing tools with commercial or academic applications tend to use Genetic Algorithm (GA) to optimize their design process (Barros et al., 2010b; Kruiskamp and Leenaerts, 1995;

* Corresponding author.
E-mail address: maymandi@um.ac.ir (M. Maymandi-Nejad).

http://dx.doi.org/10.1016/j.engappai.2016.11.007
Received 25 April 2016; Received in revised form 2 September 2016; Accepted 22 November 2016
Available online 01 December 2016
0952-1976/

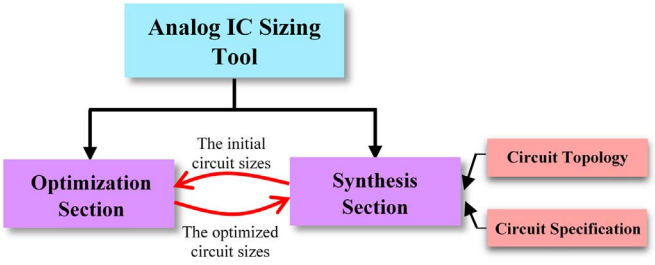

Fig. 1. Fundamental sections of the analog IC sizing tools.

Li, 2009; Liu et al., 2009b; Menozzi et al., 1996). The capability, efficiency and robustness of GA to find an acceptable solution in an appropriate time are the main reasons for the tendency to use this algorithm. However, the search ability, the convergence rate and structural complexities in setting GA parameters have been criticized. For instance, in (Rudolph, 1994) it is proven that a canonical GA cannot converge to the global optimum and it is concluded that in theory, GA with elitism converges to the global optimum, although practically, this is not always possible.

In recent years, swarm intelligence algorithms particularly Particles Swarm Optimization (PSO), have received much attention in the development of analog IC design tools. Some analog IC sizing methods based on swarm intelligence algorithms are summarized in Table 1.

Among these swarm intelligence algorithms, a promising method has been introduced based on Newton's laws of gravity and motion. This method is called the Gravitational Search Algorithm (GSA) proposed by Rashedi et al. (2009). GSA has some positive features such as easy implementation, fast convergence to global optimum and relatively low computational cost (Rashedi et al., 2009, 2010). In most engineering applications, GSA in comparison with other optimization algorithms such as PSO and GA, has shown a better performance (Hemmatian et al., 2014; Mallick et al.; Rashedi et al., 2011; Shams et al., 2015).

In contrast to the aforementioned capabilities of GSA, there is sometimes a possibility of premature convergence. In this case, the algorithm converges to a non-optimal solution and cannot escape from this situation. In effect, when the algorithm converges to a local optimum point, it loses its ability to explore the search space in finding the optimum solution or other good near global optimums. To resolve this problem, different solutions have been presented like adding an extra operator to GSA such as disruption operator (Sarafrazi et al., 2011), chaotic operator (Han and Chang, 2012), and black hole operator (Doraghinejad et al., 2012) or applying other techniques like Clustered-GSA (Shams et al., 2015).

In this paper, a new technique called the *shrinking circles* is proposed. This technique is used to balance the exploration and exploitation abilities when the optimization algorithm is converging

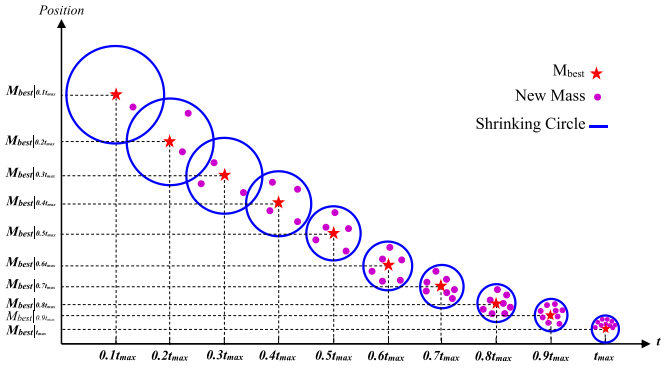

Fig. 2. The search process to find a better position around M_{best} by the shrinking circles technique.

Table 2
Unimodal test functions.

Test function	S
$F_1(X)=\sum_{i=1}^{n} x_i^2$	$[-100, 100]^n$
$F_2(X)=\sum_{i=1}^{n} \lvert x_i\rvert+\prod_{i=1}^{n} \lvert x_i\rvert$	$[-10, 10]^n$
$F_3(X)=\sum_{i=1}^{n}\left(\sum_{j=1}^{i} x_j\right)^2$	$[-100, 100]^n$
$F_4(X)=\max_i \{\lvert x_i\rvert,\ 1 \le i \le n\}$	$[-100, 100]^n$
$F_5(X)=\sum_{i=1}^{n-1} [100(x_{i+1}-x_i^2)^2+(x_i-1)^2]$	$[-30, 30]^n$
$F_6(X)=\sum_{i=1}^{n} ([x_i+0.5])^2$	$[-100, 100]^n$
$F_7(X)=\sum_{i=1}^{n} ix_i^4+random[0, 1)$	$[-1.28, 1.28]^n$

Table 1
Analog IC sizing methods based on swarm intelligence algorithms.

Synthesis	Ref.	Case Study	Technology	Objective	Constraints Handling Method	Algorithm
Equation-based Optimization	(Fakhfakh et al., 2010)	CMOS LNA	AMS 0.35 μm	Voltage Gain ↑	N.M.[a]	PSO
	(Vural and Yildirim, 2012)	Differential amplifier Two-stage op-amp	TSMC 0.35 μm	Area ↓ Area ↓	N.M.	PSO
	(Kotti et al., 2011)	CCII+	AMS 0.35 μm	R_x ↓, f_{ci} ↑	N.M.	PSO, ACO
	(Mallick et al.)	Differential amplifier Two-stage op-amp	TSMC 0.35 μm	Area ↓ Area ↓	N.M.	GSA-PSO
Simulation-based Optimization	Kumar and Duraiswamy, 2012	Two-stage op-amp	N.M.	Gain ↑	With out	PSO
	(Thakker et al., 2009)	CMOS buffer chain Two-stage CMOS op-amp Ultra-low-voltage CMOS Miller OTA High-gain low-power three-stage Op-Amp	0.13 μm 0.13 μm TSMC 0.35 μm, 0.18 μm 0.18 μm UMC	Error Function ↓	N.M.	HPSO
	(Sallem et al., 2010)	CMOS class AB CCII+ Differential-based Class AB CCII+ CMOS OTA based CCII+	0.35 μm 0.35 μm 0.35 μm	Rx ↓	N.M.	Heuristic (Fakhfakh et al., 2007)
	(Shams et al., 2015)	CMOS LNA	N.M.	Noise Figure ↓ S_{21} ↑, S_{12} ↑ S_{11} ↓, S_{22} ↓	Static Penalty Function	Clustered-GSA

[a] Not Mentioned.