Applied Soft Computing 35 (2015) 525–540

Contents lists available at ScienceDirect

Applied Soft Computing

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

A hybrid algorithm based on particle swarm and chemical reaction optimization for multi-object problems

Zhiyong Li [a,*], Tien Trong Nguyen [a,b], ShaoMiao Chen [a], Tung Khac Truong [b]

[a] College of Computer Science and Electronic Engineering, Hunan University, Changsha 410082, China
[b] Faculty of Information Technology, Industrial University of Ho Chi Minh City, Ho Chi Minh, Viet Nam

ARTICLE INFO

Article history:
Received 4 December 2013
Received in revised form 24 June 2015
Accepted 24 June 2015
Available online 9 July 2015

Keywords:
Chemical reaction optimization
Particle swarm optimization
Multi-object optimization
Diversity

ABSTRACT

Over the past decade, the particle swarm optimization (PSO) has been an effective algorithm for solving single and multi-object optimization problems. Recently, the chemical reaction optimization (CRO) algorithm is emerging as a new algorithm used to efficiently solve single-object optimization.

In this paper, we present HP-CRO (hybrid of PSO and CRO) a new hybrid algorithm for multi-object optimization. This algorithm has features of CRO and PSO, HP-CRO creates new molecules (particles) not only used by CRO operations as found in CRO algorithm but also by mechanisms of PSO. The balancing of CRO and PSO operators shows that the method can be used to avoid premature convergence and explore more in the search space.

This paper proposes a model with modified CRO operators and also adding new saving molecules into the external population to increase the diversity. The experimental results of the HP-CRO algorithm compared to some meta-heuristics algorithms such as FMOPSO, MOPSO, NSGAII and SPEA2 show that there is improved efficiency of the HP-CRO algorithm for solving multi-object optimization problems.

1. Introduction

The presence of multiple objectives in a problem, in principle, gives rise to a set optimal solutions (Pareto-optimal) [15] instead of a single solution. One of these Pareto optimal solutions cannot be said to be better than others.

In many real-life problems, objectives under consideration conflict with each other, and optimizing a particular solution with respect to a single objective can result in unacceptable results with respect to the other objectives. A reasonable solution to a multi-objective problem is to investigate a set of solutions, each of which satisfies the objectives at an acceptable level without being dominated by any other solution, and there does not exist any solution that dominates it yet.

In order to design a well structured optimization algorithm for solving multi-object problems, the algorithm should not only good at exploration and good at exploitation but also good to maintaining diversity. If an algorithm is good at exploration searching then it may be poor at exploitation searching and vice versa. In order to achieve good performances on problem optimizations, the two abilities should be well balanced [10].

In recent years, investigating the performance of evolutionary algorithms (EAs) for multi-object optimization problems (MOPs) has attracted a growing interest from the research community. During the past decade, several multi-objective evolutionary algorithms have been proposed and applied in multi-object optimization problems, such as MOPSO [3], NSGAII [7], FMOPSO [24], TV-MOPSO [31], DPSO [33], SPAE2 [38]. The primary reason for this is their ability to find multiple Pareto-optimal solutions in one single simulation run. We will review the most important of them.

The Non-dominate Sorting Genetic Algorithm Version 2 (NSGAII) was proposed by Deb et al. [7], this algorithm can be divided into 3 stages.

Stage 1: The population is classified on the basis of ranking. The ranking of elements is based on the non-dominate. The non-dominate elements of the same rank are classified in the same group.
Stage 2 uses genetic algorithm to find solutions. In this state, the algorithm includes crossover and mutation operators, the crossover operator is a global search operation and the mutation operator is a local search operation. It seems a good method to find solutions. However, throughout the entire algorithm, the crossover operator executes more as compared to mutation operator which seldom executes.

* Corresponding author. Tel.: +86 13607436411.
E-mail address: zhiyong.li@hnu.edu.cn (Z. Li).

http://dx.doi.org/10.1016/j.asoc.2015.06.036
1568-4946/

Stage 3 is used to assign the crowding distance coefficient. The determination of an element is present in the next generation or not based on ranking and crowding distance.

The Multi-object Particle Swarm Optimization (MOPSO) proposed by Coello et al. [3], this basically is a version of PSO for multi-object optimization problems, the combination between PSO and mutation operator to increase search capabilities. This algorithm can be divided into 2 stages.

Stage 1: initialization of population, velocity of each particle, store the positions of the particle that represent non-dominated vectors in the repository. In this paper, repository, external and archive can be used interchangeably.
Stage 2: compute the velocity of particle and the new positions of particles. Next, updating the repository based on the new particles positions and particles in the repository. The authors have added mutation operator that enriches the exploratory capabilities of MOPSO.

