Applied Soft Computing 32 (2015) 293–299

Contents lists available at ScienceDirect

Applied Soft Computing

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

Multiobjective design optimization of a nano-CMOS voltage-controlled oscillator using game theoretic-differential evolution

T. Ganesan[a,*], I. Elamvazuthi[b], P. Vasant[c]

[a] Department of Chemical Engineering, Universiti Teknologi PETRONAS, 32610 Bandar Seri Iskandar, Perak Darul Ridzuan, Malaysia
[b] Department of Electrical & Electronics Engineering, Universiti Teknologi PETRONAS, 32610 Bandar Seri Iskandar, Perak Darul Ridzuan, Malaysia
[c] Department of Fundamental & Applied Sciences, Universiti Teknologi PETRONAS, 32610 Bandar Seri Iskandar, Perak Darul Ridzuan, Malaysia

ARTICLE INFO

Article history:
Received 3 August 2013
Received in revised form 15 August 2014
Accepted 8 March 2015
Available online 6 April 2015

Keywords:
Multiobjective
Particle swarm optimization (PSO)
Differential evolution (DE)
Game-theoretic differential evolution (GTDE)
Nano-CMOS VCO
Hypervolume indicator (HVI)

ABSTRACT

Engineering problems presenting themselves in a multiobjective setting have become commonplace in most industries. In such situations the decision maker (DM) requires several solution options prior to selecting the best or the most attractive solution with respect to the current industrial circumstances. The weighted sum scalarization approach was employed in this work in conjunction with three metaheuristic algorithms: particle swarm optimization (PSO), differential evolution (DE) and the improved DE algorithm (GTDE) (which was enhanced using ideas from evolutionary game theory). These methods are then used to generate the approximate Pareto frontier to the nano-CMOS voltage-controlled oscillator (VCO) design problem. Some comparative studies were then carried out to compare the proposed method as compared to the standard DE approach. Examination on the quality of the solutions across the Pareto frontier obtained using these algorithms was carried out using the hypervolume indicator (HVI).

1. Introduction

Many optimization problems are frequently encountered by engineers and decision makers working with systems involving nano-circuits [1–3]. Currently, standard circuit performance optimization is usually carried out manually. This approach usually takes a lot of time and requires plenty of skills. These difficulties compound drastically especially when optimizing circuits at a nano-level. Besides optimization, debugging and troubleshooting such circuit designs can take several days and are usually very costly [4]. In most optimization scenarios, the decision-maker deals with conflicting objective functions such as power consumption factors and voltage-controlled oscillator (VCO) frequency [5,6]. In this work, a multi-objective framework is introduced for the performance optimization of a 45 nm CMOS VCO.

In multi-objective optimization, one approach that has been effective in measuring the quality of the solution set that constructs the Pareto-frontier (in cases where the Pareto frontier is unknown) is the hypervolume indicator (HVI) [7]. Recently, this indicator has been frequently applied in many works involving MO problems [8–10]. The HVI is the only indicator which is strictly Pareto-compliant and can be used to measure the quality of solution sets (degree of dominance) in MO optimization problems [8,11]. In this work, this measurement metric is employed to measure the solution quality and perform comparative analysis.

Metaheuristic approaches have become common place in industries where optimization problems are encountered. One such state-of-the-art approach is differential evolution (DE). DE is a population-based evolutionary algorithm that has been derived from genetic algorithms (GA) [12]. DE was introduced in the mid-nineties by Storn and Price [13]. From then, DE has been employed extensively to solve problems which are nonlinear, non-continuous, noisy, multidimensional, have many local minima, constraints or highly stochastic. For more extensive works involving DE in industrial engineering see [14–16]. Recently, DE has also been employed in MO engineering problems. For instance, in Ganesan et al. [14,17], the optimal parameters were identified for the MO cement-bonded mold system using the DE and Hopfield DE strategies. In the works of Li et al. [18] and de Oliviera et al. [19] the DE technique was employed for antenna array design and in graphic processing unit (GPU) optimization.

* Corresponding author. Tel.: +60 53687550.
E-mail address: tim.ganesan@gmail.com (T. Ganesan).

http://dx.doi.org/10.1016/j.asoc.2015.03.016
1568-4946/

In current times, metaheuristic has become a very common tool when it comes to engineering optimization especially in sectors involving design and operations. For instance, population-based techniques such as DE and Artificial Bee Colony (ABC) [20] have been employed for parametric optimization of turning operations [21–23]. In addition to evolutionary techniques, swarm based algorithms such as Cuckoo Search [24] has also been utilized in milling operations. The central idea in these efforts was to implement metaheuristic techniques to determine the optimal machining parameters for milling [24,25]. Structural design optimization has been a standing problem for many years since this problem is usually multivariate, nonlinear and highly complex. Thus, recently techniques such as evolutionary strategies, particle swarm optimization (PSO) and firefly search have been employed to deal with these problems [26–29]. In these works the metaheuristic is seen to perform well by producing efficient results which often outperform the common practices by a significant margin. Metaheuristics has also been employed in other engineering systems related to assembly line balancing and design optimization in manufacturing [30,31]. In these systems, strategies such as hybrid ABC approaches [32], Taguchi method [33], PSO [34], hybrid simulated annealing [35] and immune system [36,37] have been utilized effectively.

