European Journal of Operational Research 247 (2015) 732–744

Contents lists available at ScienceDirect

European Journal of Operational Research

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

Continuous Optimization

A novel multi-objective particle swarm optimization with multiple search strategies

Qiuzhen Lin, Jianqiang Li*, Zhihua Du, Jianyong Chen, Zhong Ming

College of Computer Science and Software Engineering, Shenzhen University, Shenzhen 518060, PR China

ARTICLE INFO

Article history:
Received 23 August 2014
Accepted 30 June 2015
Available online 8 July 2015

Keywords:
Multiple objective programming
Particle swarm optimization
Evolutionary computation
Multiple search strategies

ABSTRACT

Recently, multi-objective particle swarm optimization (MOPSO) has shown the effectiveness in solving multi-objective optimization problems (MOPs). However, most MOPSO algorithms only adopt a single search strategy to update the velocity of each particle, which may cause some difficulties when tackling complex MOPs. This paper proposes a novel MOPSO algorithm using multiple search strategies (MMOPSO), where decomposition approach is exploited for transforming MOPs into a set of aggregation problems and then each particle is assigned accordingly to optimize each aggregation problem. Two search strategies are designed to update the velocity of each particle, which is respectively beneficial for the acceleration of convergence speed and the keeping of population diversity. After that, all the non-dominated solutions visited by the particles are preserved in an external archive, where evolutionary search strategy is further performed to exchange useful information among them. These multiple search strategies enable MMOPSO to handle various kinds of MOPs very well. When compared with some MOPSO algorithms and two state-of-the-art evolutionary algorithms, simulation results show that MMOPSO performs better on most of test problems.

1. Introduction

In many real-world engineering applications, the problem that needs to optimize multiple objectives simultaneously is often encountered, which is called multi-objective optimization problems (MOPs) (Deb, Pratap, Agarwal, & Meyarivan, 2002; Ishibuchi & Murata, 1998; Samanlioglu, 2013). For example, the goals in job shop scheduling are commonly required to minimize the makespan, total workload, and critical workload, while the targets in product design are certainly needed to minimize the cost of product and optimize its quality. Since the conflicts exist among the objectives, the improvement of one objective may deteriorate other objectives and resultantly it generates a set of equally-optimal solutions, which is termed *Pareto-optimal set* (*PS*). The corresponding mapping of *PS* in objective space is termed *Pareto-optimal front* (*PF*). As the size of *PF* may be infinite, it is impractical to find out all the Pareto-optimal solutions. Thus, an important job of MOPs is to obtain a finite size of *PS* that is distributed uniformly along the *PF*, which supports the decision maker to select the appropriate solutions for different practical cases (Lin & Chen, 2013; Zhang & Li, 2007).

Currently, nature-inspired metaheuristic algorithms have been recognized to be well suitable for solving MOPs since they can handle some complex problems that are characterized with multimodality, nonlinearity, and discontinuity (Jones, Mirrazavi, & Tamiz, 2002). Among them, particle swarm optimization (PSO) is an interesting nature-inspired algorithm that mimics the social cooperative and competitive behavior of bird flocking and fish schooling (Kennedy & Eberhart, 1995). Due to the fast convergence speed and easy implementation, it has attracted a great interest of researchers and been designed for solving many single-objective optimization problems (SOPs) and various engineering applications (Dang et al., 2013; Nayeri, Yang, & Elsherbeni, 2013; Unler & Murat, 2010). The promising results provided by PSO for solving SOPs validate its effectiveness and efficiency to locate the optimal results in a large and complex problem landscape. This motivates the researchers to extend PSO for MOPs and plenty of multi-objective PSO (MOPSO) algorithms are presented accordingly (Moubayed, Pertovski, & McCall, 2014; Coello Coello, Pulido, & Lechuga, 2004; Goh, Tan, Liu, & Chiam, 2010; Zhan et al., 2013). Generally, most of the existing MOPSO algorithms can be classified into two categories. The first class embeds the Pareto dominance relationship into PSO, which is used to determine the personal best and global best particles (Nebro et al., 2009; Sierra & Coello Coello, 2005; Wang & Yang, 2010). The second kind adopts decomposition approach to transform MOPs into a set of SOPs, where traditional PSO can be directly applied to solve MOPs (Moubayed, Pertovski, & McCall, 2010; Martinez & Coello Coello, 2011; Peng & Zhang, 2008). These MOPSO algorithms perform very well in solving