The Fuzzy Multiobject Particle Swarm Optimization (FMOPSO) proposed by Liu et al. [24], is similar to the MOPSO, this algorithm used the Synchronous Particle Local Search (SPLS) to improve the distribution of non-dominated solutions and Fuzzy Global Best (f_gbest) to update particle position. The archive is updated each in cycle, e.g., if the candidate solution is not dominated by any particles in the archive, it will be added to the archive. Likewise, any archive particles dominated by this solution will be removed from the archive. The authors also proved that it is effective.

Improving the Strength Pareto Evolutionary Algorithm (SPEA2) proposed by Zitzler et al. [38], this algorithm is a revision of the strength Pareto approach SPEA [39]. This paper suggested maintaining an external archive population at every generation storing all non-dominated solutions discovered so far beginning from the initial population. This external population participates in all genetic operations.

At each generation, a combined population with the external and current population is first structured. All non-dominated solutions in the combined population are assigned a fitness based on the number of solutions the dominate and dominated solutions are assigned fitness worse than the worst fitness of any non-dominated solution. A deterministic clustering technique is used to ensure diversity among non-dominated solutions.

However, all four algorithms have weaknesses:

- The algorithms execute mostly the global search and rarely the local search, which leads to algorithm imbalance which is the main reason they cannot find good solutions and may not avoid premature convergence.
- Using the global and local search operators based on probability is inefficient.

Particle swarm optimization [12] is a stochastic optimization technique that is inspired by the behavior of bird flocks. It is very simple to simulate and has been found to be quite efficient in handling the single objective optimization. Recently, investigating the performance of PSO for multi-object optimization problems has been steadily gaining attention from the research community [3,24,31,33]. PSO is well-known with for global search efficient and not really efficient in local search. This is the reason some authors add local search operator to increase search ability [3,24].

CRO is an evolutionary optimization technique developed by Lam and Li [18]. CRO is an optimization technique inspired by chemical reaction process. It mimics the interactions of molecules in a chemical reaction to reach a low energy stable state. It has emerged as a new algorithm executing very efficiently in solving single-object optimization [17,19,32] with local search efficiency and not really efficient in global search. This point has been proven in [25].

Taking advantage of the compensatory property of CRO and PSO we propose a new algorithm that combines the evolutionary natures of both for multi-object optimization.

The contribution of this paper as follows:

- A new hybrid method is proposed for multi-object optimization. It is good not only at exploration for global search, but also good at exploitation for local search.
- The modified chemical reaction optimization operators are proposed for multi-object optimization.
- A new parameter is proposed to balance between CRO and PSO operators.
- Based on crowding distance mechanism, a new method is proposed to increase the diversity of archiving solutions.

This algorithm for multi-object optimization indicates not only finding better solutions but also helps achieving a good diversity with respect to the objective space.

The remainder of this paper is organized as follows: the next section, we outline the related works. Additionally, we present the modified version of CRO to solve multi-object optimization problems. The description of the proposed HP-CRO is presented in Section 3. In Section 4, we introduce seven problems test study, experimental simulation and results. Finally, conclusion and future works are present in Section 5.

2. Related works

2.1. Concept of multi-objective optimization

A general multi-objective optimization minimization problems can be defined as [15]:

$$\begin{aligned} &\text{minimize:} && \vec{f}(\vec{x}) = [f_1(\vec{x}), f_2(\vec{x}), \ldots, f_n(\vec{x})] \\ &\text{subject to:} && g_i(\vec{x}) \le 0, \quad i = 1, 2, \ldots, m, \\ & && h_j(\vec{x}) = 0, \quad j = 1, 2, \ldots, p, \end{aligned}$$

where $\vec{x} = (x_1, x_2, \ldots, x_n)$ is the vector on the decision search space; $g_i(\vec{x}) \le 0$ and $h_j(\vec{x}) = 0$ are the constraint functions of the problem.

Given two vectors $\vec{x}, \vec{y} \in R^n$, $\vec{x}$ dominates $\vec{y}$ (denoted $\vec{x} \prec \vec{y}$) if $\vec{x}$ is better than $\vec{y}$ in at least one objective and is not worse than $\vec{y}$ in any objective. $\vec{x}$ is not dominated does not exist another current solution $\vec{x}_i$ in the current population, such that $\vec{x}_i \prec \vec{x}$. The set of non-dominated solutions in the objective space is known as Pareto front[1] (PF^*).

2.2. The chemical reaction algorithm

In CRO, a candidate solution for a specific problem is encoded as a molecule. Each molecule represents a point in the search space, and hence a possible solution to the problem. A population consists of a finite number of molecules, each molecule is decided by an evaluating mechanism to obtain its potential energy (PE[2]). Based on this potential energy and undergoing CRO operators, a new molecule(s) is generated.

In a chemical reaction process, a sequence of collisions among molecules occurs. Molecules collide either with each other or

[1] Download from http://delta.cs.cinvestav.mx/~ccoello/EMOO/testfuncs/.
[2] PE will be introduced in Section 2.3.