The solution method introduced in this work involves the integration of evolutionary game theory (EGT) into DE to enhance specific search functionalities. EGT has been used in combination with metaheuristic algorithms like PSO to solve problems involving the simulations of evolutionary games [38,39]. In addition, this algorithmic form has been also used to solve optimization test functions where the algorithm's performance was benchmarked [40]. The central aim of this work is to solve and obtain a set of Pareto-efficient solution options for the MO performance optimization of a 45 nm CMOS VCO. The 45 nm CMOS VCO problem was formulated and systematically validated in Chio et al. [40]. The approach proposed in this work is the game-theoretic differential evolution (GTDE). The solutions produced by the GTDE approach is then benchmarked and evaluated using the HVI.

This paper is organized as follows. In Section 2 of this paper, the 45 nm CMOS VCO problem description is presented. In Section 3 the DE, PSO and GTDE techniques are discussed and this is followed by Section 4 which analyzes the computational results. Finally, this paper ends with the concluding remarks and recommendations for future works.

2. Design problem

In Kougianos and Mohanty [41], the design of the nano-CMOS VCO was optimized in a MO framework by employing a baseline design. In the mentioned work, three objectives, the frequency of oscillation (F_{OSC}), average dynamic power (P_{ave}) and leakage power minimization (P_{leak}), were identified. The design parameters are as follows:

1. Gate oxidation thickness (T_{ox})
2. Width to length ratio for the PMOS inverter transistors (β_1)
3. Width to length ratio for the NMOS inverter transistors (β_2)
4. Width to length ratio for the PMOS current-starved transistors (β_3)
5. Width to length ratio for the NMOS current-starved transistors (β_4)

The objective functions and the constraints of the design parameters are given as follows:

$$F_{osc} = 786.43 - 93.36T_{ox} + 60.3\beta_2 \quad (1)$$

$$P_{ave} = 35.05 + 5.7\beta_4 + 3.3\beta_3 \quad (2)$$

$$P_{leak} = 376.35 - 28.58T_{ox} + 29.32\beta_1 + 36.17\beta_2 \quad (3)$$

$$\begin{aligned} &1.4\,\text{nm} \le T_{ox} \le 1.7\,\text{nm} \\ &5 \le \beta_1 \le 10 \\ &1.72 \le \beta_2 \le 3.44 \\ &5 \le \beta_3 \le 10 \\ &1.72 \le \beta_4 \le 3.44 \end{aligned} \quad (4)$$

The MO design optimization of the nano-CMOS VCO problem is shown as follows:

$$\begin{aligned} &\text{Max} \rightarrow F_{osc} \\ &\text{Min} \rightarrow P_{ave} \\ &\text{Min} \rightarrow P_{leak} \\ &\text{subject to design constraint} \quad s \end{aligned} \quad (5)$$

3. Computational approach

3.1. Differential evolution (DE)

DE is a class of evolutionary meta-heuristic algorithms first introduced by Storn and Price [13]. The core idea of this technique is the assimilation of perturbative methods into standard evolutionary algorithms. DE starts by the initialization of a population of at least four individuals denoted as P. These individuals are real-coded vectors with some size N. The initial population of individual vectors (the first generation denoted $gen = 1$) are randomly generated in appropriate search ranges. One principal parent denoted x_i^p and three auxiliary parents denoted x_i^a are randomly selected from the population, P. In DE, every individual, I, in the population, P, would become a principle parent, x_i^p, at one generation or the other and thus have a chance in mating with the auxiliary parents, x_i^a. The three auxiliary parents then engage in 'differential mutation' to generate a mutated vector, V_i.

$$V_i = x_1^a + F(x_2^a - x_3^a) \quad (6)$$

where F is the real-valued mutation amplification factor which is usually between 0 and 1. Next V_i is then recombined (or exponentially crossed-over) with x_i^p to generate child trial vector, x_i^{child}. The probability of the cross-over, CR, is an input parameter set by the user. In DE, the survival selection mechanism into the next generation is called 'knock-out competition'. This is defined as the direct competition between the principle parent, x_i^p, and the child trial vector, x_i^{child}, to select the survivor of the next generation as follows:

$$x_i(gen+1) = \begin{cases} x_i^{child}(gen) \leftrightarrow f(x_i^{child}) \text{ better than } f(x_i^p) \\ x_i^p(gen) \leftrightarrow \text{otherwise} \end{cases} \quad (7)$$

Therefore, the knock-out competition mechanism also serves as the fitness evaluation scheme for the DE algorithm. The parameter setting for the DE algorithm is given in Table 1: The algorithm of DE is shown in Algorithm 1.

Table 1
Differential evolution (DE) parameter settings.

Parameters	Values
Individual size, N	6
Population size, P	7
Mutation amplification factor, F	0.3
Cross-over probability, CR	0.667