* Corresponding author. Tel.: +86 75526534407; fax: +86-75526534078.
E-mail address: lijq@szu.edu.cn (J. Li).

http://dx.doi.org/10.1016/j.ejor.2015.06.071

some MOPs. However, when tackling the complex MOPs characterized with multimodality and the existence of many local *PFs*, *e.g.*, WFG1 (Huband, Barone, While, & Hingston, 2005) and DTLZ3 (Deb, Thiele, Laumanns, & Zitzler, 2005), most MOPSO algorithms fail to effectively approach the true *PF*. This is mainly because they only adopt a single search strategy to update the velocity of each particle, which may lack the capabilities to tackle some kinds of complex MOPs.

To repair this weakness, multiple search strategies may be an alternative technology as it has been studied experimentally in PSO for solving SOPs and proven to be an effective and efficient approach to enhance the capabilities of PSO when handling various types of SOPs (Hu, Wu, & Weir, 2013; Li, Yang, & Nguyen, 2012; Zuo, Zhang, & Tan, 2014). Inspired by the reported multiple search strategies for SOPs, it is reasonable to believe that multiple search strategies can be applied in MOPSO to further improve its convergence speed and the robustness when dealing with different kinds of MOPs. Therefore, a novel MOPSO algorithm with multiple search strategies is presented in this paper, called MMOPSO. Decomposition approach is adopted in MMOPSO to decompose MOPs into a set of SOPs and then each particle is assigned to optimize each SOP. Two search strategies for updating the particle's velocity are designed to accelerate the convergence speed and maintain the population diversity respectively. Their cooperation is controlled by a pre-defined threshold. All the non-dominated solutions visited by the particles are stored in a finite-size external archive. Once the external archive is full, only the non-dominated solutions with bigger crowding-distance values will be remained, which are considered to be the elitist solutions and good representatives of the entire *PF*. To let the elitist information be shared among the external archive, an evolutionary search strategy, composed by simulated binary crossover (SBX) and polynomial mutation (PM), is performed, which enhances the exploratory capabilities of MMOPOS. When compared with the existing MOPSO algorithms, the novelty of MMOPSO can be described as follows.

(1) Different from the single search pattern adopted in most MOPSO algorithms, two search strategies are designed in MMOPSO for updating the velocity of each particle, which are aimed at accelerating the convergence speed and maintaining the population diversity respectively. Their executions are determined by a pre-defined threshold to retain the balance of exploitation and exploration.
(2) An evolutionary search strategy is run on the external archive of PSO, which is beneficial for the information exchange among the elitist individuals. The evolutionary operators can provide another search power for PSO and remedy the weaknesses of PSO-based search when handling some difficult MOPs.
(3) New definitions of personal-best and global-best particles are given in MMOPSO. Traditionally, personal-best and global-best particles are the best ones visited by each particle and the swarm respectively. Whereas, in MMOPSO, as decomposition approach is adopted to transform MOPs into a set of SOPs, personal-best and global-best particles are respectively considered to be the best values of each aggregation problem and all SOPs. Therefore, MMOPSO can focus on optimizing each aggregation problem by using PSO search.

The advantages of multiple search strategies will be investigated and validated by the experimental studies. Total 24 standard benchmark problems, including Fonseca (Fonseca & Flemming, 1998), Kursawe (1990), Schaffer (1985), ZDT (Zitzler, Deb, & Thiele, 2000), WFG (Huband et al., 2005) and DTLZ (Deb et al., 2005) series test problems, are utilized to evaluate the comprehensive performance of MMOPSO. When compared with some MOPSO algorithms and two state-of-the-art multi-objective evolutionary algorithms (MOEAs), *e.g.*, DDMOPSO (Moubayed et al., 2014), CMPSO (Zhan et al., 2013), SMPSO (Nebro et al., 2009), dMOPSO (Martinez & Coello Coello, 2011), OMOPSO (Sierra & Coello Coello, 2005), NSGA-II (Deb et al., 2002) and MOEA/D (Li & Zhang, 2009), MMOPSO performs better on most of test problems when considering both of the convergence speed and population diversity.

The rest of this paper is organized as follows. Section 2 introduces the related background, including some important terms of MOPs, decomposition approach, traditional PSO and the existing MOPSO algorithms. In Section 3, the details of MMOPSO are described, where the framework of MMOPSO and multiple search approaches are illustrated. The experimental studies are given in Section 4, which compare the performance of MMOPSO with various multi-objective optimization algorithms and analyze the advantages of multiple search strategies in MMOPSO. At last, conclusions are summarized in Section 5.

2. Related work

2.1. Multi-objective optimization problems

A continuous and unconstrained multi-objective optimization problem can be formulated as follows.

$$\min_{x\in\Omega} F(x) = (f_1(x), f_2(x), \ldots, f_m(x))^T \tag{1}$$

where $x = (x_1, x_2, \ldots, x_n)$ is a n-dimensional decision vector bounded in the *decision space* Ω, m is the number of objective functions and the mapping function F: $\Omega \rightarrow R^m$ defines m objective functions bounded in the *objective space* R^m. Since the objectives often contradict each other, the improvement of one objective may deteriorate other objectives. Therefore, the output of MOPs is generally a set of equally-optimal solutions, which can be determined by Pareto optimality (Bosman & Thierens, 2003).

Definition 1. *(Pareto-dominance)*: A decision vector x is said to dominate another decision vector y (noted as $x \succ y$) if and only if

$$(\forall i \in \{1, 2, \ldots, m\} : f_i(x) \le f_i(y)) \wedge (\exists j \in \{1, 2, \ldots, m\} : f_j(x) < f_j(y)) \tag{2}$$

Definition 2. *(Pareto-optimal)*: A solution x is said to be Pareto-optimal if and only if

$$\neg\exists y \in \Omega : y \succ x. \tag{3}$$

Definition 3. *(Pareto-optimal set)*: The set ***PS*** includes all Pareto-optimal solutions, defined as

$$\boldsymbol{PS} = \{x | \neg\exists y \in \Omega : y \succ x\}. \tag{4}$$

Definition 4. *(Pareto-optimal front)*: The set ***PF*** includes the values of all the objective functions corresponding to the Pareto-optimal solutions in ***PS***.

$$\boldsymbol{PF} = \{F(x) = (f_1(x), f_2(x), \ldots, f_m(x))^T | x \in \boldsymbol{PS}\}. \tag{5}$$

2.2. Decomposition approach

Recently, decomposition approach is widely embedded into nature-inspired metaheuristic for solving MOPs (Gong et al., 2014; Liu, Gu, & Zhang, 2014). It is based on the facts that a Pareto-optimal solution for MOPs, under some mild conditions, could be an optimal solution of a scalar optimization problem, whose optimization target is an aggregation of all the objectives. Therefore, the finding of *PF* can be decomposed into a set of SOPs (Li & Zhang, 2009; Zhang & Li, 2007). Currently, the popular decomposition approaches include the weighted sum, Tchebycheff and boundary intersection approaches. Among them, boundary intersection method has shown certain advantages over the other two approaches as discussed in (Martinez & Coello Coello, 2011; Zhang & Li, 2007). Thus, boundary intersection method is adopted in MMOPSO, which uses the pre-defined weighted vectors λ and a penalty value θ to minimize the distance d_1 to the